



CACIC 2016

Del 3 al 7 de Octubre de 2016
Universidad Nacional de San Luis

XXII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

Libro de Actas



XXII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN CACIC 2016: OCTUBRE 2016, SAN LUIS, ARGENTINA

Organizadores:

Red de Universidades con Carreras en Informática RedUNCI

Universidad Nacional de San Luis

Facultad de Ciencias Físico Matemáticas y Naturales

Departamento de Informática.

Fecha de Catalogación:

Libro de Actas CACIC 2016

San Luis : Universidad Nacional de San Luis

RedUNCI, 2016.

Libro digital, PDF

ISBN (En construcción)



AUTORIDADES RedUNCI

Coordinador Titular

Feierherd Guillermo (Universidad Nacional de Tierra Del Fuego)

Coordinador Alterno

Pesado Patricia (Universidad Nacional de La Plata)

Junta Directiva

Piccoli Fabiana (Universidad Nacional de San Luis)

Estayno Marcelo (Universidad Nacional de Lomas de Zamora)

Dapozo Gladys (Universidad Nacional del Nordeste)

Padovani Hugo (Universidad de Morón)

Kuna Horacio (Universidad Nacional de Misiones)

Finocchieto Jorge (Universidad CAECE)

Spositto Osvaldo (Universidad Nacional de La Matanza)

Russo Claudia (Universidad Nacional del Noroeste de Buenos Aires)



AUTORIDADES DE LA UNIVERSIDAD NACIONAL DE SAN LUIS

Rectorado

Rector:

Dr. Felix Nieto Quintas.

Vice-rector:

Dr. Roberto Saa

Secretaria Académica:

Dra. Marcela Printista.

Facultad de Ciencias Físico Matemáticas y Naturales

Decano:

Dr. Fernando Bulnes

Vicedecano:

MCs. Roberto Guerrero

Secretario Académico:

Lic. Nicolás Cortes

Departamento de Informática

Directora:

Dra. María Fabiana Piccoli

Subdirectora:

Esp. Irma Pianucci



COMITÉ ORGANIZADOR LOCAL

Responsable

Dra. María Fabiana Piccoli

Sub-responsable

Dra. Nora Reyes

Miembros

Lic. Susana Esquivel

Lic. Jacqueline Fernández

MCs. Patricia Roggero

Lic. Apolloni, Javier

Dra. Marcela Printista

Dra. Gisella Dorzán

Tec. José Navratil

Sec. Cecilia Betervide



**COMITÉ ACADÉMICO DEL
XXII CONGRESO ARGENTINO DE CIENCIAS DE LA
COMPUTACIÓN
(CACIC 2016)**

Fernández Slezak Diego (**Universidad de Buenos Aires – Cs. Exactas**)
Echeverría, Adriana (**Universidad de Buenos Aires – Ingeniería**)
Pesado, Patricia (**Universidad Nacional de La Plata**)
Rueda Sonia (**Universidad Nacional del Sur**)
Piccoli, Fabiana (**Universidad Nacional de San Luis**)
Aciti Claudio(**Universidad Nacional del Centro de la Provincia de Buenos Aires**)
Balladini, Javier(**Universidad Nacional del Comahue**)
Sposito, Osvaldo (**Universidad Nacional de La Matanza**)
Alfonso, Hugo(**Universidad Nacional de La Pampa**)
Estayno, Marcelo(**Universidad Nacional de Lomas de Zamora**)
Feierherd, Guillermo(**Universidad Nacional de Tierra del Fuego**)
Gil, Gustavo (**Universidad Nacional de Salta**)
Márquez, María Eugenia (**Universidad Nacional de la Patagonia Austral**)
Otazú, Alejandra (**Universidad Nacional de San Juan**)
Aranguren, Silvia(**Universidad Autónoma de Entre Ríos**)
Buckle, Carlos (**Universidad Nacional de la Patagonia San Juan Bosco**)
Tugnarelli, Mónica (**Universidad Nacional de Entre Ríos**)
Dapozo, Gladys (**Universidad Nacional del Nordeste**)
Kantor Raul (**Universidad Nacional de Rosario**)
Kuna, Horacio (**Universidad Nacional de Misiones**)
Russo, Claudia (**Universidad Nacional del Noroeste de Buenos Aires**)
Carmona, Fernanda (**Universidad Nacional de Chilecito**)
García Martínez, Ramón (**Universidad Nacional de Lanús**)
Duran Elena (**Universidad Nacional de Santiago del Estero**)
Arroyo Arzubi, Alejandro (**Esc. Sup. Técnica - Facultad de Ingeniería del Ejército**)



Loyarte, Horacio (**Universidad Nacional del Litoral**)
Arroyo, Marcelo (**Universidad Nacional de Río Cuarto**)
Fridlender, Daniel (**Universidad Nacional de Córdoba**)
Herrera Cognetta, Analía (**Universidad Nacional de Jujuy**)
Vivas, Luis (**Universidad Nacional de Río Negro**)
Prato, Laura (**Universidad Nacional de Villa María**)
Panessi, Wálter (**Universidad Nacional de Luján**)
Barrera, María Alejandra (**Universidad Nacional de Catamarca**)
Campazzo Eduardo (**Universidad Nacional de La Rioja**)
Oliveros, Alejandro (**Universidad Nacional de Tres de Febrero**)
Luccioni, Griselda María (**Universidad Nacional de Tucumán**)
Morales, Martín (**Universidad Nacional Arturo Jauretche**)
Zachman Patricia (**Universidad Nacional del Chaco Austral**)
Foti, Antonio (**Universidad Nacional del Oeste**)
Forradelas, Raymundo (**Universidad Nacional de Cuyo**)
Doumecq, Julio Cesar (**Universidad Nacional de Mar del Plata**)
Díaz-Caro, Alejandro (**Universidad Nacional de Quilmes**)
Padovani Hugo (**Universidad de Morón**)
De Vincenzi, Marcelo (**Universidad Abierta Interamericana**)
Guerci, Alberto (**Universidad de Belgrano**)
Panizzi, Marisa (**Universidad Kennedy**)
Bournissen Juan (**Universidad Adventista del Plata**)
Finocchietto, Jorge (**Universidad CAECE**)
Alvarez Adriana (**Universidad de Palermo**)
Grieco, Sebastián (**Universidad Católica Argentina Rosario**)
Zanitti, Marcelo (**Universidad del Salvador**)
Giménez, Rosa (**Universidad del Aconcagua**)
Beyersdorf, Carlos (**Universidad Gastón Dachary**)
Guglianone, Ariadna (**Universidad del CEMA**)
Cassol, Ignacio (**Universidad Austral**)
Rathmann, Liliana (**Universidad Atlántida Argentina**)
Bertone, Rodolfo (**Universidad Católica de La Plata**)
Gomez, Silvia (**Instituto Tecnológico de Buenos Aires**)
Pinciroli, Fernando (**Universidad Champagnat**)



COMITÉ CIENTÍFICO DEL XXII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN (CACIC 2016)

- | | |
|---------------------------------------|------------------------------------|
| Abásolo, María José (Argentina) | Depetris, Beatriz (Argentina) |
| Acosta, Nelson (Argentina) | Diaz, Javier (Argentina) |
| Alfonso, Hugo (Argentina) | Dix, Juerguen (Alemania) |
| Ardenghi, Jorge (Argentina) | Doallo, Ramón (España) |
| Arroyo, Marcelo (Argentina) | Docampo, Domingo (España) |
| Astudillo Hernán (Chile) | Dujmovic, Jozo (USA) |
| Baldasarri, Sandra (España) | Echaiz, Javier (Argentina) |
| Balladini, Javier (Argentina) | Esquivel, Susana (Argentina) |
| Barbosa, Luiz (Portugal) | Estayno, Marcelo (Argentina) |
| Bertone, Rodolfo (Argentina) | Estevez, Elsa (Argentina) |
| Bría, Oscar (Argentina) | Falappa, Marcelo (Argentina) |
| Brisaboa, Nieves (España) | Feierherd, Guillermo (Argentina) |
| Buckle, Carlos (Argentina) | Fillottrani, Pablo (Argentina) |
| Castro Lechtaler, Antonio (Argentina) | Finocchietto, Jorge (Argentina) |
| Castro, Silvia (Argentina) | Fleischman, William (USA) |
| Cechich, Alejandra (Argentina) | Frati, Emanuel (Argentina) |
| Chavez, Edgar (México) | García Garino, Carlos (Argentina) |
| Coello Coello, Carlos (México) | García Villalba, Javier (España) |
| Constantini, Roberto (Argentina) | Género, Marcela (España) |
| Cuevas, Alfredo Simón (Cuba) | Giacomantone, Javier (Argentina) |
| Cukierman, Uriel (Argentina) | Gómez, Sergio (Argentina) |
| De Giusti, Armando (Argentina) | Gröller, Eduard (Austria) |
| De Giusti, Laura (Argentina) | Guerrero, Roberto (Argentina) |
| De Vincenzi, Marcelo (Argentina) | Janowski, Tomasz (Naciones Unidas) |
| Deco, Claudia (Argentina) | Kantor, Raul (Argentina) |



Kuna, Horacio (Argentina)
Lanzarini, Laura (Argentina)
Leguizamón, Guillermo (Argentina)
Lopez Gil, Fernando (España)
Loui, Ronald Prescott (EEUU)
Luque, Emilio (España)
Madoz, Cristina (Argentina)
Malberti, Alejandra (Argentina)
Malbran, Maria (Argentina)
Manresa Yee, Cristina (España)
Marco Javier (España)
Marín, Mauricio (Chile)
Mas Sansó, Ramón (España)
Matrángolo Carlos (Argentina)
Micolini, Orlando (Argentina)
Motz, Regina (Uruguay)
Naiouf, Marcelo (Argentina)
Navarro Martín, Antonio (España)
Olivas Varela, José Angel (España)
Padovani, Hugo (Argentina)
Pandolfi, Daniel (Argentina)
Pardo, Álvaro (Uruguay)
Pesado, Patricia (Argentina)
Piattini, Mario (España)
Piccoli, María Fabiana (Argentina)
Printista, Marcela (Argentina)
Puppo, Enrico (Italia)
Ramón, Hugo (Argentina)
Rexachs, Dolores (España)
Reyes, Nora (Argentina)
Riesco, Daniel (Argentina)
Roig Vila, Rosabel (España)
Rossi, Gustavo (Argentina)
Rosso, Paolo (España)
Rueda, Sonia (Argentina)
Ruiz Francisco (España)
Russo, Claudia (Argentina)
Sanz, Cecilia (Argentina)
Saroka, Raúl (Argentina)
Simari, Guillermo (Argentina)
Spositto, Osvaldo (Argentina)
Steinmetz, Ralf (Alemania)
Suppi, Remo (España)
Tarouco, Liane (Brasil)
Tirado, Francisco (España)
Vaquila Isidoro (Argentina)
Velho, Luiz (Brasil)
Vendrell, Eduardo (España)
Vénere, Marcelo (Argentina)
Villagarcia Wanza, Horacio (Argentina)
Zanarini, Dante (Argentina)



Actas del XXII CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN

(CACIC 2016)

(Octubre 2016, San Luis, Argentina)

XVII Workshop Agentes y Sistemas Inteligentes (WASI)	1
XVI Workshop Procesamiento Distribuido y Paralelo (WPDP)	137
XIV Workshop Computación Gráfica, Imágenes y Visualización (WCGIV)	220
XIII Workshop Tecnología Informática Aplicada en Educación (WTIAE)	301
XIII -Workshop Ingeniería de Software (WIS)	462
XIII Workshop Bases de Datos y Minería de Datos (WBDMD)	659
XI Workshop Arquitectura, Redes y Sistemas Operativos (WARSO)	842
VIII Workshop Innovación en Sistemas de Software (WISS)	914
VII Workshop Procesamiento de Señales y Sistemas de Tiempo Real (WPSTR)	1052
V Workshop de Seguridad Informática (WSI)	1113
V Workshop de Innovación en Educación en Informática (WIEI)	1185
V ETHICOMP Latinoamérica	1367



XVII Workshop Agentes y Sistemas Inteligentes (WASI)

Comité Científico:

Alejandro García (Argentina)
Alejandro Zunino (Argentina)
Ana Casali (Argentina)
Ana Maguitman (Argentina)
Carlos Chesñevar (Argentina)
Carolina Salto (Argentina)
Daniela Godoy (Argentina)
Diego Martínez (Argentina)
Facundo Bromberg (Argentina)
Francisco Bavera (Argentina)
Gabriela Minetti (Argentina)
Guillermo Aguirre (Argentina)
Guillermo Leguizamón (Argentina)
Guillermo R. Simari (Argentina)
Hugo Alfonso (Argentina)
Ignacio Ponzoni (Argentina)
Jessica Carballido (Argentina)
José Castaño (Argentina)
Laura Alonso Alemany (Argentina)
Laura Cecchi (Argentina)
Laura Lanzarini (Argentina)
Leticia Cagnina (Argentina)
Luciano Tamargo (Argentina)
Marcelo Armentano (Argentina)
Marcelo Errecalde (Argentina)
Marcelo Falappa (Argentina)
Pablo Ponzio (Argentina)
Silvia Schiaffino (Argentina)
Valentín Cassano (Canadá)
Victoria Aragón (Argentina)



XVII Workshop Agentes y Sistemas Inteligentes (WASI)

ID	Trabajo	Autores	
9116	Ajuste de Parámetros Evolutivos para el Método Paralelo de Reducción de Incertidumbre ESSIM-DE	Laura Tardivo (UNRC), Paola Caymes Scutari (UTN-FRM), Germán Bianchini (UTN-FRM), Miguel Mendez Garabetti (UNCUYO)	4
9178	On the Application of Argument Accrual to Reasoning with Inconsistent Possibilistic Ontologies	Sergio Alejandro Gómez (UNS)	14
9181	Agent-based simulation of a project fractal company	María de las Mercedes Canavesio (UTN-FRSF), Constanza Quaglia (UTN-FRSF), Ernesto Martínez (UTN-FRSF)	24
9185	Clasificación Automática de Textos Periodísticos usando Random Forest.	Javier Izetta Riera (UNJu), Juan Salinas (UBA)	34
9211	Multiple Revision on Horn Belief Bases	Nestor JorgeValdez(UNCa), Marcelo A. Falappa (UNS)	45
9247	Procesamiento inteligente de flujos de datos vía sistemas argumentativos	José N. Paredes (UNS), Fabio R. Gallo (UNS), Gerardo I. Simari (UNS), Marcelo A. Falappa (UNS)	55
9270	Vigilancia Tecnológica e Inteligencia Competitiva basada en Técnicas de Minería de la Web	Karina Eckert (UGD), Victor M. Alvarenga (UGD), Matías M. Barboza (UGD), Leandro M. Witzke (UGD), Luis Airaldi (UTN-FRRe)	65
9277	Comparison of Different Approaches for Adapting Mutation Probabilities in Genetic Algorithms	Natalia Silvana Stark (UNLPAM), Gabriela Minetti (UNLPAM), Carolina Salto (UNLPAM)	75
9282	Introducing a Context-Aware Scheme in an intelligent reasoning process	Paola Budán (UNSE), Maximiliano Budán (UNS), María Vanina Martínez (UNS), Guillermo Ricardo Simari (UNS)	85
9308	Immune Algorithm for Solving the Dynamic Economic Dispatch	Victoria Aragón (UNSL), Susana Esquivel (UNSL)	95



Problem

ID	Trabajo	Autores	
9319	ASrank: un análisis de robustez	Fernando Sanz Troiani (UNLPAM), Gabriela Minetti (UNLPAM)	105
9323	Optimización de cadenas de adición	Fernando Aquino (UTN-FRCU), Guillermo Leguizamón (UNSL)	114
9330	Una propuesta para mejorar la comunicación mediante persuasión	Guillermo Aguirre (UNSL), Marcelo Errecalde (UNSL), Guillermo Ricardo Simari (UNS)	127

Ajuste de Parámetros Evolutivos para el Método Paralelo de Reducción de Incertidumbre ESSIM-DE

María Laura Tardivo^{1,2,3}, Paola Caymes-Scutari^{1,2}, Germán Bianchini¹ y Miguel Méndez-Garabetti^{1,2}

¹ Laboratorio de Investigación en Cómputo Paralelo/Distribuido (LICPaD), Departamento de Ingeniería en Sistemas de Información (UTN-FRM), Mendoza, Argentina.

² Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET),

³ Departamento de Computación, Facultad de Ciencias Exactas, Físico-Químicas y Naturales (UNRC), Córdoba, Argentina.

lauratardivo@dc.exa.unrc.edu.ar, {pcaymesscutari,gbianchini}@frm.utn.edu.ar, mmendez@mendoza-conicet.gob.ar

Abstract. Los incendios forestales son uno de los fenómenos naturales más perjudiciales del planeta. Año tras año devastan miles de hectáreas y causan cambios irreversibles en las zonas afectadas. Por tal motivo, surge el interés de contar con herramientas que puedan realizar pronósticos con antelación al desarrollo de los incendios. Sin embargo, los sistemas de predicción suelen presentar restricciones, no solo por las limitaciones computacionales de representación, sino también por la existencia de incertidumbre en los datos de entrada, debido a la dificultad de determinar sus valores con exactitud en tiempo real. Una adecuada determinación de los parámetros de entrada puede mejorar significativamente la calidad de predicción del método. En este trabajo se presenta un estudio de sintonización estática de los parámetros evolutivos del Sistema Estadístico Evolutivo con Modelo de Islas y Evolución Diferencial, con el fin de mejorar la calidad de la predicción.

Keywords: Reducción de Incertidumbre, Paralelismo, Análisis Estadístico, Sintonización Estática, Incendios Forestales.

1 Introducción

Los incendios forestales son el resultado de interacciones complejas entre las condiciones del medio ambiente, la topografía del terreno, los focos de ignición y el tipo de combustible presente en el terreno. Si bien son un componente necesario para mantener el equilibrio de los ecosistemas, se pueden convertir en una gran amenaza para la población bajo ciertas condiciones ambientales, como sequía prolongada y temperaturas altas. En la actualidad se realiza un gran esfuerzo a nivel científico para desarrollar técnicas y estrategias que logren reducir los efectos causados por los

incendios. En particular, el desarrollo de métodos de predicción puede colaborar de manera significativa en el proceso de lucha contra el fuego. La predicción de cualquier fenómeno natural implica la construcción de modelos que permitan describir su comportamiento, a fin de representar de la mejor manera posible la complejidad inherente al fenómeno. Generalmente, los modelos reciben un conjunto de parámetros de entrada representando las condiciones particulares del fenómeno, y proveen una salida que refleja la evolución del mismo. En el caso de los incendios forestales, los parámetros pueden representar la dirección y velocidad del viento, la pendiente del terreno, la humedad y tipo de vegetación, etc. Sin embargo, alguno de estos parámetros poseen valores que cambian dinámicamente durante el desarrollo del incendio, y además presentan dificultades para ser medidos en tiempo real. Esta falta de precisión se conoce como incertidumbre, y para que el modelo se aproxime de la mejor manera posible a la realidad, se hace necesario contar con estrategias de reducción de incertidumbre en los parámetros de entrada.

El Sistema Estadístico Evolutivo con Modelo de Islas y Evolución Diferencial (ESSIM-DE, por sus siglas en inglés) es un método de simulación y reducción de incertidumbre que ha sido aplicado satisfactoriamente al problema de la predicción de incendios forestales [8]. ESSIM-DE utiliza Análisis Estadístico, Computación de Alto Desempeño y la metaheurística Evolución Diferencial como método de optimización. ESSIM-DE posee dos clases de parámetros de entrada. Por un lado, incluye los parámetros que se utilizan para representar las condiciones inherentes al incendio forestal. Entre ellos encontramos las condiciones climáticas del entorno (velocidad y dirección del viento) y las condiciones del terreno donde se desarrolla el incendio (pendiente del terreno, tipo de vegetación, humedad de la vegetación, etc). Por otro lado, posee parámetros de control relativos a los métodos de cálculo que utiliza en su esquema de procesamiento. Una adecuada elección de los parámetros de control implica obtener una mejor caracterización de las posibles zonas afectadas por el fuego, brindando una mejor representación del fenómeno real. En este trabajo presentamos un estudio de calibración estática sobre los parámetros evolutivos de ESSIM-DE. Estos parámetros son necesarios para establecer la operatividad del método de manera tal de orientar la búsqueda hacia mejores soluciones.

El trabajo se organiza de la siguiente manera. La sección 2 describe brevemente los diferentes métodos de predicción. La sección 3 detalla el método ESSIM-DE, incluyendo las diferentes etapas de su modo de operación y su diseño computacional. La sección 4 presenta los experimentos realizados para calibrar los parámetros evolutivos de ESSIM-DE, incluyendo los resultados obtenidos. Finalmente, la sección 5 enuncia las principales conclusiones y trabajos a futuro.

2 Métodos de predicción

En términos generales, los métodos de Predicción Clásica consisten en evaluar la posición del fuego luego de cierto período inicial, utilizando algún tipo de simulador del comportamiento de incendios. Un esquema general de este tipo de métodos se

describe en la Figura 1. El Simulador de Incendios (SI) es alimentado por dos conjunto de datos: la línea de fuego real del incendio en el tiempo t_n (LFR_n), y la información que describe el entorno en el cual se desarrolla el incendio.

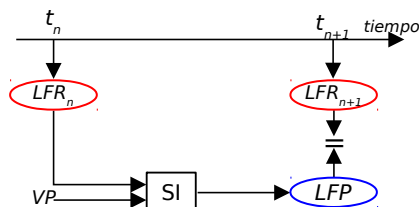


Fig. 1. Predicción Clásica (SI: Simulador de Incendios, LFR: Línea de Fuego Real, LFP: Línea de Fuego de Predicción, LFR_n : Línea de Fuego Real en el instante n , VP: vector de parámetros).

Generalmente la línea de fuego real es representada por un mapa que describe el área en donde el fuego se ha iniciado para un instante de tiempo determinado. Cada parámetro de entrada tiene un valor asignado, y el conjunto de parámetros es utilizado junto con la LFR_n por el SI para realizar la predicción de la línea de fuego (LFP) para el siguiente instante de tiempo (t_{n+1}), a través de una única simulación. Ejemplos de predicción clásica aplicados a incendios forestales son [1] y [2]. Dadas la complejidad del modelo, la incertidumbre sobre los parámetros de entrada y debido a que la predicción está basada en una única simulación, esta metodología de predicción generalmente provee predicciones que son lejanas a la realidad.

Otro tipo de estrategias de predicción son los Métodos Conducidos por Datos con Múltiples Soluciones Solapadas (Data Driven Methods with Multiple Overlapped Solutions- DDM-MOS). Los DDM-MOS consideran un mayor número de valores para cada parámetro, en vez de considerar un solo valor como los métodos clásicos. Los DDM-MOS realizan un proceso de búsqueda utilizando algún algoritmo o heurística para encontrar el conjunto de parámetros que mejor describe el comportamiento previo del incendio, de manera tal que pueda ser utilizado para predecir el comportamiento futuro cercano de la evolución del incendio, basado en el principio de localidad espacial y temporal. Finalmente los DDM-MOS generan un conjunto de posibles soluciones con casos solapados y combinaciones de parámetros para realizar predicciones. Ejemplos de DDM-MOS son [3] [4] y [5]. La operatividad de estos métodos se basa en la utilización de Análisis Estadístico, Computación de Alto Desempeño y Algoritmos Evolutivos Paralelos.

3 Sistema Estadístico Evolutivo con Modelo de Islas y Evolución Diferencial (ESSIM-DE)

El Sistema Estadístico Evolutivo con Modelo de Islas y Evolución Diferencial (ESSIM-DE) es un método de reducción de incertidumbre que se enmarca dentro de

los DDM-MOS y que ha sido aplicado satisfactoriamente a la predicción de incendios forestales. ESSIM-DE utiliza como metaheurística principal a Evolución Diferencial para orientar la búsqueda hacia mejores soluciones en cada paso de simulación y además utiliza una doble jerarquía Master/Worker en su esquema de comunicación.

3.1 Evolución Diferencial

El algoritmo Evolución Diferencial (DE) se clasifica dentro de los Algoritmos Evolutivos y ha sido utilizado con éxito en diferentes problemas de optimización de funciones [6]. En su operador de mutación, utiliza diferencias de vectores para orientar la búsqueda hacia soluciones de mejor calidad. DE comienza generando una población inicial de individuos con valores aleatorios dentro de los rangos del problema. En cada generación g se recorre la población y por cada individuo $X_{i,g}$ el proceso de mutación consiste en seleccionar al azar tres individuos de la población diferentes entre sí ($X_{r1,g}$, $X_{r2,g}$ y $X_{r3,g}$). Estos tres individuos son utilizados en un operador de recombinación basado en la diferencia de vectores y en un factor de mutación F para generar un nuevo vector $V_{i,g}$, denominado *vector mutante*.

En la etapa de cruzamiento, el vector mutante es combinado con el individuo $X_{i,g}$ para generar un nuevo vector $U_{i,g}$, denominado *vector trial*. Este vector posee características de ambos progenitores y la herencia es regulada por un parámetro del algoritmo denominado *probabilidad de cruzamiento (CR)*. Finalmente, en la etapa de selección se determina el mejor candidato entre el individuo corriente $X_{i,g}$ y el vector trial $U_{i,g}$. Aquel que tenga mejor valor respecto de la función a optimizar será el que sobrevivirá a la generación siguiente, es decir, siendo f la función a optimizar, $U_{i,g}$ reemplaza a $X_{i,g}$ si y solo si $f(U_{i,g}) < f(X_{i,g})$ (para problemas de minimización), caso contrario, no ocurre reemplazo o selección.

Como puede observarse, el algoritmo DE sigue un esquema de procesamiento secuencial, iterando sobre la población a través de las generaciones y aplicando los operadores evolutivos. Si bien DE tiene un gran potencial y ha sido utilizado en una gran variedad de problemas de optimización concretos, su desempeño puede ser mejorado agregando modificaciones a la estructura de procesamiento original con el fin de obtener mejoras significativas [7]. En este trabajo se utiliza un esquema de paralelización de doble jerarquía y modelo de islas.

3.2 ESSIM-DE: Descripción operativa

El Modelo de Islas consiste en organizar los procesos en grupos, cada uno con una población diferente. Las islas intercambian individuos de manera periódica a través del proceso de migración. Desde una visión jerárquica, los procesos están organizados en dos niveles. En una jerarquía superior se dispone de un proceso denominado **Monitor**, que es el encargado de enviar información de inicialización a cada isla, recolectar los datos procesados en la etapa final de la simulación y determinar los valores de salida, identificando cuál es la isla que ha resultado con mejor aptitud. La Figura 2 describe el diagrama general de ESSIM-DE. Se puede observar que el

componente **Monitor** está relacionado con cada componente **Isla** a través de la comunicación antes mencionada. Por su parte, el nivel inferior de jerarquía responde a un esquema de isla. En cada isla un proceso Master es el encargado de inicializar la población inicial y aplicar los operadores de mutación y cruzamiento para generar nuevos individuos (etapa **EO-Master**). Para determinar si los nuevos individuos generados permanecerán en la generación siguiente, el Master distribuye cada individuo generado entre ciertos procesos workers, quienes evalúan el desempeño de cada uno aplicando la función de aptitud y retornando al Master la evaluación obtenida. En la Figura 2, la comunicación entre el Master y los workers está representada por flechas de doble dirección que unen los componentes **Master** y **EO-Worker**. La Etapa de Optimización (**EO-Worker**) es llevada a cabo por los procesos workers. Esta etapa realiza la evaluación de la aptitud de los individuos mediante dos subetapas internas denominadas Simulación de Incendios (**SI**) y Evolución Diferencial (**DEA**). **SI** debe ser alimentada con la línea de fuego real del incendio en el instante de tiempo t_i-1 (**LFRi-1**) junto con el vector de parámetros de entrada (**VP**). Cuando **SI** concluye con la simulación de los individuos, el resultado de cada simulación es introducido en la etapa **DEA** para comparar el mapa simulado con el mapa real para el instante de tiempo t_i (**LFRi**). Aquí, según la diferencia entre el mapa simulado y el mapa real, se calcula el valor de aptitud para cada individuo.

A medida que los workers van obteniendo resultados, envían estos valores al Master, quien determina si los individuos generados en el proceso de mutación y cruzamiento sobrevivirán a la siguiente generación. Para ello, verifica si la aptitud calculada por los workers es mayor a la aptitud del individuo corriente, en cuyo caso, el nuevo individuo generado reemplaza al individuo corriente. Este proceso se realiza para cada individuo de la población. El Master también se encarga de efectuar la migración grupal de individuos hacia otra isla. En la etapa de **migración** se selecciona un conjunto de individuos de la población actual para ser enviados a otra isla. Entre los individuos seleccionados se incluyen los dos mejores de la población (aquellos dos que tengan los mejores valores de aptitud) y el resto del conjunto es seleccionado al azar. Los individuos que arriban reemplazan a los peores individuos de la población destino. Esta migración grupal permite incrementar la diversidad poblacional en cada isla, a partir de la incorporación de nuevos individuos que han evolucionado con las características de la población de la isla de partida, y que se incorporarán, evolucionarán y desarrollarán en la población destino.

Una vez que la población alcanza cierto nivel de aptitud (o umbral de aptitud U), es introducida en la Etapa de Calibración (**EC-Master**). En esta etapa la población que ha evolucionado alimenta a una subetapa denominada Etapa Estadística (**EE**). La salida de **EE** es un mapa de probabilidad que es utilizado en la subetapa **BKign** (búsqueda del valor clave de ignición **Kign**) para calcular el patrón de comportamiento del incendio. El valor hallado, **Kign**, es utilizado para realizar la predicción para el próximo instante de tiempo (t_{i+1}). La evaluación de la aptitud del mapa de probabilidad se lleva a cabo en la etapa que implementa la Función de Aptitud (**FA**).

Además, la salida **BKign** se combina con el mapa de probabilidad proporcionado por **EE** para generar la predicción (**LFP**) en la Etapa de Predicción (**EP**).

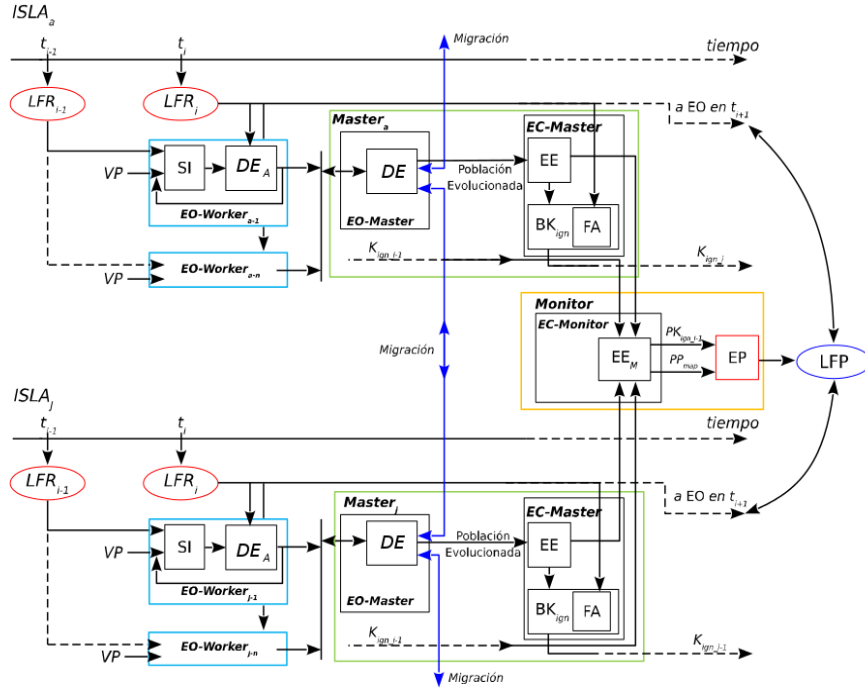


Fig. 2. Diagrama General de ESSIM-DE. **SI**: Simulador de Incendios, **DE**: Evolución Diferencial; **DE_A**: Evolución Diferencial (evaluación de aptitud); **EO**: Etapa de Optimización; **EE**: Etapa Estadística; **BKign**: Búsqueda de Kign; **Kign**: valor clave de ignición; **FA**: Función de Aptitud; **EC**: Etapa de Calibración; **EP**: Etapa de Predicción; **LFP**: Línea de Fuego Propuesta; **LFR_x**: Línea de Fuego Real en el instante x; **VP**: Vector de Parámetros de entrada; **EE_M**: Etapa Estadística Monitor; **PKign**: Valor clave de ignición pre-calculado; **PP**: Mapa de Probabilidad pre-calculado.

En la etapa **EC-Master**, la salida de **EE** se envía a **EC-Monitor** junto con los j valores de **Kign** calculados por las j islas. En la Etapa **EC-Monitor** el nodo Monitor selecciona el mejor valor de **Kign** entre los obtenidos por cada isla. El valor hallado **Kign** es utilizado por el monitor para realizar la predicción para el próximo instante de tiempo (t_{i+1}).

4 Experimentación

Para analizar la calidad de predicción del método ESSIM-DE se han utilizado dos casos reales de estudio correspondientes a incendios controlados realizados en

diferentes terrenos ubicados en Portugal (precisamente en Serra de Lousã, Gestosa), todos pertenecientes al proyecto SPREAD [9]. Cada caso posee una determinada duración expresada en minutos, una pendiente en grados y una superficie específica en metros cuadrados. El período de duración total de cada incendio ha sido dividido en lapsos menores de tiempo denominados pasos de simulación. En la Tabla I puede observarse la información detallada para cada caso de prueba considerado.

4.1 Muestreo por Hipercubo Latino

Para realizar un ajuste estático sobre alguno de los parámetros de entrada se utilizó la técnica de Muestreo por Hipercubo Latino (Latin Hypercube Sampling – LHS) [10] sobre un subconjunto de parámetros de ESSIM-DE, los cuales determinan principalmente el desempeño del algoritmo evolutivo. LHS utiliza un esquema de muestreo estratificado, permitiendo mejorar la cobertura del espacio muestral. Los parámetros que se eligieron para aplicar esta técnica de muestreo fueron la probabilidad de cruzamiento ($0 < CR < 1$), el factor de mutación ($0 < F < 1$), el umbral de aptitud ($0.55 \leq U \leq 0.85$) y la cantidad de iteraciones evolutivas ($30 \leq I \leq 100$).

Tabla I: Dimensiones, pendiente, tiempos e incremento de cada caso de prueba

Caso	Ancho (mts)	Largo (mts)	Pendiente (grados)	Inicio (min.)	Fin (min.)	Incremento (min.)
A	89	109	21	2.0	14.0	2.0
B	60	90	6	2.0	10.0	2.0

Para cada parámetro se definieron los rangos entre los cuales se realizó el muestreo estratificado y se generaron 20 vectores con dicha técnica. Cada uno de estos vectores se utilizó como conjunto de parámetros de control para ESSIM-DE, y se realizaron 10 ejecuciones con diferentes semillas para cada mapa, con cada uno de los 20 vectores.

Los diferentes vectores de entrada, en conjunto con el resto de los parámetros del método, se aplicaron en los experimentos bajo un entorno de ejecución paralelo, utilizando una configuración de 3 islas y 7 workers por isla.

4.2 Función de Aptitud

La calidad de predicción tanto de ESS como de ESSIM-DE es evaluada mediante una función de aptitud basada en el índice Jaccard. Conceptualmente se divide el terreno en celdas cuadradas debido a que el simulador utiliza una aproximación basada en celdas. La siguiente ecuación muestra la expresión:

$$Aptitud = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

donde A representa el conjunto de celdas en el mapa real sin el subconjunto de celdas quemadas antes de iniciar las simulaciones, y B representa el conjunto de celdas en el mapa simulado sin el subconjunto de celdas quemadas antes de comenzar la simulación. La corrección en el conjunto de celdas mediante la eliminación del subconjunto de celdas quemadas se utiliza para evitar resultados sesgados.

El valor de la Ec. 1 se encuentra en el rango $[0,1]$, una aptitud de 1 corresponde a una predicción perfecta, puesto que la superficie predicha en tal caso sería igual a la superficie real. Una aptitud igual a cero, indica el máximo error.

4.3 Resultados Obtenidos

En esta sección se muestran los resultados obtenidos de las ejecuciones con las diferentes configuraciones generadas por el muestreo LHS definido en la sección 4.1. Para una mejor claridad en el análisis, se seleccionaron los mejores cinco resultados obtenidos de las 20 configuraciones. La tabla II muestra las cinco mejores configuraciones para cada mapa. Los gráficos de las Figura 3 y 4 muestran los resultados obtenidos para los mapas A y B respectivamente. Se puede observar en la Figura 3 que la configuración 3 es la que mejores resultados obtiene, logrando un valor de aptitud cercano a 0.85 en el paso de simulación 2.

Los resultados obtenidos para el caso B son muy interesantes desde nuestro punto de vista. Observando la Figura 4, se puede visualizar que todas las configuraciones obtienen una calidad de predicción similar en el paso de simulación 4 (alrededor de 0.82), y si bien en general obtienen una calidad alta en el paso de simulación 6, se aprecia un descenso en la aptitud obtenida en el paso de simulación 8. Por ejemplo, la configuración 5 obtiene los mejores resultados de aptitud para los dos primeros pasos de simulación, pero obtiene una aptitud baja en el último paso de simulación. Por su parte, se observa que los resultados obtenidos con la configuración 3 son intermedios entre todas las configuraciones: en el paso de simulación 6 se obtiene una aptitud cercana a las más altas obtenidas, y en el último paso de simulación se obtiene la mejor predicción respecto del resto de las configuraciones.

De ambas gráficas se puede observar que una calibración realizada con un espectro muestral reducido permite obtener mejoras en los diferentes pasos de simulación. También es posible notar que la calibración de los parámetros es dependiente del problema que se esté considerando. Es importante remarcar que en el caso de los métodos de predicción, no siempre es posible encontrar mejoras en todos los pasos de simulación y para todos los escenarios de quema. Encontrar una mejor predicción en algún paso de simulación significa haber encontrado una mejor delimitación de la zona que se verá posiblemente afectada por el avance del fuego, en el instante de tiempo siguiente. Obtener mejoras parciales puede ayudar a reducir el impacto del fuego, pudiendo obtener a corto plazo medidas preventivas antes y durante el avance del incendio.

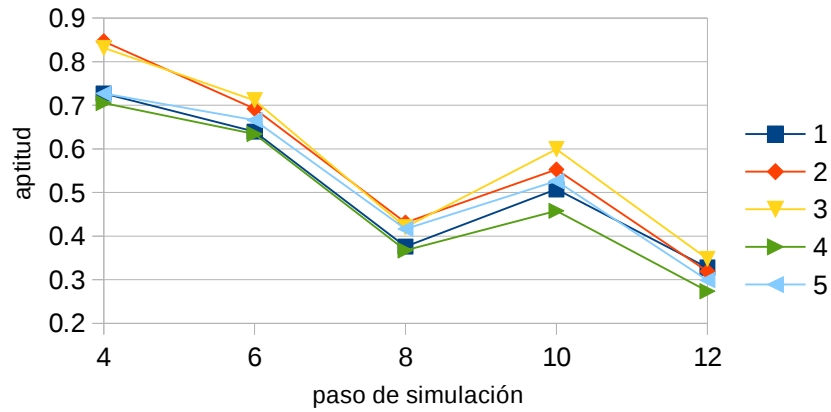


Fig. 3. Caso A: Resultados obtenidos con las 5 mejores combinaciones de parámetros.

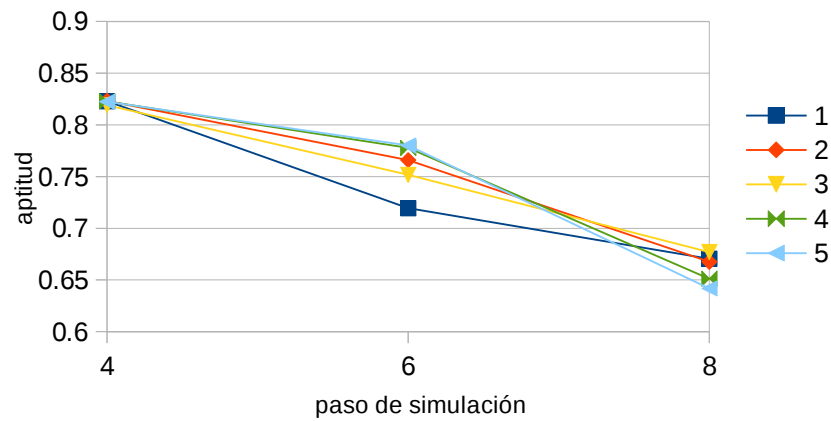


Fig. 4. Caso B: Resultados obtenidos con las 5 mejores combinaciones de parámetros.

Tabla II: Cinco mejores combinaciones de parámetros obtenidos en la calibración. Cr: probabilidad de cruzamiento, F: Factor de mutación, U: umbral de aptitud, I: cota de iteraciones en el ciclo evolutivo

Caso	A				B			
Nro	Cr	F	U	I	Cr	F	U	I
1	0.42	0.68	0.83	56	0.76	0.88	0.65	35
2	0.86	0.38	0.75	92	0.21	0.43	0.84	85
3	0.76	0.88	0.65	35	0.29	0.93	0.69	42
4	0.21	0.43	0.84	85	0.91	0.53	0.80	51
5	0.29	0.93	0.69	42	0.97	0.61	0.68	65

5 Conclusiones y trabajos futuros

En este trabajo se presentó una calibración estática de un subconjunto de parámetros del método ESSIM-DE, utilizando muestreo por Hipercubo Latino. Los resultados obtenidos evidencian que la calibración de los parámetros permite mejorar la calidad de las predicciones, lo que permite una mejor caracterización de la zona de riesgo a fin de reducir el impacto del avance del fuego. Como cada simulación representa un escenario de quema diferente, las características variables de cada problema (condiciones del terreno, del ambiente, morfología del foco de ignición) determinan escenarios muy dispares entre un mapa y otro. En el caso de los incendios forestales, se pudo corroborar que los valores de los parámetros obtenidos en la calibración varían de un mapa a otro. Por este motivo, se planea desarrollar como trabajo futuro un entorno de sintonización automática, de manera tal que la calibración se realice dinámicamente, con el objetivo de independizar la configuración de los parámetros de control relativos a la simulación respecto de las características inherentes al entorno de desarrollo del incendio.

Referencias

1. Finney, M.A: FARSITE: Fire Area Simulator-model development and evaluation. Res. Pap. RMRS-RP-4, Ogden, UT: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, (1998)
2. ADAI Products: FIRESTATION <http://www.adai.pt/products/firestation/>
3. Bianchini, G., Cortés, A., Margalef, T., Luque, E.: S²F²M-Statistical System for Forest Fire Management. LNCS, vol. 3514, pp. 427--434 (2005)
4. Bianchini, G., Caymes Scutari, P., Méndez Garabetti, M.: Evolutionary-Statistical System: A parallel method for improving forest fire spread prediction. Journal of Computational Science, vol. 6, pp. 58--66 (2015)
5. Méndez-Garabetti, M., Bianchini, G., Caymes-Scutari, P., Tardivo, M.L.: Increase in the quality of the prediction of a computational wildfire behavior method through the improvement of the internal metaheuristic. Fire Safety Journal, vol 82, pp. 49--62, (2016)
6. Price, K., Storn R., Lampinen J.: Differential Evolution: A Practical Approach to Global Optimization, Springer, New York, 2005.
7. Tardivo, M.L., Caymes Scutari, P., Méndez Garabetti, M., Bianchini, G.: Sintonización Estática de Parámetros Operacionales para un Modelo Paralelo de Evolución Diferencial. 1er Congreso Nacional de Ingeniería Informática y Sistemas de Información, pp. 770--779, ISSN: 2346-9927 (2013)
8. Tardivo, M.L., Caymes Scutari, P., Méndez Garabetti, M., Bianchini, G., Cortés, A.: Three Evolutionary Statistical Parallel Methods for Uncertainty Reduction in Wildland Fire Prediction. Proceedings of the 2016 International Conference on High Performance Computing & Simulation (IEEE HPCS 2016), Innsbruck, Austria, pp. 721--728 (2016)
9. Viegas, D.X. (Coordinator) et al.: Project Spread--Forest Fire Spread Prevention and Mitigation, <http://www.algosystems.gr/spread/> (2004)
10. McKay, M. D., Beckman, R. J., Conover, W.J.: A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code. Technometrics, vol 21, pp. 239--246, (1979)

On the Application of Argument Accrual to Reasoning with Inconsistent Possibilistic Ontologies

Sergio Alejandro Gómez

Artificial Intelligence Research and Development Laboratory (LIDIA)
Department of Computer Science and Engineering
Universidad Nacional del Sur
Av. Alem 1253, (8000) Bahía Blanca, ARGENTINA
Email: `sag@cs.uns.edu.ar`

Abstract. We present an approach for performing instance checking in a suitable subset of possibilistic description logic programming ontologies by using argument accrual. Ontologies are interpreted in possibilistic logic programming under Dung’s grounded semantics. We present a reasoning framework with a case study and a Java-based implementation for enacting the proposed approach.

Keywords: Argument accrual, ontology reasoning, inconsistency handling, Description Logics.

1 Introduction

Reasoning with inconsistent ontologies plays a fundamental role in Semantic Web applications. An ontology defines axiomatically a set of concepts that, given assertions about the membership of individuals to some concepts, allows to determine the membership of individuals to concepts (a task known as instance checking). Ontologies can suffer from incoherence and inconsistency; either correcting these anomalies or dealing with them with non-monotonic reasoning techniques are the two main accepted solutions [1]. Argumentation [2] is an approach to defeasible reasoning that can be applied to handling inconsistency in ontologies. In argumentation, given an inconsistent knowledge base, arguments compete to decide which are the accepted consequences. Argument accrual formalizes the notion that having more arguments for a certain conclusion makes it more credible [3].

In this paper we explore the application of argument accrual to reasoning with inconsistent ontologies. To the best of our knowledge, argument accrual were firstly studied by [4] and its application to the problem of ontology reasoning has only been suggested by Groza [5] in the context of fuzzy description logics. We use possibilistic description logic programming as the language for ontology representation (i.e. an ontology is ultimately interpreted as a possibilistic logic program). Arguments are then computed and accrued to compute the membership of instances to concepts using structured argumentation under Dung’s

grounded semantics. Our approach is qualitative providing a case study to show how our approach works, and a downloadable Java-based implementation for enacting our results.

The rest of the paper is structured as follows. In Sect. 2 we recall the fundamentals of possibilistic description logic ontologies. In Sect. 3 we review how argumentation under Dung’s semantics with arguments expressed in possibilistic logic programming is achieved and how possibilistic description logic ontologies can be interpreted as possibilistic logic programs. In Sect. 4 we introduce how to reason with inconsistent possibilistic ontologies using argument accrual. We conclude in Sect. 5.

2 Possibilistic Description Logic Ontologies

Classical Description Logics. *Description Logics* (DL) are a well-known family of knowledge representation formalisms [6]. In this work, we will consider a very tight subset \mathcal{L}_{DL} of DLs, to which we will restrict our discussion, based on the notions of *concepts* (unary predicates, classes). Concept descriptions are built from concept names C, D, \dots using the constructors conjunction ($C \sqcap D$), disjunction ($C \sqcup D$) and negation ($\neg C$). The empty concept is denoted by \perp . A DL ontology $\Sigma = (T, A)$ consists of two finite and mutually disjoint sets: a *Tbox* T which introduces the *terminology* and an *Abox* A (assertional box) which contains facts about particular objects in the application domain. The Tbox contains inclusion axioms $C \sqsubseteq D$, where C and D are (possibly complex) concept descriptions, meaning that every individual of C is also a D . Objects in the Abox are referred to by a finite number of *individual names* and these names may be used in assertional statements $a : C$ (meaning the individual a is a member of concept C).

Description Logic Programming. In this work we are only interested in the reasoning task known as *instance checking* that refers to determining if an individual a is a member of a concept C . Assigning semantics to a DL ontology can be done based on that DL is isomorphic with first-order logic restricted to two variables. Then, $C \sqsubseteq D$ can be interpreted as the formula $(\forall x)(c(x) \rightarrow d(x))$ and $a : C$ as $c(a)$. Description Logic Programming (DLP) approaches [7] take advantage of this to interpret those expressions as the Prolog rules “ $d(X) :- c(X)$.” and “ $c(a)$.”, resp. Therefore instance checking of a is a member of C reduces to proving the goal “ $\leftarrow c(a)$ ”.

Possibilistic Description Logics. We now recall the fundamentals of possibilistic description logic ontologies [8, 9]. A *possibilistic DL ontology* is a set of possibilistic axioms of the form $(\varphi, W(\varphi))$ where φ is an axiom expressed in \mathcal{L}_{DL} and $W(\varphi) \in [0, 1]$ is the degree of certainty (or priority) of φ . Namely, a possibilistic DL ontology Σ is such that $\Sigma = \{(\varphi_i, W(\varphi_i)) : i = 1, \dots, n\}$. Only somewhat certain information is explicitly represented in a possibilistic ontology. That is, axioms with a null weight ($W(\varphi) = 0$) are not explicitly represented

in the knowledge base. The weighted axiom $(\varphi, W(\varphi))$ means that the certainty degree of φ is at least equal to $W(\varphi)$. A possibilistic DL ontology Σ will also be represented by a pair $\Sigma = (T, A)$ where elements in both T and A may be uncertain. Note that if we consider all $W(\varphi_i) = 1$, then we find a classical DL ontology $\Sigma^* = \{\varphi_i : (\varphi_i, W(\varphi_i)) \in \Sigma\}$. We say that Σ is consistent if the classical ontology obtained from Σ by ignoring the weights associated with axioms is consistent, and inconsistent otherwise. Notice that the weights $W(\cdot)$ for axioms must be provided by the knowledge engineer that designs the knowledge base, thus providing the relative importance of axioms/rules and assertions/facts.

Example 1. (Originally presented in [10].) Let $\Sigma_1 = (T, A)$ be the ontology modeling a variation of the famous Tweety example from the non-monotonic literature. It expresses that a bird usually flies, all penguins are birds, penguins do not usually fly, birds with broken wings normally do not fly either, and pilots can almost always fly. It is known that Tweety is a penguin with almost certainly a broken wing and most likely a pilot. Formally:

$$T = \left\{ \begin{array}{lll} (\text{Bird} \sqsubseteq \text{Flies}, 0.6), & (\text{Penguin} \sqsubseteq \text{Bird}, 1.0), & (\text{Pilot} \sqsubseteq \text{Flies}, 0.9) \\ (\text{Penguin} \sqsubseteq \neg\text{Flies}, 0.8), & (\text{Bird} \sqcap \text{BrokenWing} \sqsubseteq \neg\text{Flies}, 0.7), & \end{array} \right\}$$

$$A = \left\{ \begin{array}{l} (\text{TWEETY} : \text{BrokenWing}, 0.8), \quad (\text{TWEETY} : \text{Penguin}, 1.0), \\ (\text{TWEETY} : \text{Pilot}, 0.9) \end{array} \right\}$$

In Σ_1^* , because Tweety a penguin (and therefore a bird), he is both a member of Flies and $\neg\text{Flies}$, meaning that he is a member of \perp . Traditional reasoners are not able to infer anything from such an inconsistent ontology, thus invalidating even reasoning with consistent subsets of the offending ontology.

3 Ontology Reasoning in Possibilistic Argumentation à la Dung

We now recall how to reason with possibly inconsistent ontologies by using Dung-style argumentation, an approach we originally explored in [10].

3.1 Notions of Possibilistic Defeasible Logic Programming

The P-DeLP [11] language \mathcal{L} is defined from a set of ground fuzzy atoms (fuzzy propositional variables) p, q, \dots along with the connectives \sim (strong negation), \wedge (written as a comma in Prolog clauses) and \leftarrow . A literal $L \in \mathcal{L}$ is a ground (fuzzy) atom $\sim q$, where q is a ground (fuzzy) propositional variable. A *rule* in \mathcal{L} is a formula of the form $Q \leftarrow L_1 \wedge \dots \wedge L_n$, where Q, L_1, \dots, L_n are literals in \mathcal{L} . When $n = 0$, the formula $Q \leftarrow$ is called a *fact*. The term *goal* will refer to any literal $Q \in \mathcal{L}$. Facts, rules and goals are the well-formed formulas in \mathcal{L} . A *certainty-weighted clause*, or simply weighted clause, is a pair (φ, α) , where φ is a formula in \mathcal{L} and $\alpha \in [0, 1]$ expresses a lower bound for the certainty of φ in terms of a necessity measure.

The proof method for P-DeLP formulas, noted \vdash , is based on the generalized modus ponens rule, that from $(P \leftarrow Q_1, \dots, Q_k, \gamma)$ and $(Q_1, \beta_1), \dots, (Q_k, \beta_k)$ allows to infer $(P, \min(\gamma, \beta_1, \dots, \beta_k))$. A clause (φ, α) is referred to as certain when $\alpha = 1$ and uncertain otherwise. A set of clauses Γ is deemed as *contradictory*, denoted $\Gamma \vdash \perp$, when $\Gamma \vdash (Q, \alpha)$ and $\Gamma \vdash (\sim Q, \beta)$, with $\alpha > 0$ and $\beta > 0$, for some atom Q in \mathcal{L} . A program is a set of weighted rules and facts in \mathcal{L} in which certain and uncertain information is distinguished. As an additional requirement, certain knowledge is required to be non-contradictory. A *P-DeLP program* \mathcal{P} (or just *program* \mathcal{P}) is a pair (Π, Δ) , where Π is a non-contradictory finite set of certain clauses, and Δ is a finite set of uncertain clauses. We build arguments \mathcal{A} for Q with weight γ , noted as $\langle \mathcal{A}, Q, \gamma \rangle$, inductively: If (Q, γ) is a fact, then $\langle \{(Q, \gamma)\}, Q, \gamma \rangle$ is an argument; from $(P \leftarrow Q_1, \dots, Q_n, \gamma)$ and n arguments $\langle \mathcal{A}_1, Q_1, \beta_1 \rangle, \dots, \langle \mathcal{A}_n, Q_n, \beta_n \rangle$, when Q_1, \dots, Q_n, P is consistent, then we get $\langle \bigcup_{i=1}^n \mathcal{A}_i \cup \{(P \leftarrow Q_1, \dots, Q_n, \gamma)\}, P, \min(\beta_1, \dots, \beta_n, \gamma) \rangle$.

Conflict among arguments is formalized by the notions of counterargument and defeat. Let \mathcal{P} be a program, and let $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$ and $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$ be two arguments in \mathcal{P} . We say that $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$ *counterargues* $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$ iff there exists a subargument (called *disagreement subargument*) $\langle \mathcal{S}, Q, \beta \rangle$ of $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$ such that $\Pi \cup \{(Q_1, \alpha_1), (Q, \beta)\}$ is contradictory. The literal (Q, β) is called *disagreement literal*. Defeat among arguments involves the consideration of *preference criteria* defined on the set of arguments. The criterion applied here is based on necessity measures associated with arguments. Let \mathcal{P} be a program, and let $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$ and $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$ be two arguments in \mathcal{P} . We will say that $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$ is a *defeater* for $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$ iff $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$ counterargues argument $\langle \mathcal{A}_2, Q_2, \alpha_2 \rangle$ with disagreement subargument $\langle \mathcal{A}, Q, \alpha \rangle$, with $\alpha_1 \geq \alpha$. If $\alpha_1 > \alpha$ then $\langle \mathcal{A}_1, Q_1, \alpha_1 \rangle$ is called a *proper defeater*, otherwise ($\alpha_1 = \alpha$) it is called a *blocking defeater*. Notice that we digress from the original P-DeLP formalism of Chesñevar et al. [11] in that (i) we include facts in the support of arguments and (ii) facts are allowed to have a weight different than one (so allowing them to be considered as presumptions).

Example 2. Consider again the ontology Σ_1 presented in Ex. 1. This ontology is interpreted as the equivalent possibilistic program \mathcal{P}_1 where:

$$\mathcal{P}_1 = \left\{ \begin{array}{l} (brokenWing(tweety), 0.8), \quad (penguin(tweety), 1.0), \\ (pilot(tweety), 0.9), \\ (flies(X) \leftarrow bird(X), 0.6), \\ (bird(X) \leftarrow penguin(X), 1.0), \\ (\sim flies(X) \leftarrow penguin(X), 0.8), \\ (\sim flies(X) \leftarrow bird(X), brokenWing(X), 0.7), \\ (flies(X) \leftarrow pilot(X), 0.9) \end{array} \right\}.$$

Exactly 8 arguments can be built from this program (notice that they coincide with the ones presented in Ex. 3):

- $\langle \mathcal{A}_1, brokenWing(tweety), 0.8 \rangle$ where $\mathcal{A}_1 = \{ (brokenWing(tweety), 0.8) \}$
- $\langle \mathcal{A}_2, penguin(tweety), 1.0 \rangle$ where $\mathcal{A}_2 = \{ (penguin(tweety), 1.0) \}$
- $\langle \mathcal{A}_3, pilot(tweety), 0.9 \rangle$ where $\mathcal{A}_3 = \{ (pilot(tweety), 0.9) \}$

- $\langle \mathcal{A}_4, bird(tweety), 1.0 \rangle$ where $\mathcal{A}_4 = \{ (bird(tweety) \leftarrow penguin(tweety), 1.0) \} \cup \mathcal{A}_2$
- $\langle \mathcal{A}_5, \sim flies(tweety), 0.8 \rangle$ where

$$\mathcal{A}_5 = \{ (\sim flies(tweety) \leftarrow penguin(tweety), 0.8) \} \cup \mathcal{A}_2$$

- $\langle \mathcal{A}_6, \sim flies(tweety), 0.7 \rangle$ where

$$\mathcal{A}_6 = \{ (\sim flies(tweety) \leftarrow bird(tweety), brokenWing(tweety), 0.7) \} \cup \mathcal{A}_1 \cup \mathcal{A}_4$$

- $\langle \mathcal{A}_7, flies(tweety), 0.9 \rangle$ where $\mathcal{A}_7 = \{ (flies(tweety) \leftarrow pilot(tweety), 0.9) \} \cup \mathcal{A}_3$
- $\langle \mathcal{A}_8, flies(tweety), 0.6 \rangle$ where $\mathcal{A}_8 = \{ (flies(tweety) \leftarrow bird(tweety), 0.6) \} \cup \mathcal{A}_4$

The attacks among these arguments are exactly those presented in Fig. 1. Notice that here the attacks are made into final conclusions (thus they are direct attacks). Nonetheless the reasoning framework presented here and the application we built also allow modeling attacks into premises (i.e. indirect attacks).

3.2 Dung-Style Abstract Argumentation

Abstract argumentation frameworks [12] do not presuppose any internal structure of arguments, thus considering only the interactions of arguments by means of an attack relation between arguments. An *abstract argumentation framework* \mathcal{AF} is a pair (Arg, \rightarrow) where Arg is a set of arguments and \rightarrow is a relation of Arg into Arg . For two arguments \mathcal{A} and \mathcal{B} in Arg , the relation $\mathcal{A} \rightarrow \mathcal{B}$ means that the argument \mathcal{A} attacks the argument \mathcal{B} . Abstract argumentation frameworks can be concisely represented by directed graphs, where arguments are represented as nodes and edges model the attack relation.

Example 3. Consider the argumentation framework $\mathcal{AF}_3 = (Arg, \rightarrow)$ where $Arg = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_5, \mathcal{A}_6, \mathcal{A}_7, \mathcal{A}_8\}$ and $\rightarrow = \{(\mathcal{A}_5, \mathcal{A}_8), (\mathcal{A}_6, \mathcal{A}_8), (\mathcal{A}_7, \mathcal{A}_5), (\mathcal{A}_7, \mathcal{A}_6)\}$. The framework is shown graphically in Fig. 1 and, although it is not necessary from a mathematical viewpoint, we can assign meaning to the above arguments to provide some intuition (notice that these arguments coincide with the ones in Ex. 2):

- \mathcal{A}_1 : Tweety has a broken wing
- \mathcal{A}_2 : Tweety is a penguin
- \mathcal{A}_3 : Tweety is a pilot
- \mathcal{A}_4 : Tweety is a bird
- \mathcal{A}_5 : Tweety does not fly because he is a penguin and penguins do not usually fly
- \mathcal{A}_6 : Tweety does not fly because he has a broken wing
- \mathcal{A}_7 : Tweety flies because he is also a pilot
- \mathcal{A}_8 : Tweety flies because he is a bird and birds normally fly

Semantics are usually given to abstract argumentation frameworks by means of extensions. An extension E of an argumentation framework $\mathcal{AF} = (Arg, \rightarrow)$ is subset of Arg that gives some coherent view on the argumentation underlying \mathcal{AF} . In this work, we will reason under grounded semantics despite that other semantics have been proposed. Let $\mathcal{AF} = (Arg, \rightarrow)$ be an argumentation framework. An extension $E \subseteq Arg$ is *conflict-free* iff there are no $\mathcal{A}, \mathcal{B} \in E$

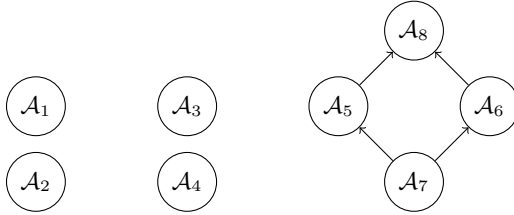


Fig. 1. Abstract argumentation framework presented in Ex. 3

with $\mathcal{A} \rightarrow \mathcal{B}$. An argument $\mathcal{A} \in Arg$ is *acceptable* with respect to an extension $E \subseteq Arg$ iff for every $\mathcal{B} \in Arg$ with $\mathcal{B} \rightarrow \mathcal{A}$ there is a $\mathcal{A}' \in E$ with $\mathcal{A}' \rightarrow \mathcal{B}$. An extension $E \subseteq Arg$ is *admissible* iff it is conflict free and all $\mathcal{A} \in E$ are acceptable with respect to E . An extension $E \subseteq Arg$ is *complete* iff it is admissible and there is no $\mathcal{A} \in Arg \setminus E$ that is acceptable with respect to E . An extension $E \subseteq Arg$ is *grounded* iff it is complete and E is minimal with respect to set inclusion.

The intuition behind admissibility is that an argument can only be accepted if there are no attackers that are accepted and if an argument is not accepted then there has to be an acceptable argument attacking it. The idea behind the completeness property is that all acceptable arguments should be accepted. The grounded extension is the minimal set of acceptable arguments and uniquely determined. It can be computed as follows: first, all arguments that have no attackers are added to the empty extension E and those arguments and all arguments that are attacked by one of these arguments are removed from the framework; then the process is repeated; if one obtains a framework where there is no unattacked arguments, the remaining arguments are also removed.

Example 4. Consider again the argumentation framework \mathcal{AF}_3 presented in Ex. 3. The grounded extension E of \mathcal{AF}_3 is given by $E = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_7, \mathcal{A}_8\}$.

3.3 Expressing Possibilistic DL Ontologies as Possibilistic Logic Programs

Grosz et al. [7] provide a way of expressing a subset of Description Logic ontologies in logic programming, namely the description logic programming subset of DL that can be expressed as Horn knowledge bases. The idea consist of expressing both DL assertional statements and terminological axioms as equivalent Horn clauses. We will explain only the part of the algorithm relevant to this work.

Given an ontology $\Sigma = (T, A)$, for every terminological axiom or assertional statement $(\varphi, W(\phi))$ we will generate a possibilistic axiom $(\mathcal{T}(\varphi), W(\varphi))$, where $\mathcal{T}(\cdot)$ is the transformation function from the language of description logics to the language of Horn clauses. The specification of the \mathcal{T} is as follows: Assertional statements in A of the form $a : C$ are expressed as facts $c(a)$. We obtain an equivalent ontology composed only of inclusion axioms of the form $C_1 \sqcap \dots \sqcap C_n \sqsubseteq D$ which are expressed as Horn clauses of the form $d(X) \leftarrow c_1(X), \dots, c_n(X)$.

Given a possibilistic DL ontology Σ , Σ is expressed as an equivalent P-DeLP program \mathcal{P} . With this program, a grounded extension E will be computed. If $(c(a), \alpha)$ belongs to E then we will say that the individual A is a member of the concept C with certainty degree α .

Example 5. Recall from Ex. 4 that $E = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_7, \mathcal{A}_8\}$. Therefore we can affirm that, from \mathcal{A}_1 , TWEETY is a member of BrokenWing with certainty degree 0.8; from \mathcal{A}_2 , TWEETY is a member of Penguin with certainty degree 1.0; from \mathcal{A}_3 , TWEETY is a member of Pilot with certainty degree 0.9; from \mathcal{A}_4 , TWEETY is a member of Bird with certainty degree 1.0; from \mathcal{A}_7 , TWEETY is a member of Flies with certainty degree 0.9, and from \mathcal{A}_8 , TWEETY is a member of Flies with certainty degree 0.6.

From \mathcal{A}_7 and \mathcal{A}_8 , as Tweety is both a member of Flies with both degrees 0.6 and 0.9, we take a credulous approach considering that TWEETY is a member with degree 0.9.

4 Ontology Reasoning with Argument Accrual

Now we deal with the problem of accruing arguments for justifying the membership of individuals to concepts, thus redefining the task of instance checking by means of argument accrual. Our approach relies in previous work of Gómez Lucero et al. by presenting a variation of their approach that we apply to ontology reasoning. Gómez Lucero et al. [3] presented an approach to model accrual of arguments in a possibilistic setting where, given different arguments supporting the same conclusion, they are able to accumulate their strength in terms of possibilistic values. For this, they define the notion of *accrued structure* whose necessity degree is computed in terms of two mutually recursive functions: $f_{\Phi}^+(\cdot)$ (the accruing function) and $f_{\Phi}^{MP}(\cdot)$ (that propagates necessity degrees). The latter is parameterized w.r.t. a user-defined function ACC that supports *non-depreciation* (i.e. accruing arguments results in a necessity degree no lower than any single argument involved in the accrual) and *maximality* (i.e. accrual means total certainty only if there is an argument with necessity degree 1). We recall the notion of *argument accrual* as interpreted by [3]:

Definition 1. Let \mathcal{P} be a P-DeLP program and let Ω be a set of arguments in \mathcal{P} supporting the same conclusion H , i.e. $\Omega = \{\langle \mathcal{A}_1, H, \alpha_1 \rangle, \dots, \langle \mathcal{A}_n, H, \alpha_n \rangle\}$. The accrued structure \mathcal{AS}_H for H is a 3-uple $[\Phi, H, \alpha]$, where $\Phi = \mathcal{A}_1 \cup \dots \cup \mathcal{A}_n$ and α is obtained as follows. Let Q be a literal in Φ and let $(\varphi_1, \beta_1), \dots, (\varphi_n, \beta_n)$ be all the weighted clauses in Φ with head Q , then

$$f_{\Phi}^+(Q) = ACC(f_{\Phi}^{MP}(\varphi_1), \dots, f_{\Phi}^{MP}(\varphi_n)).$$

Let (φ, β) be a weighted clause in Φ , then

$$f_{\Phi}^{MP}(\varphi) = \begin{cases} \beta & \text{if } \varphi \text{ is a fact } Q \\ \min(f_{\Phi}^+(P_1), \dots, f_{\Phi}^+(P_n)) & \text{if } \varphi = Q \leftarrow P_1, \dots, P_n. \end{cases}$$

And ACC is the one-complement accrual: $ACC(\alpha_1, \dots, \alpha_n) = 1 - \prod_{i=1}^n (1 - \alpha_i)$.

Given $[\Phi, H, \alpha]$ and $[\Theta, K, \gamma]$, $[\Theta, K, \gamma]$ is an accrued substructure if $\Theta \subseteq \Phi$. Also $[\Theta, K, \gamma]$ is a complete accrued substructure of $[\Phi, H, \alpha]$ iff for any other accrued substructure $[\Theta', K, \gamma']$ of $[\Phi, H, \alpha]$ it holds that $\Theta' \subset \Theta$. We say $[\Psi, K, \beta]$ attacks $[\Phi, H, \alpha]$ at literal H' iff there is a complete accrued substructure $[\Phi', H', \alpha']$ of $[\Phi, H, \alpha]$ such that $K = \overline{H'}$ and $\beta > \alpha'$.¹

Definition 2. Let \mathcal{P} be a possibilistic logic program. Let $\text{Accruals}(\mathcal{P})$ be the set of complete accrued structures of \mathcal{P} . Let $\mathcal{ASF}(\mathcal{P}) = (\text{Accruals}(\mathcal{P}), \text{attacks})$ be the argumentation framework induced by the accruals of \mathcal{P} where $\text{attacks} \subseteq \text{Accruals}(\mathcal{P}) \times \text{Accruals}(\mathcal{P})$. The extension of \mathcal{P} is defined as the grounded extension of $\mathcal{ASF}(\mathcal{P})$ where attacks stands for the attack relation between complete accrued structures.

Notice that the notions of both attack and valid conclusions of the system presented here differ from those of [3], thus leading to a different behavior.

Example 6. Consider again the ontology Σ_1 and its interpretation as the possibilistic program \mathcal{P}_1 . The following complete accrued structures can be computed from \mathcal{P}_1 : $\mathcal{AS}_1 = [\mathcal{A}_2, \text{penguin}(\text{tweety}), 1.0]$, $\mathcal{AS}_2 = [\mathcal{A}_7 \cup \mathcal{A}_8, \text{flies}(\text{tweety}), 0.96]$, $\mathcal{AS}_3 = [\mathcal{A}_3, \text{pilot}(\text{tweety}), 0.9]$, $\mathcal{AS}_4 = [\mathcal{A}_1, \text{brokenWing}(\text{tweety}), 0.8]$, $\mathcal{AS}_5 = [\mathcal{A}_5 \cup \mathcal{A}_6, \sim \text{flies}(\text{tweety}), 0.94]$ and $\mathcal{AS}_6 = [\mathcal{A}_4, \text{bird}(\text{tweety}), 1.0]$. We show \mathcal{AS}_2 and \mathcal{AS}_5 in Fig. 2. In this case \mathcal{AS}_2 attacks \mathcal{AS}_5 at the literal $\text{flies}(\text{tweety})$ (see Fig. 3). Therefore the grounded extension of $\mathcal{ASF}(\mathcal{P}_1)$ is $\{\mathcal{AS}_1, \mathcal{AS}_2, \mathcal{AS}_3, \mathcal{AS}_4, \mathcal{AS}_6\}$.

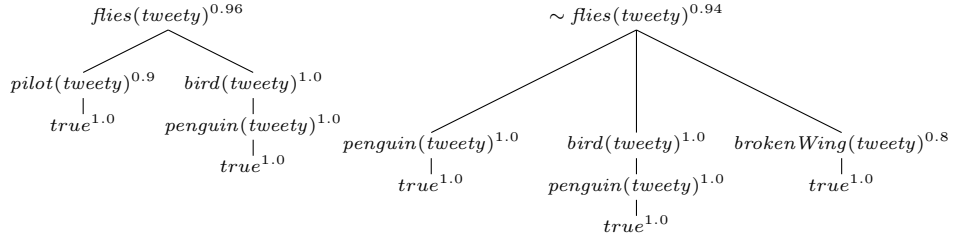


Fig. 2. Accrued structures \mathcal{AS}_2 and \mathcal{AS}_5 from \mathcal{P}_1 (Notice that the leftmost branch of \mathcal{AS}_5 stands for a sub-structure for $\sim \text{flies}(\text{tweety})$ based on the argument \mathcal{A}_5 while the two rightmost branches for another based on the argument \mathcal{A}_6 .)

Definition 3. Given a possibilistic ontology Σ , a concept C and an individual a , we say that a is member of C with certainty degree α iff there is a complete accrued structure for $[\Phi, c(a), \alpha]$ in the grounded extension of $\mathcal{ASF}(\mathcal{T}(\Sigma))$.

¹ $\bar{\cdot}$ is referred to as the complement operator where \overline{P} is $\sim P$ and $\overline{\sim P}$ is P .

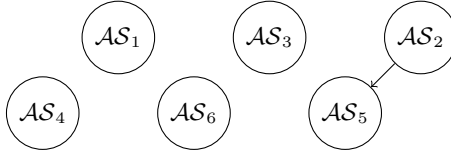


Fig. 3. Abstract argumentation framework $\mathcal{ASF}(\mathcal{P}_1)$ of Ex. 6

Example 7. Consider again the ontology Σ_1 , $\mathcal{T}(\Sigma_1)$ is the program \mathcal{P}_1 . The complete accrued structures from this program are those presented in Ex. 6. As $\mathcal{AS}_2 = [\mathcal{A}_7 \cup \mathcal{A}_8, \text{flies}(\text{tweety}), 0.96]$ belongs to the grounded extension of accrued structures of \mathcal{P}_1 , then we conclude TWEETY is a member of Flies with certainty degree 0.96.

Property 1. When each accrued structure is formed by exactly one argument, the grounded extension of the argumentation framework induced by the program coincides with the argumentation framework induced from the accrued program.

Proof: Let Σ be a possibilistic ontology and \mathcal{P} be $\mathcal{T}(\Sigma)$. Suppose that there is only one argument \mathcal{A} in \mathcal{P} with certainty degree α supporting H . Then the accrued structure \mathcal{AS} for H is formed only by \mathcal{A} . Because of how $f_{\Phi}^+(H)$ is defined, the certainty degree of \mathcal{AS} is also α . Then the graph of the argumentation framework formed by arguments is identical to the argumentation framework formed by accrued structures. Therefore their grounded extensions coincide.

Implementation details. In order to enact the approach proposed in this work, we developed a Java-based implementation that extends the one already presented in [10]. This application can be downloaded from the author’s institutional site at <http://cs.uns.edu.ar/~sag/engine-v2/>. The implementation also provides the functionality of loading several examples, editing them, obtaining the equivalent P-DeLP program, computing grounded extensions and also provides a graphical representation of the argumentation framework, individual arguments and accrued structures based on the D3.JS and Dracula Javascript libraries. With this, we were able to replicate the accrued structures presented by [3]. Based on the experience gained by testing examples, we argue that this approach presents a more complex way to compute the vertexes of the argumentation framework but the resulting graph is smaller, thus leading to a much simpler reasoning framework.

5 Conclusions and Future Work

We have presented an approach for performing instance checking in inconsistent possibilistic description logic ontologies based on argument accrual, leading to more concise representations. Our approach involves translating ontologies into the language of possibilistic logic programming and grouping arguments for the

same conclusion via argument accrual. We have developed a Java-based implementation that allows the user to input an ontology and select reasoning either with or without argument accrual under a grounded semantics based on Dung-style argumentation. In our implementation, we only implemented direct attack between pairs of accrued structures; implementing the full approach of [3] in the context of Dung-based structured argumentation requires further research.

Acknowledgments: This research is funded by Secretaría General de Ciencia y Técnica, Universidad Nacional del Sur, Argentina.

References

1. Zhang, X., Xiao, G., Lin, Z., den Bussche, J.V.: Inconsistency-tolerant reasoning with OWL-DL. *International Journal of Approximate Reasoning* **55** (2014) 557–584
2. Bench-Capon, T.J.M., Dunne, P.E.: Argumentation in artificial intelligence. *Artificial Intelligence* **171**(10-15) (2007) 619–641
3. Lucero, M.G., Chesñevar, C.I., Simari, G.R.: Modelling Argument Accrual in Possibilistic Defeasible Logic Programming. In ad G. Chemello, C.S., ed.: *ECSQARU 2009*, LNAI 5590. (2009) 131–143
4. Verheij, B.: Rules, Reasons, Arguments: Formal studies of argumentation and defeat. PhD thesis, University of Maastricht (1996)
5. Letia, I., Groza, A.: Modelling Imprecise Arguments in Description Logics. *Advances in Electrical and Computer Engineering* **9**(3) (2009) 94–99
6. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: *The Description Logic Handbook – Theory, Implementation and Applications*. Cambridge University Press (2003)
7. Grosz, B.N., Horrocks, I., Volz, R., Decker, S.: Description Logic Programs: Combining Logic Programs with Description Logics. *WWW2003*, May 20-24, Budapest, Hungary (2003)
8. Benferhat, S., Bouraoui, Z., Lagrue, S., Rossit, J.: Merging Incommensurable Possibilistic DL-Lite Assertional Bases. In Papini, O., Benferhat, S., Garcia, L., Mugnier, M.L., eds.: *Proceedings of the IJCAI Workshop 13 Ontologies and Logic Programming for Query Answering*. (2015) 90–95
9. Gómez, S.A., Chesñevar, C.I., Simari, G.R.: Using Possibilistic Defeasible Logic Programming for Reasoning with Inconsistent Ontologies. In Giusti, A.D., Diaz, J., eds.: *Computer Science & Technology Series. XVII Argentine Congress of Computer Science Selected Papers*. (2012) 19–29
10. Gómez, S.A.: Towards a practical implementation of a reasoner for inconsistent possibilistic description logic programming ontologies. In: *Proceedings of the 2do. Simposio Argentino de Ontologías y sus Aplicaciones (SAOA 2016)*. SADIO (2016) (accepted)
11. Alsinet, T., Chesñevar, C.I., Godo, L.: A level-based approach to computing warranted arguments in possibilistic defeasible logic programming. In Besnard, P., Doutre, S., Hunter, A., eds.: *COMMA*. Volume 172 of *Frontiers in Artificial Intelligence and Applications*., IOS Press (2008) 1–12
12. Dung, P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning and logic programming. In: *Proceedings of the 13th International Joint Conference in Artificial Intelligence (IJCAI)*. (1993) 852–857

Agent-based simulation of a project fractal company

Canavesio, Mercedes¹, Quaglia, Constanza¹, Martinez, Ernesto^{1,2},

¹UTN – Facultad Regional Santa Fe, Lavaisse 610, Santa Fe Argentina

²INGAR – UTN CONICET, Avellaneda 3500, Santa Fe, Argentina
mcanaves@frsf.utn.edu.ar, ecmarti@conicet-santafe.gob.ar

Abstract. In response to competitive pressures small and medium enterprises have begun to form alliances between them. This allows each company to increase their ability to react and adapt to changes in their business environment, concentrating on their core competencies, increase the availability of resources and gain economies of scale. Thus, the project-based fractal company model enables virtual and temporary integration between different companies to achieve specific business objectives. The successful implementation of the model lies in the establishment of client-server relationships between project managers. This paper analyses the use of agent-based simulation to show behaviors that emerge from the interaction of project managers and resources managers when they establish client-server relationships. To do this, use Netlogo language as a tool intends to demonstrate emergent behavior resulting from interactions between agents of the fractal company.

Keywords: Agent-based model and simulation. Fractal Company. Generative Modeling.

1 Introduction

Intense competition in the manufacturing and service sectors due to globalization, product customization, variations in demand patterns and rapid technological development demands new enterprise models. To survive, companies must increase product portfolio, reduce time-to-market, shorten product-life cycles and at the same time maintain good product quality and reduce investment cost. Competitive threats are much worse for small and medium enterprises (SMEs) which are reengineering their production and management systems to compete successfully. SMEs have to organize themselves in effective production networks to achieve a higher degree of flexibility, agility and low costs to cope with the increasing rate of change and complexity of a highly competitive environment. To address competitive threats and concentrate on their core competences and strengths networking is the alternative of choice for each individual SME survival and prosperity [1]. Objectives of a SMEs network include increased agility in responding to competitive threats, a more

comprehensive pool of skills and resources, economy of scale and product portfolio diversification.

To grasp all the benefits of virtual enterprise networking is mandatory to define an integrated company model to influence by design the behavior of each SME and relationships thereof. To solve this problem, [2] proposes an integrated company model revolving around the concept of a project-oriented fractal company for SMEs networking. In this model, the fractal management unit is modeled as a project. Each project is seen as an autonomous, temporary entity within the enterprise network, in which different types of expertise are combined to achieve a concrete goal (e.g., in product development, satisfying resource usages as scheduled, etc.). The underlying idea in the project-based fractal company is establishing client-server (temporal) relationships between project and resource managers in an open market economy.

During the interactions between managers to establish these relationships emerging client-server behaviors that cannot be predicted a priori are generated. Knowing these emergent behaviors is important in determining changes in policies and / or rules of interaction in negotiations between managers achieve significant economic benefits for the whole of the fractal company.

Analyzing the literature, it is possible to find papers where the use of agent-based simulation for analysis of complexity theory and its relationship with emerging behaviors in social systems, it is proposed given that they arise as a result of simultaneous interactions of multiple heterogeneous agents that make up the system and whose occurrence cannot be deduced a priori with only thorough analysis of the system in question. Thus, [3] proposes agent-based simulation as a methodology for research in the social sciences. [4] make an important description of the use of both simulation and system dynamics for the analysis of complex systems demonstrating the advantages of the first application by the results that can be observed. In addition, important contributions were made in the area of social organizations and complex systems in the use of agent-based simulation to demonstrate emergent behaviors [5, 6, 7].

In this work, generative modeling [8] will be used as a computational tool to analyze emergent properties of this new integrated enterprise model. This novel approach, based on agents and simulation, will allow an objective assessment of dynamic behaviors of a distributed-fractal management system modeled as an artificial society of autonomic agents.

The efficiency and effectiveness of the results obtained in achieving its goals will depend on the server selection strategy that every project manager performs.

During the interactions between managers to establish these client-server relationships, emerging behaviors that cannot be predicted a priori are simulated. Knowing these emergent behaviors is important in determining changes in policies and / or rules of negotiations between managers in order to achieve significant economic benefits for the fractal company as a whole.

This paper presents agent-based simulation as a tool for discovering emergent behaviours that arise from interactions between managers selecting partners with whom to establish more beneficial client-server relationships. To do this, it is proposed an agent based model using Netlogo language [9].

2 Project-based fractal company

The fractal company [10] is a conceptual enterprise model that aims to achieve a high degree of flexibility to react and adapt quickly to environmental changes using decentralized and autonomous organizational units known as fractals. A fractal is defined as a structure that describes an identical pattern that replicates itself at different abstraction levels in a recursive way. In the project-based fractal company, a fractal management unit is modeled as a project [2]. Thus, each project fractal is seen as an autonomous, self-optimizing, self-learning and goal-driven entity, in which different types of expertise are combined to achieve a concrete goal or deliverable (e.g. completing an order, discovery of a new drug for cancer treatment, introducing a new product, satisfying resource usages as scheduled, etc.). A fractal approach is used in order to allow enterprises to be able to self-adjust to the changes in the environment [2, 11, 12, 13]. Using projects at all abstraction levels is both a generator of efficiency and absorber of complexity to embed a highly adaptive and responsive organizational design that can balance each project internal complexity with environmental pressures on both the short- and long-term horizons. The other advantage of the proposed recursive structure is that naturally lends decentralized decision making [2].

This management fractal unit is made up of a project-manager and a managed object (Figure1). Each project-fractal unit is an autonomic unit which the project manager agent implements the monitor-analyze-plan-execute (MAPE) loop [14]. Thus, the project manager is able to: i) sense the present situation of the fractal unit, ii) monitor the managed object and its external environment in order to construct and execute plans based on an analysis of available information iii) learn to act based on experience and evaluative feedback of actions taken, and, iv) interact with other project managers. The project-based fractal model separates the management of goals from the management of the resources needed to obtain such goals but in both cases the fractal management unit is conceived as a project. Thus, the managed object may be either an ends or a means in the fractal company system. Furthermore, learning allows the accumulation of knowledge based on the actions executed by other project managers.

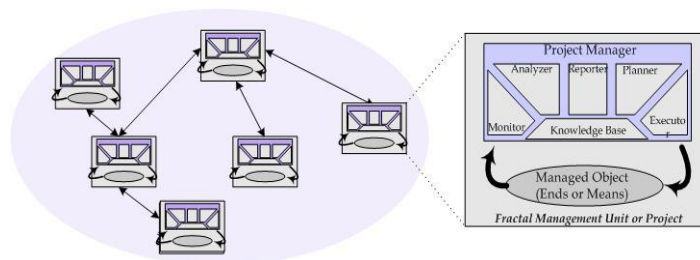


Fig. 1. Internal Structure of a project-fractal management unit

In each project, depending on the type of managed object (ends or means), its project manager plays the ends manager role or the means manager role, respectively. Project managers interact and communicate through client-server relationships established among them. In each relationship, the ends-manager is the consumer or client for a given resource and the means-manager is the supplier or server of the concerned resource. These relationships are established by free negotiation among selfish actors in the fractal company that are interested in contributing to achieve business goals and deliverables.

3 Generative simulation of the project-fractal company

3.1 Agent-based modeling and simulation

Agent-based modeling and simulation is a computational method that enables an analyst to create, analyze and experiment with simulation models composed of agents that interact within a well-defined environment [15]. One of the advantages of computational modeling is that it forces the modeler to be precise: unlike theories and models expressed in natural language, a computer simulation program has to be completely and exactly specified if it has to run on a computer. Another advantage is that in silicon experiments with a computational model is cheaper or occasionally is to test hypotheses regarding mechanism design. Furthermore, an experiment can be set up and repeated many times, using a range of parameters or allowing some factors to vary randomly.

Agent-based modeling and simulation is able to discover emergent connection between systems components, tie experience with detailed processes to system-level knowledge and archetype patterns, and in this way it may identify outcomes that are outside the range of analytical thinking [5]. To this aim, agent modeling must follow an iterative model construction interaction mechanism along with supporting data. This conceptual description is then converted to a generative model that can be used to test hypothesis. The resulting model is then simulated and the initial results are examined. The internal behavior definitions and de interaction mechanism design in the model are then updated based on system goals pursued, and the model is simulated again. This progressive refinement process continues until the model reproduces both the behaviors and results of the target system.

Generative models using agent-based objects are a very natural way for understanding and designing complex adaptive systems. Regardless of the presence of equilibrium, the path from micro-worlds to emergent properties (macro-behavior) requires systemic synthesis rather analytical tool [5].

3.2 Generative Modeling

Generative modeling [16] refers to the application of computational models to understand complex social systems. The usual barriers set by the standard modeling tools, such as the need to keep the model within reasonable size are overcome with the computational modeling approach. In computational models a controlled micro world simulation is used to analyze the macro level behaviors resulting from on-going interactions on micro scale worlds. The model serves as a tool to validate the hypothesis (simulated experimentation) and as a result to improve the proposed designs [5, 6]. This tool reveals emergent connections between the component systems and makes viable the identification of unexpected consequences of agent

interactions. Therefore it allows discovering and analyzing the social structures and the properties that emerge from a given mechanisms design.

Contrasting with the traditional top-down approach, a bottom-up emergence in an artificial society of agents which mainly communicate through the actions taken, define patterns that arise from the chosen mechanism design. Moreover, those aggregate patterns should be immune to reasonable variations in the individual agent behavior. Although the agents are simple, the result of its ongoing interaction can be very complex and difficult to predict.

When interactions are not independent, feedback regulation can play a substantial role. Feedback fundamentally alters the dynamics of a system. In a system with negative feedback, changes get quickly absorbed and the system gains stability. With positive feedback, changes are amplified leading to instability. Thinking about positive and negative feedback loops provides a rather shallow view into the micro motives for complex archetype behaviors.

Most of the existing analytic tools require that the underlying agents have a high degree of homogeneity. This homogeneity is not a feature often observed in the world but rather a necessity imposed by analytical modeling approach [5]. Unlike traditional tool, computational methods are able to incorporate heterogeneous agents easily. In this way, software agents managing the manufacturing settings (resources and products) can interact with each other and coordinate and plan their activities so that they can further their own design objectives. Essential in this process are the abilities to negotiate and reach deals with other agents. Thus the emergent behavior form interactions among heterogeneous agents with different objectives highlights the importance of resorting to the generative modeling approach of ABMS which allows designing meta rules that change the interaction rules in distributed decision-making.

3.3 Prototype implementation in Netlogo

A computational model of the Project-based fractal company world was implemented in Netlogo [9], a software environment specifically designed for generative modeling of artificial agent societies. The Netlogo world is made up of agents. Agents are entities that can follow instructions and develop a decision-making policy. Each agent carries out its own activity, in an asynchronous manner regarding other agents.

In Netlogo, there are four types of agents: turtles, patches, links and the observer. Turtles are agents that move around in the world. The world is two dimensional and is divided up into a grid of patches. Each patch is a square piece of “ground” over which turtles can move. Links are agents that connect two turtles. The observer does not have a location, it is constantly looking out over the world of turtles and patches.

For Project-based Fractal Company, turtles are both project and resource manager who negotiate a resource usage for a task involved in the project schedule. Patches represent resource usage times on schedule.

To program the model, global variables were used when the information was public (available for every agents), such as number of project managers, resources, projects, etc. Each turtle was defined with a group of attributes (preference, projects, etc.) and behaviors (selection partner algorithm). They have private information for a

single agent or type of agent, or when their values changes depending on the turtle or patch. Figure 2 shows a view the setting model.



Fig. 2. Initial simulation setting

3.3.1. Simulation Results

The Netlogo environment enables exploration of emergent phenomena. As simulation results an emergent Gantt chart of resource usage was obtained.

The first run of the model considers that agents both project managers and resources have established and fixed preferences for the selection of partners (Figure 3). The second run shows the Gantt chart emerging considering that agents incorporate an algorithm that allows over time their most beneficial partner. Agents have been provided with an algorithm reinforcement learning applied to bilateral markets such as the fractal company, where there are project managers who want to be associated with certain resource managers while the latter have preferences regarding managers projects to those willing to serve. So in each iteration there must be a match between a project manager and a resource manager set including a client-server relationship. A detailed description of the algorithm and its application is in [17].

Comparing emergent Gantt chart shows that when agents learn about their preferences hold a better use of resource time. Figure 3 shows the great idle resource times. This leads to lost currency for both agents. Project managers for delays in the due date of their projects and resource managers in idle resources cost.

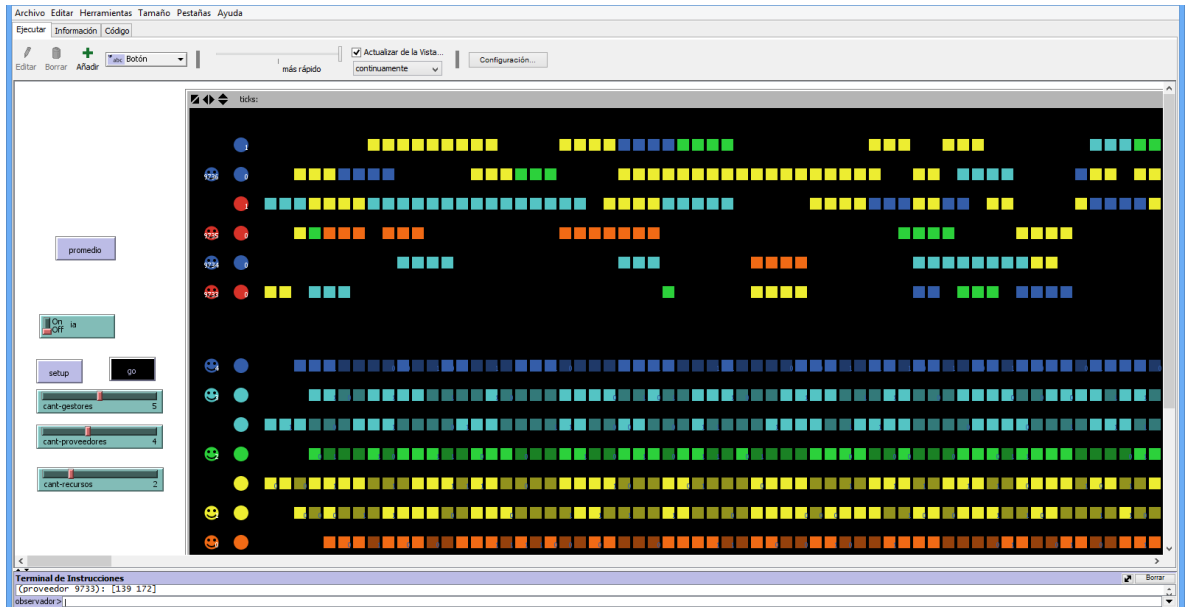


Fig. 3. Agents have fixed preferences for the selection of partners.

The following graphic behavior ratifies what said. While project managers chose their resources providers randomly, all at the same time they chose the *resource manager 1*, whereas when they learn about their potential partners this supplier is not chosen for any project manager. This indicates that this resource manager is a bad supplier Figure 4.

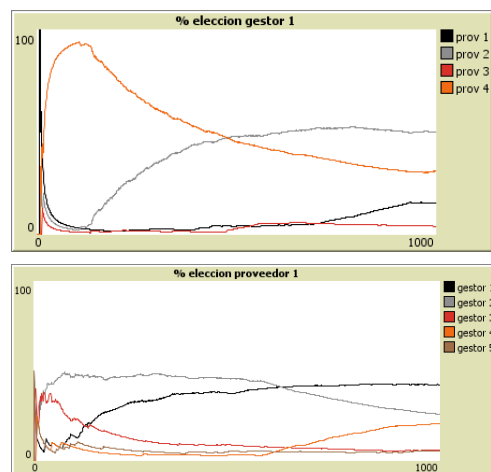


Fig. 4. Project Managers Behaviours.

4 Conclusions

The paper presents the generative simulation as a tool for analysis of complex adaptive systems, where the behaviours that emerge as a result of multiple interactions between agents cannot be predicted a priori with the simple analysis of them.

For this model the world of fractal-based company projects in order to analyze their behavior macros, as a result of interactions between project managers and resources to establish client-server relationships between them. The model also allowed to study patterns of behavior both when agents use a random selection as when they learn to choose their best partner for closing contract business. In either case without a simulation of the world, which can interact with hundreds of managers it is not possible to predict the behavior emerging macros. This knowledge will improve the mechanisms of interaction to achieve optimal behavior in the formation of virtual and temporary networks between companies.

References

1. Basole, R.C., Rouse,W.B., McGinnis, L.F., Bodner, D.A., Kessler, W.C. Models of Complex Enterprise Networks. *Journal of Enterprise Transformation.* 1:3, Pp208-230. (2011).
2. Canavesio, M.M.; Martinez, E.C. Enterprise modeling of a project-oriented fractal company for SMEs networking. *Computers in Industry.* Elsevier Nro 58, Pp 974-813 (2007)
3. Rodriguez Zoya,L., Aguirre,J.L. Teoría de la complejidad y ciencias sociales. Nuevas estrategias epistemológicas y metodológicas. *Revista crítica de Ciencias sociales y jurídicas.* Vol30. N 2 (2011)
4. Izquierdo, R.L, Galan, J.M., Santos, J.I., del Olmo, R. Modelado de sistemas complejos mediante simulación basada en agentes y mediante dinámica de sistemas. *EnPIRIA. Revista de Metodologías de Ciencias Sociales.* N 16. Pp 85-112 (2008)
5. Miller, J.H., Page, S.E. *Complex adaptive systems: An introduction to computational models of social life.* Princenton University Press. (2007)
6. Pavón Mestras,J., Lopez Paredes,A., Galan Ordax,J.M. Modelado basada en agentes para el estudio de sistemas complejos. *Novatico* 218. Pp 13-18. (2012)
7. Rolon, M, Canavesio, M, Martinez, E . (2009) Agent based modeling and simulation of intelligent distributed scheduling systems. *Computer aided chemical engineering.* Vol 26. Pp. 985-990
8. North, M. ; Macal, M. *Managing Business Complexity, Discovering Strategic Solution with Agent-based Modeling and Simulation,* Oxford: Osford University Press. (2007)
9. Wilesky,U., *Netlogo modeling environment.*(1999)
10. Warnecke, H..J. *The fractal company: a revolution in corporate culture.* Springer-Verlag . (2003)
11. Ramanathan, J. Fractal architecture for the adaptive complex enterprise. *Communications of the ACM.* Vol 48. No 5. Pp 51-57. (2005)
12. Kirikova, M., *Towards flexible information architecture for fractal information systems.* I International conference on information, process and knowledge management. IEEE Computer Society.(2009)

13. Bider,I., Perjons, E., Elias, M., Untangling the dynamic structure of an enterprise by applying a fractal approach to business processes. Proceedings of PoEM (2012)
14. Kephart, J., Chess, D., The vision of autonomic computing. Computer. Vol 36 Nro.1. IEEE Computer society. Pp 41-50. (2003)
15. Gilbert,N., Agent-based models. Guildford: Sage Publications. (2008)
16. Epstein,J.M., Generative social science: studies in agent-based computational modeling. Princeton University Press (2006)
17. Quaglia, C. Canavesio, M., Martinez, E., Agentes inteligentes aprenden a establecer relaciones entre socios en mercados bilaterales. Revista electrónica Argentina-Brasil de tecnologías de la información y la comunicación. Vol 1. Nro 2 (2014)

Clasificación Automática de Textos Periodísticos usando Random Forest.

Juan Salinas y Javier Izetta

Facultad de Ingeniería,
Universidad Nacional de Jujuy, Argentina
{juangsalinas90, javierizetta}@gmail.com
<http://www.fi.unju.edu.ar>

Resumen En los últimos años el periodismo regional, al igual que en todas partes del mundo, pasó de su formato clásico de publicación al electrónico. Así las *webs* de noticias regionales también se ven obligadas a evolucionar y mejorar sus prestaciones a través de una mejor organización y categorización previa de toda la información disponible para el lector. En este trabajo se propone abordar la clasificación automática de textos periodísticos digitales a través del *Aprendizaje Automatizado*. Se presentan dos clasificadores automáticos de textos periodísticos extraídos de páginas webs de noticias del NOA basados en *Random Forest* y se proponen dos técnicas para la reducción de dimensionalidad del espacio de características. Estos clasificadores fueron evaluados con distintas colecciones de noticias demostrando un buen desempeño.

Keywords: categorización automática de textos, random forest, stemming.

1. Introduction

La *Categorización Automática de Texto* (CAT) tiene por objetivo hallar una función óptima de clasificación de documentos a partir de atributos constituidos por palabras que describen cada categoría específica [1]. En la actualidad la CAT cuenta con un amplio campo de aplicación, entre los más destacables podemos encontrar: filtrado y organización del contenido de páginas web [2], detección de correos electrónicos *Spam* [3], filtrado de noticias [4], detección de plagios e identificación de autores [5] y análisis de opinión [6].

En este trabajo se aplica el enfoque de "bolsas de palabras" que es el más utilizado en CAT. En este enfoque se consideran las palabras individuales y su información léxica. Otro aspecto a tener en cuenta es que un texto podría pertenecer a varias, una o todas las categorías especificadas [7]. El presente trabajo se enfoca en la clasificación de única etiqueta, es decir, cada documento de la colección está asociado a solo una clase o categoría.

Actualmente la CAT puede tratarse como un problema de "Clasificación Supervisada". Los métodos computacionales desarrollados para tal fin forman parte de lo que se conoce como *Aprendizaje Automatizado* (AA). Diversos algoritmos

de AA, fueron utilizados para dar solución a la problemática de CAT. Particularmente varios trabajos proponen utilizar árboles de decisión para la clasificación de textos [8] [9] [10]. En este trabajo se propone utilizar el método *Random Forest* que utiliza un conjunto (o bosque) de árboles de decisión para predecir un valor de salida [11].

2. Clasificación de textos como un problema de AA

Desde el punto de vista del AA, la CAT se puede definir formalmente a partir de la siguiente situación:

Se tiene un conjunto de documentos $D = \{\mathbf{d}_1, \dots, \mathbf{d}_i, \dots, \mathbf{d}_n\}$, donde cada documento \mathbf{d}_i se expresa como $\mathbf{d}_i = \{t_1, \dots, t_j, \dots, t_m\} \in \mathfrak{R}^m$. \mathbf{d}_i es un vector en un espacio m-dimensional. \mathfrak{R}^m se denomina *feature space*, espacio de variables o espacio de características, indistintamente. Cada documento tiene asignada una clase o etiqueta real $l(\mathbf{d}_i)$ conocida de antemano. Para un problema con c clases, $l(\mathbf{d}_i)$ puede tomar c valores discretos distintos.

Entonces se tiene por objetivo hallar una función clasificadora \hat{f} tal que para cada documento \mathbf{d}_i se tiene $\hat{f}(\mathbf{d}_i) = l(\mathbf{d}_i)$, donde $\hat{f}(\mathbf{d}_i)$ es la etiqueta o clase asignada por \hat{f} al documento \mathbf{d}_i . Además se pretende que \hat{f} pueda clasificar correctamente nuevos documentos no contemplados anteriormente.

3. Random Forest

Se trata de un método desarrollado por Leo Breiman y Adele Cutler en 2001 [11]. Este método utiliza un conjunto (o bosque) de árboles de decisión para predecir un valor de salida. Para la clasificación de un nuevo ejemplo, cada árbol en el ensamble produce un voto, es decir, toma el ejemplo como entrada y produce una salida indicando a que clase pertenece. La decisión final del ensamble se realiza a partir de la clase con mayor cantidad de votos.

4. Construcción de los clasificadores propuestos

La construcción de los clasificadores se abordó a través de dos etapas claramente delimitadas. La primera etapa, a la que suele llamarse etapa de *entrenamiento*, se inicia con la recopilación manual de una serie de textos periodísticos de diarios digitales del noroeste argentino extraídos de la *web* (documentos de entrenamiento). Esta colección se procesa para lograr una representación adecuada para el entrenamiento de los clasificadores. Luego se realiza una reducción del conjunto de características generado por la colección (reducción de dimensionalidad) con el fin de mejorar el rendimiento durante el aprendizaje de los clasificadores. Una vez que éstos fueron entrenados, tiene lugar la segunda etapa, llamada etapa de *prueba*, que consiste en la evaluación del desempeño de los clasificadores con nuevos documentos no considerados durante la etapa anterior. En la Figura 1 es posible observar con más detalle los pasos para la construcción de los clasificadores propuestos.

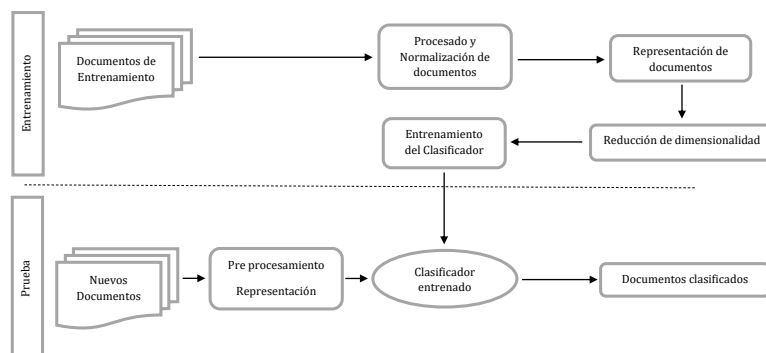


Figura 1: Esquema básico de construcción de los clasificadores.

4.1. Preprocesado y Normalización de documentos

En este paso se busca definir *tokens* o términos, para ello, en los clasificadores propuestos un *token* queda conformado por aquella cadena de caracteres delimitados por espacios en blanco. Además, en esta instancia, se descartan aquellos caracteres tales como símbolos y números ya que no aportan información alguna para la clasificación. Y también aquellos *tokens* o términos identificados como "palabras de parada" (*stopwords*), este conjunto de palabras está constituido por preposiciones, artículos, pronombres, conjunciones, contracciones y ciertos verbos y adverbios. En este trabajo se usó para tal fin el conjunto de *stopwords* para el español definido por el proyecto *Snowball* disponible en [12].

4.2. Representación de documentos

Este paso consiste en la transformación de los documentos en una representación adecuada para que el algoritmo de aprendizaje sea capaz de procesarlos. En este trabajo se propone utilizar para la representación de los textos periodísticos el modelo vectorial propuesto por Salton [13] y el esquema de pesado TF-IDF (*Term Frequency - Inverse Document Frequency*)[14]. En CAT este modelo de representación es uno de los más utilizados debido a sus altas prestaciones cuando se combina a esquemas de pesado y normalización de longitud de documentos [14] [15]. En el modelo vectorial los documentos son formalmente representados a través de vectores cuya dimensión estará dada por la cantidad de términos del vocabulario generado por la colección de documentos.

Cada componente del vector representa la importancia que tiene ese término en el documento y en la colección. Salton propone en [15] calcular los pesos mediante la combinación de la frecuencia relativa de los términos (TF) con la frecuencia inversa en los documentos (IDF), de manera que se tiene:

$$TF - IDF(t_j, \mathbf{d}_i) = f_{ij} \cdot \log\left(\frac{N}{df(t_j)}\right) \quad (1)$$

Donde $df(t_j)$ es el número de documentos en los que el término t_j aparece, f_{ij} es la frecuencia del término t_j en el documento \mathbf{d}_i y N es el número de documentos en la colección.

4.3. Reducción de dimensionalidad

Es importante aclarar que los dos pasos anteriores (preprocesado y representación documentos) se realizaron de la misma manera para la construcción de los dos clasificadores propuestos. A partir de este paso (reducción de dimensionalidad) en este trabajo se propone dos variantes para la reducción de dimensionalidad. Una basada en la selección de un subconjunto del conjunto de términos originales, alternativa a la que denominaremos CATST. Y otra basada en la transformación del conjunto de términos originales a la que llamaremos CATLT. A continuación se exponen ambas alternativas:

■ Propuesta CATST

La ley del mínimo esfuerzo de Zipf [16], comprueba que en una colección de documentos coexisten términos muy pocos frecuentes y específicos para determinados documentos, junto con aquellos términos muy frecuentes que representan la colección de documentos en general. En base a esta ley Luhn [17], afirma que existe un rango de términos que son relevantes para un determinado documento, cuando la tarea es la recuperación de documentos a través de una consulta. Esta misma idea se puede aplicar a CAT, es decir, es posible hallar un rango de términos relevantes para cada categoría. En un problema de clasificación de texto lo que se pretende es encontrar términos que tengan el mayor poder de discriminación entre las categorías. Esto implica centrarnos en términos que sean característicos de cada grupo de documentos pertenecientes a cada categoría, es decir, términos de frecuencia media que no son exclusivamente específicos de uno o muy pocos documentos ni absolutamente generales a toda la colección de documentos. Para encontrar este rango de términos proponemos realizar los siguientes pasos:

1. Particionar el conjunto de términos originales ordenados de manera decreciente según su frecuencia, en 4 partes iguales. Tomar como "punto de partida" para la determinación del rango, aquel término que se ubica en la parte media del primer cuarto, tal como se puede apreciar en la Figura:2a.
2. Tomar los términos correspondientes al 10, 20, 30 y sucesivamente hasta un 90 por ciento hacia la izquierda (cut-on) y derecha (cut-off) de este "punto de partida" para formar nueve rangos candidatos, tal como se aprecia en la Figura: 2b.

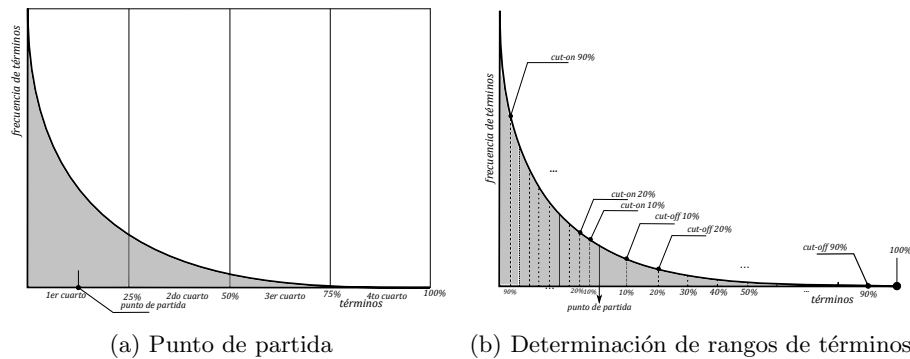


Figura 2: Propuesta CATST.

3. Entrenar un clasificador utilizando cada rango candidato.
4. Evaluar y seleccionar el rango que mejor desempeño obtenga.

■ Propuesta CATLT

Otra alternativa que se propone en este trabajo, es utilizar una técnica de reducción de dimensionalidad basada en la transformación del conjunto de términos originales a través del concepto de *lematización* o por su terminología en inglés, *stemming*. Los algoritmos de lematización de términos son capaces de extraer prefijos y sufijos de palabras que son literalmente diferentes, pero que tienen una raíz en común y que pueden ser consideradas como un mismo término. Cada palabra es "truncada" a su lema o raíz equivalente. Para tal fin en este trabajo se utilizó una adaptación al español del algoritmo de *Porter* [18] [19]. A pesar de que al transformar el espacio de términos en un espacio de raíces este conjunto se reduce notoriamente, se debería considerar solo aquellas raíces que tengan mayor poder de discriminación entre las categorías. A diferencia de los términos originales, cuando se trabaja con raíces, estas últimas tienen mayor poder de discriminación cuando su frecuencia es alta. Para ello proponemos encontrar un rango de raíces de la siguiente manera:

1. Ordenar las raíces en forma decreciente según su frecuencia de aparición.
2. Tomar las raíces correspondientes al 10, 20, 30 y sucesivamente hasta un 90 por ciento a partir de aquella raíz cuya frecuencia de aparición sea máxima, para formar nueve rangos candidatos, como se muestra en la Figura:3.
3. Entrenar un clasificador utilizando cada rango candidato.
4. Evaluar y seleccionar el rango que mejor desempeño obtenga.

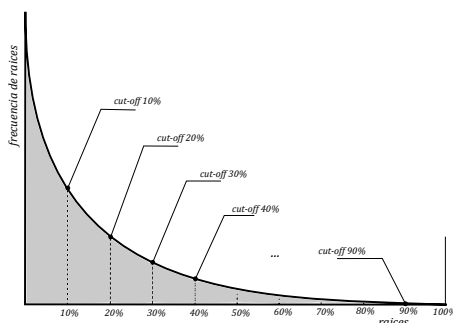


Figura 3: Propuesta CATLT.

4.4. Entrenamiento y prueba de los clasificadores

El entrenamiento y prueba de los clasificadores se llevó a cabo mediante un proceso que realiza dos bucles anidados (Figura:4).

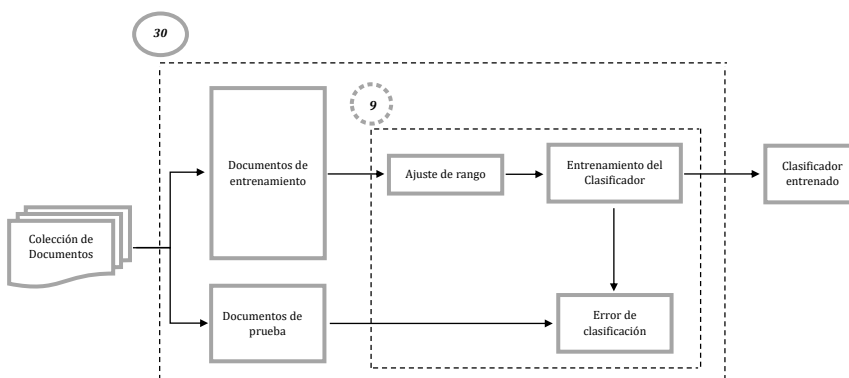


Figura 4: Proceso de entrenamiento y prueba de los clasificadores.

- En el *bucle externo* (recuadro externo en líneas punteadas) se realiza 30 veces la partición de la colección de documentos en un subconjunto de documentos para entrenamiento, seleccionando aleatoriamente un 70 % del total de documentos. El subconjunto de documentos restante (30 % del total de documentos), es utilizado para prueba. De esta manera es posible obtener una mejor estimación del desempeño de los clasificadores propuestos.

- En el *bucle interno* (recuadro interno en líneas punteadas) se entrena un clasificador por cada rango candidato, es decir nueve rangos utilizando ensambles de 100, 500 y 1000 árboles de decisión. Se evalúa el desempeño de cada rango usando el subconjunto de documentos de prueba generado por el *bucle externo*.
- Al finalizar las iteraciones de los dos bucles se selecciona el clasificador entrenado a partir del rango que obtuvo la tasa de error más baja.

5. Experimentación

5.1. Colecciones de noticias

El entrenamiento y evaluación de los clasificadores propuestos se realizó a partir de colecciones de noticias confeccionadas manualmente. El motivo de esta decisión, se debe a la falta de disponibilidad de algún repositorio que contenga alguna con categorías explícitas y en idioma español. Además utilizando los textos originales extraídos de portales de noticias del NOA, se pretende que los clasificadores "aprendan" el estilo de redacción de la zona. Las noticias se obtuvieron a partir de los periódicos digitales más leídos de la región noreste de Argentina [20]. Cada noticia seleccionada aleatoriamente corresponde a un período comprendido entre Octubre de 2015 y Marzo de 2016. Se crearon dos colecciones de documentos, la Tabla 1 muestra los detalles de cada colección.

Tabla 1: Colecciones de noticias: N = cantidad total de noticias, T = cantidad total de términos y C = categorías.

Nombre	N	T	C		
C2PD	200	9084	Policial	Deportes	
C3PES	300	13696	Política	Economía	Salud

La creación de las colecciones tiene la finalidad de evaluar el desempeño de cada clasificador propuesto en diferentes situaciones. Por un lado, el clasificador puede enfrentarse a un problema de clasificación binaria o multiclase. Por otro lado, las categorías podrían tener muchos términos en común, adicionando complejidad a la colección.

5.2. Resultados y evaluación

- Colección C2PD:
En la Tabla 2 se exponen los resultados obtenidos a partir de las experimentaciones realizadas sobre la colección C2PD, en particular, se muestra para

Tabla 2: Resultados de los clasificadores con el mejor rango de términos o raíces encontrado sobre la colección C2PD.

<i>Propuesta</i>	<i>CATST</i>	<i>CATLT</i>
100 árboles		
Cantidad de Características	8733	5202
Error medio de clasificación	0.06825556	0.03532778
Cantidad de caraterísticas en el mejor rango	7860	2601
Tiempo total de ejecución (segundos)	2610.12	561.24
500 árboles		
Cantidad de Características	8733	5202
Error medio de clasificación	0.03679667	0.02134556
Cantidad de caraterísticas en el mejor rango	7859	4681
Tiempo total de ejecución (segundos)	9492.33	2798.16
1000 árboles		
Cantidad de Características	8733	5202
Error medio de clasificación	0.03241778	0.01782056
Cantidad de caraterísticas en el mejor rango	7859	4681
Tiempo total de ejecución (segundos)	16741.23	5631.58

cada ensamble (100, 500 y 1000 árboles de decisión) el error medio de clasificación de 30 corridas de cada clasificador con el mejor rango de términos o raíces encontrado.

En primer lugar se puede observar que la propuesta CATLT al aplicar lematización de términos trabaja con una cantidad de características considerablemente menor que CATST. En segundo lugar al analizar el desempeño de los clasificadores (error medio de clasificación), se puede observar que CATLT (propuesta basada en lematización), obtiene una tasa de error menor a CATST (propuesta basada en la selección de un subconjunto de términos originales). La razón es que este último incluye en el mejor rango encontrado algunos términos con poco poder de discriminación entre las clases. Esto se debe a la dificultad de encontrar un rango que solo contenga términos altamente discriminativos. Para ello, lo que se busca son los términos con una frecuencia de aparición media, ya que éstos son los más informativos para cada clase. Aún así, no todos estos términos van a aportar buena información, llevando en algunos casos a un entrenamiento menos eficaz. Por el contrario en CATLT al trabajar con raíces en vez de términos, el proceso de ajuste del rango es más sencillo ya que solo se debe descartar las raíces con menor frecuencia de aparición. Este proceso lleva a encontrar un rango de raíces con un alto poder de discriminación entre las clases, favoreciendo el entrenamiento del clasificador. Por otro lado es posible observar que con 1000 árboles de decisión se alcanza un mejor desempeño de los clasificadores. En la Figura 5, para el ensamble de 1000 árboles de decisión, se muestran los errores de clasificación obtenidos de 30 corridas en un diagrama de cajas,

que corresponden a los mejores rangos encontrados por cada clasificador. Se puede observar en la gráfica que CATLT produce los errores más bajos y esta diferencia es significativa.

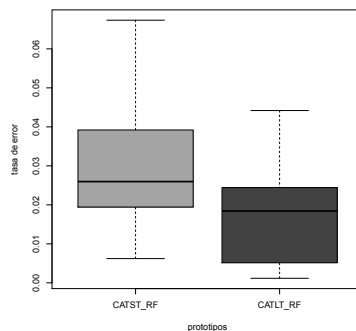


Figura 5: Errores de clasificación para el mejor rango candidato del ensamble de 1000 árboles de decisión.

También se puede observar que el tiempo se ve afectado por la cantidad de características a manipular, llevando menor tiempo de entrenamiento el clasificador CATLT (Tiempo total de entrenamiento, Tabla 2).

Por último las experimentaciones realizadas en la colección de tres categorías (C3PES) ambos clasificadores muestran comportamientos muy similares a las experiencias realizadas en las colecciones de dos categorías (Tabla 3). En términos generales, se pudo comprobar que los clasificadores propuestos son robustos, ya que el desempeño de los mismos se mantiene en las distintas colecciones.

6. Conclusiones y trabajos futuros

En este trabajo se presentó dos clasificadores automáticos de textos periodísticos del noroeste argentino. Los clasificadores desarrollados implementan dos técnicas propuestas para la reducción de dimensionalidad del espacio de características, denominadas CATST (basada en la selección de un sub conjunto de características) y CATLT (basada en lematización).

Después de demostrar el buen desempeño de ambos clasificadores en las diferentes colecciones creadas, nuestros resultados sugieren que:

- La propuesta CATLT produce tasas de errores más bajas que la propuesta CATST en todas las experimentaciones realizadas. Dado que el proceso de ajuste del rango de CATLT, al trabajar con raíces resulta más sencillo que

Tabla 3: Resultados de los clasificadores con el mejor rango de términos y raíces encontrado sobre la colección C3PES.

<i>Propuesta</i>	<i>CATST</i>	<i>CATLT</i>
100 árboles		
Cantidad de Características	13264	6961
Error medio de clasificación	0.2004667	0.1059481
Cantidad de caraterísticas en el mejor rango	11938	1392
500 árboles		
Cantidad de Características	13264	6961
Error medio de clasificación	0.147003	0.07458296
Cantidad de caraterísticas en el mejor rango	11938	6264
1000 árboles		
Cantidad de Características	13264	6961
Error medio de clasificación	0.1360896	0.06907333
Cantidad de caraterísticas en el mejor rango	11938	6264

en CATST. Esto se debe a que una raíz con frecuencia alta implica que aparece muchas veces en documentos pertenecientes a una determinada clase de la colección, siendo esa raíz representativa para esa clase dentro de la colección. Por el contrario en CATST, encontrar un rango de términos informativos para la clasificación implica centrarse en términos de frecuencia media, que no son exclusivamente específicos de uno o muy pocos documentos, ni absolutamente generales a toda la colección de documentos. Esto hace más complicada la determinación de este rango.

- Al momento de comparar el comportamiento de los ensambles, se puede observar que con 1000 árboles de decisión se alcanza un mejor desempeño de los clasificadores.
- Para concluir, se pudo comprobar la robustez de los prototipos propuestos al mantener un buen desempeño en las distintas colecciones.

Varias vías están abiertas para continuar este trabajo, por supuesto se necesita una evaluación más en profundidad de los clasificadores propuestos incluyendo más colecciones y un análisis comparativo con otras técnicas de reducción de dimensionalidad. Además se podrían emplear otros esquemas de pesado para la representación de relevancia de un término dentro de la colección.

Referencias

1. Sebastiani, F.: Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1), 1-47 (2002)
2. Gómez, J. M., de Buenaga Rodríguez, M., and Giráldez, I.: Text categorization for internet content filtering. *Inteligencia Artificial: revista iberoamericana de inteligencia artificial*, 8(22), 147-160, (2004)

3. Shakeri, S., and Rosso, P.: Spam Detection and Email Classification. Information Assurance and Computer Security. IOS Press, 155, 167, (2006)
4. Amati, G., Aloisi, D. D., Giannini, V., and Ubaldini, F.: A framework for filtering news and managing distributed data. Journal of Universal Computer Science, 3(8), 1007-1021, (1997)
5. Stamatatos, E.: A survey of modern authorship attribution methods. Journal of the American Society for information Science and Technology, 60(3), 538-556, (2009)
6. Lui, A. K. F., Li, S. C., and Choy, S. O.: An evaluation of automatic text categorization in online discussion analysis. In Advanced Learning Technologies, 2007. ICALT 2007. Seventh IEEE International Conference on (pp. 205-209), IEEE, (2007)
7. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. 137-142, Springer Berlin Heidelberg (1998)
8. Lewis, D. D., and Ringuette, M.: A comparison of two learning algorithms for text categorization. In Third annual symposium on document analysis and information retrieval, Vol. 33, pp. 81-93, (1994)
9. Apte, C., Damerau, F. J., and Weiss, S. M.: U.S. Patent No. 6,253,169. Washington, DC: U.S. Patent and Trademark Office, (2001)
10. Li, Y. H., and Jain, A. K.: Classification of text documents. The Computer Journal, 41(8), 537-546, (1998)
11. Breiman, L.: Random forests. Machine learning, 45(1), 5-32, (2001)
12. Stopword Spanish Snowball. URL: <http://snowball.tartarus.org/algorithms/spanish/stop.txt>
13. Salton, G.: The SMART retrieval system experiments in automatic document processing. (1971)
14. Salton, G.: Automatic text processing: The transformation, analysis, and retrieval of. Reading: Addison-Wesley. (1989)
15. Salton, G., and Buckley, C.: Term weighting approaches in automatic text retrieval. Information processing and management, 24(5), 513-523, (1988)
16. Zipf, G. K.: Human behavior and the principle of least effort, (1949)
17. Luhn, H. P.: The automatic creation of literature abstracts. IBM Journal of research and development, 2(2), 159-165, (1958)
18. Porter, M. F.: An algorithm for suffix stripping. Program, 14(3), 130-137. (1980)
19. Bordignon, F. R. A., and Panessi, W.: Procesamiento de variantes morfológicas en búsquedas de textos en castellano. Revista Interamericana de Bibliotecología, 24(1), (2011)
20. Todo Jujuy URL: <http://www.todojujuy.com/>, Jujuy al momento URL: <http://www.jujuyalmomento.com/>, Pregón URL: <http://www.pregon.com.ar/>, El Tribuno de Jujuy URL: <http://www.tribuno.info/ujuy/>, El Tribuno de Salta URL: <http://www.tribuno.info/salta/>, Informate Salta URL: <http://informatosalta.com.ar/>, El Intransigente. URL: <http://www.elintransigente.com/>, La Gaceta: URL: <http://www.lagaceta.com.ar/>

Multiple Revision on Horn Belief Bases

Néstor Jorge Valdez¹ and Marcelo A. Falappa²

¹ Department of Computer Science and Information, Fac. de Cs. Exactas y Naturales
Universidad Nacional de Catamarca (UNCa), Argentina

`njvaldez@exactas.unca.edu.ar`,

² Artificial Intelligence Research and Development Laboratory,
Department of Computer Science and Engineering (DCIC)

Universidad Nacional del Sur (UNS),

Institute of Computer Science and Engineering (ICIC), CONICET-UNS
e-mail: `mfalappa@cs.uns.edu.ar`

Abstract. In logic programming, Horn clauses play a basic role, and in many logical constructs their consideration is important. In this paper we study the multiple revision of a belief base where the underlying logic is composed by Horn clauses. The main difficulties as to restricting to the Horn fragment for revision operators by a single sentence are analyzed, and general results are presented about *multiple revision* operators on belief bases. We define *prioritized multiple revision* operators under a more restricted logic than classical propositional logic, i.e. *Horn logic*. We propose a set of postulates and representation theorems for each operation. This work is relevant for multiple revision in areas that employ Horn clauses, such as logic programming and deductive databases applications.

Keywords: Horn Belief Revision, Belief Base, Multiple Change.

1 Introduction

1.1 Motivation

Belief revision studies the process by which a rational agent changes its current beliefs due to the arrival of new information. The best known method so as to carry out this change of belief in a rational agent is the AGM paradigm (see Alchourrón, C. et al. [1]). In recent works, generalizations of this framework, which considers the new information (epistemic input) as a set of logical sentences, were presented. In them, we see the importance of considering multiple change operators because they offer the possibility of taking incoming information as separate pieces of information that could be treated differently during the process of change (see Fuhrmann, A. et al. [7], Falappa, M. et al. [6]).

In recent years there have been several studies that show a significant effort in defining operations of contraction and revision AGM style operating under Horn logic (see Wassermann, R. et al. [14], Zhuang, Z. [15], Valdez, N. et al. [13]). From these studies it is noted that there are some differences between classical AGM revision and Horn revision. These differences occur from the weakened expressibility of Horn clause theories. In reality, the problem is not in the Horn revision specifically. Rather, it is considered that when we focus on Horn theories,

belief change operators are not interdefinable, or rather, not easily interdefinable (see Zhuang, Z. et al. [16]). Belief change under Horn logic is important for several reasons. Firstly, many artificial intelligence systems are expressed in Horn clause language, and have also found widespread use in database theory, in areas such as logic programming, truth maintenance systems and deductive databases.

In this paper, we consider a type of multiple change under the language (expressively weak) of Horn clauses. We focus on belief revision, where new information is consistently incorporated into *the belief base* of the agent. We treat prioritized multiple change operators, where it is assumed that the new beliefs should be fully accepted. Every sentence of this input set is simultaneously processed in the revision³. As the main contribution of this paper, we characterize Horn prioritized multiple revision through a set of postulates. In each case, we present different constructions via kernel change and partial meet change techniques.

1.2 Preliminaries

We present here the terminology that we will use in the rest of the paper. We consider a propositional language \mathcal{L} , on a set of literals $\mathbf{P} = \{a, b, \dots\}$, with semantics of a standard theoretical model. We adopt a propositional language \mathcal{L} with a complete set of boolean connectives: $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$. The symbol \top represents a tautology or *truth* and the symbol \perp represents a contradiction or *falsum*. Lowercase Greek characters α, δ, \dots denote formulas and uppercase Latin characters A, B, C, \dots denote sets of formulas. The characters γ and σ are reserved to represent the functions of selection and incision to change operators, respectively. We use a consequence operation Cn that takes each set of sentences to obtain another set of sentences and satisfies the standard properties of Tarski.

We make extensive use of the theory of Horn logic. A Horn clause is a clause with at most one positive literal. A Horn formula is a conjunction of Horn clauses. A Horn theory is a set of Horn formulas. The Horn language \mathcal{L}_H is a restriction \mathcal{L} for Horn formulas. The Horn logic obtained \mathcal{L}_H has the same semantics as propositional logic obtained from \mathcal{L} , but restricted to Horn formulas.

The remainder of this paper is organized as follows: Section 2 revision basic notions are given, presenting classic single revision and then multiple revision. We conclude this section by introducing beliefs change with a Horn logic, and related works that have been carried out in this area reasoning. In Section 3 a representation of results is presented for multiple revision in Horn fragment. This extension is completed defining two operators of Horn prioritized multiple revision: Horn prioritized partial meet multiple revision operator, and Horn prioritized kernel multiple revision operator. In Section 3, we present the conclusions and possible research lines for future work.

³ We do not consider the partial acceptance or disjunctive acceptance of new incoming beliefs (Falappa, M. et al. [6]).

2 Background

2.1 Multiple Revision

The term *multiple revision* is used to refer to revision operations which allow simultaneous revision by more than one sentence. They must be distinguished from *repeated* or *iterated* revision, i.e., the application of two or more revisions in a sequence. Delgrande and Yi [3] developed a general framework for belief revision called *parallel belief revision*. They present a *basic approach*, where they develop the minimal conditions for revision by a set of formulas. In the basic approach, they perform an adaptation of the AGM postulates for revision by a set of formulas, which in turn are similar to the postulates given in [12], adapted for belief states. It is shown in [12] that Groves [9] representation theorem can be generalized for revision by a set of formulas.

Given that we deal with a finite language, the systems of spheres of Grove's construction are interdefinable with faithful rankings, and so the representation theorem of [10] can also be generalized. A revision operator \otimes satisfies basic and supplementary postulates for revision $(K \otimes 1) - (K \otimes 8)$ iff there exists a faithful ranking \leq_K for an arbitrary belief state K , such that for any set of sentences S : $(K \otimes S) = T(\min(\text{Mod}(S), \leq_K))$ [3].

Falappa et al. [6] presented various types of constructions to characterize multiple belief change (syntactic point of view). Here, we consider two of those kinds of constructions resulting from generalization techniques of revision on belief bases.

2.2 Belief change with Horn Logic

In the AGM model, an underlying logic is assumed that is at least as expressive as propositional logic and wherein a set of beliefs is represented by its logical closure. For several reasons work with sets deductively closed is not attractive from a computational point of view. Many systems operate under nonclassical logic, so the AGM model cannot be directly applied. Preferably, Horn revision theory does not diverge too much from the theory of regular revision, as the same intuitions guide both inquiries. A Horn revision operator will be a function which maps Horn knowledge bases and Horn formulas to Horn knowledge bases.

Delgrande and Peppas [4] present the main difficulties in restricting to the Horn fragment, (from a semantic point of view), as well as what can be done to overcome them. They consider Katsuno and Mendelzon's representation theorems [10] and wonder if this result representation which connects the revision postulates with faithful assignments still holds in the Horn fragment, and if any concrete operators exist for the Horn fragment.

3 Horn Multiple Revision

In this section, we develop multiple revision of Horn belief base and propose new postulates of rationality that are adopted from the postulates for multiple revision of Falappa et al. [6]. In these postulates, prevailing three fundamental principles: the new information (set of sentences) must appear at the Horn belief

base revised, the revised base must be consistent and the operation of multiple revision have to change beliefs as least as possible. Let HB be (for its acronym in English: **H**orn belief **B**ase) a Horn belief base, A , and B set of Horn clauses, and $*$ an operator of Horn multiple revision that satisfies the following postulates:

- ($HB * 1$) **Inclusion** $HB * A \subseteq HB \cup A$.
- ($HB * 2$) **Success** $A \subseteq HB * A$.
- ($HB * 3$) **Weak Success** If A is consistent then $A \subseteq HB * A$.
- ($HB * 4$) **Relative Success** $A \subseteq HB * A$ or $HB * A = HB$.
- ($HB * 5$) **Consistency** If A is consistent then $HB * A$ is consistent.
- ($HB * 6$) **Vacuity 1** If A is inconsistent then $HB * A = HB$.
- ($HB * 7$) **Vacuity 2** If $HB \cup A \not\vdash \perp$ then $HB * A = HB \cup A$.
- ($HB * 8$) **Uniformity 1** Given A and B two consistent sets, for all subset X of HB , if $(X \cup A) \vdash \perp$ if and only if $(X \cup B) \vdash \perp$ then $HB \setminus (HB * A) = HB \setminus (HB * B)$.
- ($HB * 9$) **Uniformity 2** Given A and B two consistent sets, for all subset X of HB , if $(X \cup A) \vdash \perp$ if and only if $(X \cup B) \vdash \perp$ then $HB \cap (HB * A) = HB \cap (HB * B)$.
- ($HB * 10$) **Relevance** If $\alpha \in HB \setminus (HB * A)$ then there is a set C such that $HB * A \subseteq C \subseteq (HB \cup A)$, C is consistent with A but $C \cup \{\alpha\}$ is inconsistent with A .
- ($HB * 11$) **Core-Retainment** If $\alpha \in HB \setminus (HB * A)$ then there is a set C such that $C \subseteq (HB \cup A)$, C is consistent with A but $C \cup \{\alpha\}$ is inconsistent with A .

These postulates are adaptations of similar postulates from multiple revision [6].

3.1 Horn Prioritized Multiple Revision

The purpose of this paper is to extend to the Horn logic the two classes of operators of prioritized multiple revision. Here also, for the construction of operators we will consider the generalization techniques of revision of classical belief base.

Horn Prioritized Multiple Partial Meet Revision

We begin by to obtain the operator of Horn prioritized multiple partial meet revision, but we need first to define the set of consistent remains A -consistent-remainders.

Definition 1. Let HB be and Horn belief base, and A consistent set of Horn clauses. The set of A -consistent-remainders of HB , noted by $HB \perp_{\top} A$, is the set of sets X such that:

1. $X \subseteq HB$.
2. $X \cup A$ is consistent.
3. For any X' such that $X \subset X' \subseteq HB$ then $X' \cup A$ is inconsistent.

That is, $HB \perp_{\top} A$ is the set of maximal HB -subsets consistent with A .

The revision by a set of Horn clauses is based on the concept of A -consistent-remainders. In order to complete the construction, we must define a selection function that selects the best consistent remainders.

Definition 2. Let HB be and Horn belief base. γ is and consolidated selection function for HB if and only if, for all Horn clauses set A :

1. If $HB \perp_{\top} A \neq \emptyset$ then $\emptyset \neq \gamma(HB \perp_{\top} A) \subseteq HB \perp_{\top} A$.
2. If $HB \perp_{\top} A = \emptyset$ then $\gamma(HB \perp_{\top} A) = \{HB\}$.

Observation 1 Let HB be Horn belief base and A and consistent set of Horn clauses. Suppose that $\alpha \in HB$ and $\alpha \in A$, then $\alpha \in X$ for all $X \in HB \perp_{\top} A$ and, therefore, $\alpha \in \bigcap (HB \perp_{\top} A)$.

Proof. Follows from the result of (Falappa et al. [6]).

From the above observation and Definition 2 it follows that all the Horn sentences of $HB \cap A$ are 'protected' due to that they are included in the intersection of any collection of remainders. That is, a consolidated selection function selects a subsets of the set $HB \perp_{\top} A$ whose elements all contain the set $HB \cap A$.

Definition 3. Let HB be and Horn belief base and γ consolidated selection function for HB . The Horn prioritized multiple partial meet revision on HB that is generated by γ is the operator $*_{\gamma}$ such that, for all set of Horn clauses A :

$$HB *_{\gamma} A = \begin{cases} \bigcap \gamma(HB \perp_{\top} A) \cup A & \text{if } A \text{ is consistent} \\ HB & \text{otherwise} \end{cases}$$

An operator $*$ is and Horn prioritized multiple partial meet revision on HB if and only if there is and consolidated selection function γ for HB such that for all sets A , $HB * A = HB *_{\gamma} A$.

Example 1. Let $HB = \{p, q, q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t\}$ y $A = \{\neg p, \neg t\}$. Then we have $HB \perp_{\top} A = \{\{q, q \rightarrow r, r \rightarrow s\}, \{q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t\}\}$. We have three possible outcomes for the consolidated selection function and operators Horn prioritized multiple partial meet revision associated.

$$\begin{aligned} \gamma_1(HB \perp_{\top} A) &= \{q, q \rightarrow r, r \rightarrow s\} \text{ y } HB *_{\gamma_1} A = \{q, q \rightarrow r, r \rightarrow s, \neg p, \neg t\}. \\ \gamma_2(HB \perp_{\top} A) &= \{q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t\} \text{ y } HB *_{\gamma_2} A = \{q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t, \neg p, \neg t\}. \\ \gamma_3(HB \perp_{\top} A) &= \{q \rightarrow r, r \rightarrow s\} \text{ y } HB *_{\gamma_3} A = \{q \rightarrow r, r \rightarrow s, \neg p, \neg t\}. \end{aligned}$$

Theorem 1. For each Horn belief base HB , $*$ is and operator Horn prioritized multiple partial meet revision if and only if satisfies $(HB * 1)$, $(HB * 3)$, $(HB * 5)$, $(HB * 6)$, $(HB * 9)$ and $(HB * 10)$.

Proof. - **Postulates to Construction** Let HB be a set of Horn sentences and $*$ and operator that satisfies the postulates $(HB * 1)$, $(HB * 3)$, $(HB * 5)$, $(HB * 6)$, $(HB * 9)$, and $(HB * 10)$. We must show that $*$ is and Horn prioritized multiple partial meet revision.

Part A:

1. γ is a well defined function. Suppose that $HB \perp_{\top} A = HB \perp_{\top} B$ for A and B two consistent sets. It follows from $(HB * 9)$ that $HB \cap (HB * A) = HB \cap (HB * B)$. Therefore, $\gamma(HB \cap (HB * A)) = \gamma(HB \cap (HB * B))$.
2. If $HB \perp_{\top} A = \emptyset$ then $\gamma(HB \perp_{\top} A) = \{HB\}$ from the definition of γ .

3. If $HB^\perp_\top A \neq \emptyset$ we must show that $\emptyset \neq \gamma(HB^\perp_\top A) \subseteq HB^\perp_\top A$. It follows from the definition of γ that $\gamma(HB^\perp_\top A) \subseteq HB^\perp_\top A$. Now we prove that $\gamma(HB^\perp_\top A) \neq \emptyset$. By $(HB * 3)$ it holds that $A \subseteq HB * A$ and, consequently $(HB \cap (HB * A)) \cup A \subseteq HB * A$. On the other hand, it follows from $(HB * 5)$ that $HB * A \not\vdash \perp$ and, therefore, we can conclude that $(HB \cap HB * A) \cup A \not\vdash \perp$. Hence, there must be some set $X \in HB^\perp_\top A$ such that $HB \cap HB * A \subseteq X$. Thus $\gamma(HB^\perp_\top A) \neq \emptyset$.

Part B: $*_\gamma$ is equal to $*$, that is, $HB *_\gamma A = HB * A$, for any set A . Suppose that A is consistent.

a Let $\alpha \in HB * A$. We must show that $\alpha \in HB *_\gamma A$. We have two cases:

- $\alpha \in A$ then it is trivial that $\alpha \in HB *_\gamma A$.
- $\alpha \notin A$. By $(HB * 1)$ it holds that $HB * A \subseteq HB \cup A$ and we can conclude that $\alpha \in HB$. Hence, $\alpha \in HB \cap HB * A$. Therefore, it follows from the definition of γ that $\alpha \in X$ for all $X \in \gamma(HB^\perp_\top A)$.

b Let $\alpha \notin HB * A$. We must show that $\alpha \notin HB *_\gamma A$. If $\alpha \notin HB * A$ we need to find some $X \in \gamma(HB^\perp_\top A)$ such that $\alpha \notin X$. We have two cases:

- $\alpha \in HB$. By $(HB * 10)$ there exists some C such that $HB * A \subseteq C \subseteq (HB \cup A)$, $C \cup A \not\vdash \perp$ and $C \cup A \cup \{\alpha\} \vdash \perp$. Then we may infer that $\alpha \notin A$ and $\alpha \notin C$. Let $T = HB \cap C$. Then $\alpha \notin T$, $(HB \cap (HB * A)) \subseteq T \subseteq HB$, $T \cup A \not\vdash \perp$ and $T \cup A \cup \{\alpha\} \vdash \perp$. Then we may extend T to a maximal subset X of HB consistent with A such that $(HB \cap (HB * A)) \subseteq X$ and $\alpha \notin X$. Therefore, $X \in \gamma(HB^\perp_\top A)$ and $\alpha \notin X$.
- $\alpha \notin HB$. Then no set in $HB^\perp_\top A$ will contain α .

$(H * 6)$ covers the limit case in with A is inconsistent.

- **Construction to Postulates** Let $*_\gamma$ be is and operator Horn prioritized multiple partial meet revision on HB that is generated by γ , being γ an arbitrary consolidated selection function for HB . we must show that $*_\gamma$ satisfies the postulates $(HB * 1)$, $(HB * 3)$, $(HB * 5)$, $(HB * 6)$, $(HB * 9)$, and $(HB * 10)$.

$(HB * 1)$ As during the revision is obtained $X \subseteq HB$ being $X \in HB^\perp_\top A$, then this postulate is trivially shown.

$(HB * 3)$ Suppose that A is consistent. By definition $A \subseteq HB *_\gamma A$, then it is trivially shown.

$(HB * 5)$ Suppose that A is consistent. If $HB^\perp_\top A \neq \emptyset$ then by definition every $X \in HB^\perp_\top A$ is consistent with A . Therefore, the intersection of any subset of $HB^\perp_\top A$ is consistent with A . We conclude that $HB *_\gamma A$ is consistent.

$(HB * 6)$ Trivial by definition.

$(HB * 9)$ Given A y B two consistent sets. We must show that $HB \cap (\bigcap \gamma(HB^\perp_\top A) \cup A) = HB \cap (\bigcap \gamma(HB^\perp_\top B) \cup B)$. Now, observing that: $HB \cap (\bigcap \gamma(HB^\perp_\top A) \cup A) = (HB \cap (\bigcap (\gamma(HB^\perp_\top A)))) \cup (HB \cap A) = (\bigcap \gamma(HB^\perp_\top A)) \cup (HB \cap A) = \bigcap \gamma(HB^\perp_\top A)$, where the last equality is due to the fact that $HB \cap A \subseteq X$ for all $X \in HB^\perp_\top A$ (see Observation 1) and, consequently, $HB \cap A \subseteq \bigcap \gamma(HB^\perp_\top A)$. Analogously: $HB \cap (\bigcap \gamma(HB^\perp_\top B) \cup B) = \bigcap \gamma(HB^\perp_\top B)$. Since γ is a well defined function

then $\gamma(HB^{\perp_{\top}}A) = \gamma(HB^{\perp_{\top}}B)$ (attentive to $\gamma(HB^{\perp_{\top}}A) = \gamma(HB^{\perp_{\top}}B)$) continue our assumption that for all subsets X de HB , $(X \cup A) \vdash \perp$ if and only if $(X \cup B) \vdash \perp$. Hence, $\bigcap \gamma(HB^{\perp_{\top}}A) = \bigcap \gamma(HB^{\perp_{\top}}B)$. Therefore: $HB \cap (\bigcap \gamma(HB^{\perp_{\top}}A) \cup A) = HB \cap (\bigcap \gamma(HB^{\perp_{\top}}B) \cup B)$.
 $(HB * 10)$ Suppose that $HB^{\perp_{\top}}A \neq \emptyset$. Give $\beta \in HB$ and $\beta \notin HB *_{\gamma} A$. It means there is some $X \in HB^{\perp_{\top}}A$ such that $\beta \notin X$. Hence, there is some X such that $\beta \notin X$, $X \cup A$ is consistent but $X \cup A \cup \beta$ is inconsistent. Now, suppose that $HB^{\perp_{\top}}A = \emptyset$ in this case A is inconsistent. By definition, $HB *_{\gamma} A = HB$ and relevance is vacuously satisfied.

Horn Prioritized Multiple Kernel Revision

Now let's get, the operator Horn prioritized multiple kernel revision, which is based on sets A -inconsistent-kernels.

Definition 4. Let HB be and Horn belief base, and A consistent set of sentences of Horn clauses. The set of A -inconsistent-kernels of HB , noted by $HB^{\perp_{\perp}}A$, is the set of sets X such that:

1. $X \subseteq HB$.
2. $X \cup A$ is inconsistent.
3. For any X' such that $X' \subset X \subseteq HB$ then $X' \cup A$ is consistent.

That is, $HB^{\perp_{\perp}}A$ is the set of minimal HB -subsets inconsistent with A .

In order to complete the construction, we must define an incision function that cuts in every inconsistent kernel.

Definition 5. Let HB be a Horn belief base. σ is a consolidated incision function for HB if and only if, for all consistent set of Horn clauses A :

1. $\sigma(HB^{\perp_{\perp}}A) \subseteq \bigcup HB^{\perp_{\perp}}A$.
2. If $X \in HB^{\perp_{\perp}}A$ then $X \cap (\sigma(HB^{\perp_{\perp}}A)) \neq \emptyset$.

Observation 2 Let HB be a Horn belief base and A consistent set of Horn clauses. Suppose that $\alpha \in HB$ and $\alpha \in A$, then $\alpha \notin \bigcup (HB^{\perp_{\perp}}A)$ and , therefore, $\alpha \cap \bigcup (HB^{\perp_{\perp}}A) = \emptyset$.

Proof. His demonstration is similar in procedure to the observation of 1.

From the above observation and Definition 5 it follows that all the Horn sentences of A are 'protected' it the sense that they can not be considered for removing by the consolidated incision function. That is, a consolidated incision function selects among the sentences of $HB \setminus A$ that make $HB \cup A$ inconsistent.

Definition 6. Let HB be and Horn belief base and σ a consolidated incision function for HB . The Horn prioritized multiple kernel revision on HB that is generated by σ is the operator $*_{\sigma}$ such that, for all set of Horn clauses A :

$$HB *_{\sigma} A = \begin{cases} (HB \setminus \sigma(HB^{\perp_{\perp}}A)) \cup A & \text{if } A \text{ is consistent} \\ HB & \text{otherwise} \end{cases}$$

An operator $*$ is a Horn prioritized multiple kernel revision for HB if and only if there is a consolidated incision function σ for HB such that for all sets A , $HB * A = HB *_{\sigma} A$.

Example 2. Let $HB = \{p, q, q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t\}$ and $A = \{\neg p, \neg t\}$. Then we have $HB \perp_{\perp} A = \{\{p\}, \{q, q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t\}\}$. Some of the possible outcomes for the consolidated incision function and operators Horn prioritized multiple kernel revision associated are:

$$\begin{aligned}\sigma_1(HB \perp_{\perp} A) &= \{p, q \wedge s \rightarrow t\} \text{ and } HB *_{\sigma_1} A = \{q \wedge s \rightarrow t, \neg p, \neg t\}. \\ \sigma_2(HB \perp_{\perp} A) &= \{p, q, q \wedge s \rightarrow t\} \text{ and } HB *_{\sigma_2} A = \{q, q \rightarrow r, r \rightarrow s, \neg p, \neg t\}. \\ \sigma_3(HB \perp_{\perp} A) &= \{p, q\} \text{ and } HB *_{\sigma_3} A = \{q \rightarrow r, r \rightarrow s, q \wedge s \rightarrow t, \neg p, \neg t\}. \\ \sigma_4(HB \perp_{\perp} A) &= \{p, q, q \rightarrow r\} \text{ and } HB *_{\sigma_4} A = \{r \rightarrow s, q \wedge s \rightarrow t, \neg p, \neg t\}.\end{aligned}$$

Theorem 2. *For each Horn belief base HB , $*$ is and operator Horn prioritized multiple kernel revision if and only if satisfies $(HB * 1)$, $(HB * 3)$, $(HB * 5)$, $(HB * 6)$, $(HB * 8)$, y $(HB * 11)$.*

Proof. - Postulates to Construction Let HB be Horn belief base and $*$ an operator that satisfies the postulates $(HB * 1)$, $(HB * 3)$, $(HB * 5)$, $(HB * 6)$, $(HB * 8)$, y $(HB * 11)$. We must show that $*$ is a Horn prioritized multiple kernel revision.

Part A:

1. σ is a well defined function. Suppose that $HB \perp_{\perp} A = HB \perp_{\perp} B$ for A and B two consistent sets. It follows from $(HB * 8)$ que $HB \setminus (HB * A) = HB \setminus (HB * B)$. Therefore, $\sigma(HB \perp_{\perp} A) = \sigma(HB \perp_{\perp} B)$.
2. $\sigma(HB \perp_{\perp} A) \subseteq \bigcup HB \perp_{\perp} A$. If $\alpha \in \sigma(HB \perp_{\perp} A)$ then $\alpha \in \bigcup (HB \perp_{\perp} A)$ and then $\alpha \in HB \setminus HB * A$. By $(HB * 11)$ there is any some C such that $C \subseteq (HB \cup A)$, $C \cup A \not\vdash \perp$ but $C \cup A \cup \{\alpha\} \vdash \perp$. Let X be an arbitrary element $C \cup \{\alpha\} \perp_{\perp} A$ where $C \cup \{\alpha\} \perp_{\perp} A \neq \emptyset$. Then X such that $X \subseteq C \cup \{\alpha\}$, $X \cup A \vdash \perp$ and, for any X' such that $X' \subset X$, it holds that $X' \cup A \not\vdash \perp$. Means $\alpha \in X$ and that $X \cap A = \emptyset$. As $X \subseteq C \cup \{\alpha\}$ and $C \subseteq (HB \cup A)$ we obtain that $X \subseteq HB$. Hence, $X \in HB \perp_{\perp} A$ and, since $\alpha \in X$, we can conclude that $\alpha \in \bigcup (HB \perp_{\perp} A)$.
3. Let $X \in HB \perp_{\perp} A$. We need to show that $X \cap \sigma(HB \perp_{\perp} A) \neq \emptyset$. Suppose that A is consistent. By $(HB * 5)$ then $HB * A$ is consistent. Since X is inconsistent with A by $(HB * 3)$ then $X \not\subseteq HB * A$. Therefore, there is some $\beta \in X$ and $\beta \notin HB * A$, which means that $\beta \in \sigma(HB \perp_{\perp} A)$.

Parte B: $*_{\sigma}$ is equal to $*$, that is, $HB *_{\sigma} A = HB * A$, for any set A . Suppose that A is consistent.

- a Let $\alpha \in HB * A$. It follows from the definition of $\sigma(HB \perp_{\perp} A)$ que $\alpha \notin \sigma(HB \perp_{\perp} A)$. By $(HB * 1)$ it holds that $HB * A \subseteq HB \cup A$ and we can conclude that $\alpha \in HB \cup A$. Therefore, by definition of $*_{\sigma}$, $\alpha \in HB *_{\sigma} A$.
- b Let $\alpha \notin HB * A$. We must show that $\alpha \notin HB *_{\sigma} A$. By $(HB * 3)$ we have that $\alpha \notin A$. Since by definition $HB *_{\sigma} A$ is equal to $(HB \setminus \sigma(HB \perp_{\perp} A)) \cup A$ then it only remains to show that $\alpha \notin HB \setminus \sigma(HB \perp_{\perp} A)$. To do that we consider two cases:
 - $\alpha \notin HB$. Then it follows trivially that $\alpha \notin HB \setminus \sigma(HB \perp_{\perp} A)$.
 - $\alpha \in HB$. Then it follows from definition of $\sigma(HB \perp_{\perp} A)$ that $\alpha \in \sigma(HB \perp_{\perp} A)$ and, therefore, $\alpha \in HB \setminus \sigma(HB \perp_{\perp} A)$.

$(HB * 6)$ covers the limit case in with A is inconsistent.

- **Construction to Postulates** Let σ be an arbitrary consolidated incision function for HB and $*_{\sigma}$ be the Horn prioritized multiple kernel revision on HB that is generated by σ . Then, for all sets A we must show that $*_{\sigma}$ satisfies the postulates $(HB * 1)$, $(HB * 3)$, $(HB * 5)$, $(HB * 6)$, $(HB * 8)$, and $(HB * 11)$.
 - $(HB * 1)$ Trivial by definition.
 - $(HB * 3)$ Suppose that A is consistent. By definition $A \subseteq HB *_{\sigma} A$, then the proof is trivial.
 - $(HB * 5)$ Suppose that A is consistent. Since, $\sigma(HB^{\perp}_{\perp} A)$ cuts every subset of HB inconsistent with A then $HB \setminus \sigma(HB^{\perp}_{\perp} A)$ is consistent with A .
 - $(HB * 6)$ Trivial by definition.
 - $(HB * 8)$ Given A and B two consistent sets. Suppose that for all subset X of H , $(X \cup A) \vdash \perp$ if and only if $(X \cup B) \vdash \perp$. Since σ is a well defined function then $HB^{\perp}_{\perp} A = HB^{\perp}_{\perp} B$, then $\sigma(HB^{\perp}_{\perp} A) = \sigma(HB^{\perp}_{\perp} B)$. Therefore, $HB \setminus (HB *_{\sigma} A) = HB \setminus (HB *_{\sigma} B)$.
 - $(HB * 11)$ Let $\beta \in HB$ and $\beta \notin HB *_{\sigma} A$. Means $HB *_{\sigma} A \neq HB$ and, by definition $\beta \notin (HB \setminus \sigma(HB^{\perp}_{\perp} A)) \cup A$ y $\beta \in HB$, we can conclude that $\beta \in \sigma(HB^{\perp}_{\perp} A)$. By definition $\sigma(HB^{\perp}_{\perp} A) \subseteq \bigcup HB^{\perp}_{\perp} A$, and it follows that there is some $X \in HB^{\perp}_{\perp} A$ such that $\beta \in X$. X is a minimal HB -subset inconsistent with A . Let $Y = X \setminus \{\beta\}$. Then Y is such that $Y \subset X \subseteq HB \subseteq HB \cup A$, Y is consistent with A but $Y \cup \{\beta\}$ is inconsistent with A .

4 Conclusions and Future Work

In this paper we have explored the multiple revision on belief bases regarding Horn theories. One of our conclusions, tells us that there are some differences between classical AGM revision and Horn revision. These differences occur due to the weakened expressibility of Horn clause theories. Actually, the problem is not in the Horn revision specifically. Rather, it is considered that when we focus on Horn theories, belief change operators are not interdefinable, or rather, not easily interdefinable (see Zhuang, Z. et al. [16]). Preferably, the theory of Horn revision does not diverge too much from the theory of regular revision, as the same intuitions guide both inquiries.

On the other hand, if we focus on belief revision from a semantic point of view, we find some difficulties when restricting to the Horn fragment. This is because only the sets of interpretations closed under intersection can be represented in the Horn fragment. The problem this creates for belief revision is that the *standard model-based operators* might produce results that are not in the fragment. Its solution is to consider a condition *Horn compliant* limiting the allowable pre-orders for the generation of the model-based Horn revision and the addition of an *Acyc* postulate which restricts the cycles in their *faithful rankings* (for further reading see Delgrande and Peppas [4]).

Here, our research has a syntactic approach and it allowed us to realize the extent of prioritized multiple revision operators to Horn logic, preserving their intuitions according to its postulates and constructive models. The main contribution of our research is the construction of operators of prioritized revision

under Horn logic. They are based on maximal consistent remainders (for Horn prioritized multiple partial meet revision) and minimal inconsistent kernels (for Horn prioritized multiple kernel revision). We have proposed a set of postulates and some relations among them. Also, we have presented a representation theorem for each operator. As future work, we will study merge operators or symmetric change operators in Horn fragments. Our aim is to find axiomatic characterizations for each case (partial meet and kernel). Subsequently, we want to investigate the selective multiple revision in Horn theory, that is, revisions where only a subset of the epistemic input is accepted.

References

1. Alchourrón, Carlos E. and Gärdenfors, Peter and Makinson, David: On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*. 50, 510-530 (1985a).
2. Booth, Richard and Meyer, Thomas and Varzinczak, Ivan and Wassermann, Renata. A contraction core for Horn belief change: Preliminary report. In 13th International Workshop on Nonmonotonic Reasoning (NMR), (2010a/b). (2010).
3. Delgrande, James P. and Yi Jin: Parallel belief revision: Revising by sets of formulas. *Artif. Intell.* 176, 1, 2223–2245. (2012).
4. Delgrande, James P. and Pavlos, Peppas: Belief revision in Horn theories. *Artif. Intell.* 218, 1-22. <http://dx.doi.org/10.1016/j.artint.2014.08.006>. (2015).
5. Eiter, Thomas and Gottlob, Georg: On the Complexity of Propositional Knowledge Base Revision, Updates, and Counterfactuals. *Artif. Intell.* 57, 2-3, 227-270. (1992).
6. Falappa, M. and Kern-Isberner and Simari: Prioritized and Non-prioritized Multiple Change on Belief Bases. *Journal of Philosophical Logic*, ISSN 0022-3611, New York: Springer. 41, 77-113 (2012).
7. Fuhrmann, A. and Hansson, Sven Ove: A survey of multiple contractions. *Journal of Logic, Language and Information*. 39-74 (1994).
8. Peter Gärdenfors. *Knowledge in Flux: Modelling the Dynamics of Epistemic States*. The MIT Press, Bradford Books, Cambridge, Massachusetts, 1988.
9. Grove, Adams: Two modellings for theory change. *Journal of Philosophical Logic* 17. 157-170, (1988).
10. Katsuno, Hirofumi and Mendelzon, Alberto O.: Propositional knowledge base revision and minimal change. *Artificial Intelligence*. 52(3), 263-294, (1992).
11. Liberatore, Paolo: Compilability and compact representations of revision of Horn knowledge bases. *ACM Trans. Comput. Log.* 1, 1, 131-161. (2000).
12. Pavlos, Peppas: The Limit Assumption and Multiple Revision. *J. Log. Comput.*, 14, 3, 355-371. (2004). <http://dx.doi.org/10.1093/logcom/14.3.355>.
13. Valdez, Néstor J. and Falappa, Marcelo A.: Dinámica de Conocimiento: Contracción Múltiple en Lenguajes Horn. XIX Argentine Congress of Computer Science Selected Papers. (WASI), 73-82. (2013).
14. Wassermann, Renata and Delgrande, James P.: Topics in Horn Contraction: Supplementary Postulates, Package Contraction, and Forgetting. In *Proc. NRAC* (2011).
15. Zhuang, Zhiqiang: Belief Change Under the Horn Fragment of Propositional Logic. PhD thesis, School of Computer Science and Engineering, Faculty of Engineering, University of New South Wales. (2013).
16. Zhuang, Zhiqiang and Pagnucco, Maurice and Zhang, Yan: Definability of Horn revision from Horn contraction. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, Beijing, China. (2013).

Procesamiento inteligente de flujos de datos vía sistemas argumentativos

José N. Paredes, Fabio R. Gallo, Gerardo I. Simari, y Marcelo A. Falappa
{jose.paredes,fabio.gallo,gis,mfalappa}@cs.uns.edu.ar

Departamento de Ciencias e Ingeniería de la Computación (UNS), Argentina
Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina
Instituto de Ciencias e Ingeniería de la Computación (UNS-CONICET), Argentina

Resumen Con el avance de las tecnologías de la información se hace cada vez más fácil generar y procesar datos con mayor precisión acerca de lo que ocurre en un dominio complejo. Las aplicaciones que se desenvuelven en estos entornos se caracterizan por recibir grandes cantidades de datos en poco tiempo, con altas tasas de cambio, como así también incertidumbre e incompletitud; a este tipo de torrente de datos – que no se puede almacenar para ser procesado más adelante – se los conoce con el nombre de “flujos”. En este trabajo presentamos una propuesta para realizar procesamiento de flujos de manera inteligente a través de sistemas argumentativos, proponiendo una combinación novedosa de (i) técnicas de procesamiento de flujos de datos y eventos complejos, (ii) herramientas de argumentación estructurada, (iii) razonamiento probabilístico y (iv) operadores de revisión de creencias.

Palabras clave: Procesamiento de flujo de datos, Argumentación Estructurada, Razonamiento bajo incertidumbre, Revisión de creencias.

1. Introducción y Motivación

En la actualidad, a medida que las personas desarrollan sus actividades diarias se va produciendo una gran cantidad de información digital heterógena, densa, y muy frecuente, la cual va dejando huellas a partir de registros de llamadas, mensajes de texto, uso de GPS, publicaciones en redes sociales, y muchas otras fuentes más. Claramente, estas huellas pueden ser utilizadas para describir detalles menos evidentes del mundo que nos rodea, siempre teniendo en cuenta no violar las restricciones de privacidad de cada persona.

A su vez, el hecho de que cada vez haya más dispositivos al alcance de todas las personas para producir información más abundante hace que empresas u organismos implementen tecnologías que generan datos acerca de lo que ocurre en el entorno cotidiano. En este contexto surge un dominio de aplicación que intenta aprovechar tal situación pero requiere que el *procesamiento de los flujos de datos* (PFD) se lleve a cabo en un período de tiempo muchas veces pequeño. Ahora bien, debido a las características mencionadas no es posible o necesario procesar *toda* la información entrante; por lo tanto, los sistemas que se desenvuelven en

estos entornos utilizan un concepto conocido como *ventana de datos*, las cuales permiten definir el conjunto de datos que está disponible para su procesamiento durante un lapso de tiempo. Sin embargo, si se quisiera utilizar esta información para aprovechar el valor semántico de la mejor manera posible, el PFD clásico no es suficiente; por consiguiente, proponemos utilizar técnicas de representación de conocimiento y razonamiento para este fin. La *argumentación* se encuentra dentro de esta categoría, y sus orígenes están vinculados con filósofos de la antigüedad que estudiaron la naturaleza involucrada en la actividad humana de dialogar. Concretamente, la argumentación se basa en procesos dialécticos que permite elaborar argumentos a favor y en contra de una determinada posición. Luego, es posible ponderar cada argumento en relación a otros que pueden debilitar esta posición y fortalecer una contraria. Como resultado se puede determinar cuáles son las conclusiones ganadoras luego de realizar todo este proceso. Este enfoque proporciona un acercamiento para manejar información incompleta y posiblemente contradictoria; como ventaja adicional, se puede acceder a las *explicaciones* del proceso que se utilizó para llegar a una determinada conclusión. En la literatura se han propuesto muchas formas de llevar a cabo este proceso, las cuales se dividen principalmente en *abstractas* [8] y *concretas* [10]; las primeras se caracterizan por el análisis de los argumentos disponibles y la relación de ataque entre ellos, mientras que las últimas asumen la disponibilidad de la *estructura interna* de cada argumento.

En este trabajo proponemos combinar las áreas de procesamiento de flujos de datos y argumentación, dando origen a un desarrollo novedoso dentro del área de *procesamiento inteligente de flujos de datos* (PIFD) [5]. Tomamos como base el formalismo DeLP3E planteado en [21,22], el cual divide la base de conocimiento (KB) en dos partes diferentes pero a la vez interrelacionadas: el *modelo del entorno* (EM), el cual es utilizado para definir relaciones entre eventos que, por su naturaleza, deben manejarse con modelos de incertidumbre; y por otro lado, el *modelo analítico* (AM), el cual maneja información potencialmente contradictoria mediante un análisis argumentativo basado en el formalismo DeLP con presuposiciones (PreDeLP) [15]. Este último tiene la posibilidad también de ofrecer las explicaciones que justifican una respuesta ante una determinada consulta. Desde el punto de vista del PFD, los contenidos tanto del EM como del AM pueden verse afectados por las altas tasas de información nueva; por lo tanto, los principales obstáculos que se interponen en el camino son una combinación de los que surgen en cada una de las siguientes áreas:

- (i) *Incompletitud, inconsistencia e incertidumbre*: se debe razonar con información que muchas veces es o bien insuficiente para alcanzar conclusiones certeras, ofrece datos contradictorios, o incluso describe fenómenos inherentemente inciertos (tales como los mercados financieros).
- (ii) *Volumen y velocidad*: la cantidad de información, proveniente de distintas fuentes a altas tasas, hace que sea imposible almacenarla toda para un procesamiento minucioso posterior; se debe entonces tomar la decisión de qué almacenar y qué desechar para que el sistema funcione de la mejor manera posible.

- (iii) *Dinámica de creencias*: como consecuencia de los dos primeros puntos se deriva la necesidad de manejar adecuadamente la dinámica de las creencias del sistema; los operadores de revisión y consolidación en general están pensados para entornos con tasas de cambio mucho menores.
- (iv) *Tratabilidad computacional*: el uso de herramientas ricas de representación de conocimiento y razonamiento generalmente implica pagar un costo en cuanto a complejidad computacional; esto es aun más central en los modelos de razonamiento probabilístico. Se deben desarrollar mecanismos por los cuales se puedan obtener aproximaciones y la degradación en su calidad sea lo más gradual posible; además, idealmente el usuario debería poder conocer los efectos que tiene el hecho de realizar compromisos entre tiempo de ejecución y calidad de las respuestas obtenidas.

Ejemplo 1. Considere el vehículo conducido de manera autónoma desarrollado por Google [12]; para poder transitar por las calles de una ciudad, éste requiere una gran cantidad de información acerca del entorno que lo rodea. Para esto cuenta con un conjunto de sensores:

- El LIDAR (“*Laser Imaging Detection and Ranging*”), que permite identificar objetos y medir la distancia a la que se encuentra los mismos utilizando rayos de luz. Específicamente Google hace uso del Velodyne HDL-64E S2, de alta definición, que cuenta con 64 rayos láser y gira sobre sí mismo 360 grados de manera continua hasta 900 vueltas por minuto con el fin de vigilar la situación alrededor del auto; observa 1,3 millones de puntos por segundo, con el fin de construir una imagen tridimensional de las entidades involucradas (peatones, otros vehículos, alumbrado público, árboles, etc). Tiene un alcance de 50m para el pavimento y de 120m para autos y árboles.
- Un GPS y una unidad de medición inercial. Esta unidad mide la aceleración y la velocidad angular mediante acelerómetros, giróscopos y magnetómetros. Es decir, se identifica con precisión hacia dónde se mueve el auto.
- Cuatro radares de onda de radio; tres están ubicados en el paragolpes delantero, uno en la parte frontal y los otros dos en las esquinas laterales respectivamente. En el paragolpes trasero va colocado el cuarto y último radar. Su función es identificar objetos y medir distancias alrededor del auto.

En conjunto, estos sensores ofrecen la capacidad de conocer lo que sucede en tiempo real alrededor del vehículo; esto significa que esta información varía en *cada instante* de tiempo, y gran parte de ella sólo es relevante durante un lapso de tiempo relativamente corto, tales como la posición de peatones, otros autos y cortes de calles (por manifestaciones sociales, reparaciones, etc.). Por otro lado, también se debe tener en cuenta información que en general no varía a lo largo del tiempo, como por ejemplo datos provenientes de los mapas. Google invirtió mucho esfuerzo en elaborar mapas, tanto para la elaboración de la herramienta *Google Maps* como también *Street View*; ambas contienen valor semántico que va más allá de un mapa tradicional.

Los sensores generan una gran cantidad de datos; el LIDAR produce 1,3 millones de puntos por segundo para un rango de visión horizontal de 360 grados

y un rango de visión vertical de 26,8 grados, y la salida total generada es de 100 Mbps de paquetes UDP [6]. Esto es un gran volumen que no puede guardarse ni procesarse de manera completa. En este contexto el sistema que conduce el vehículo debe tomar decisiones en tiempo real, pero considerando que la información de la cual dispone se encuentra afectada por la incertidumbre. Además, debe disponer de mecanismos que permitan manejar la nueva información que se incorpore (tales como condiciones climáticas actuales) teniendo en cuenta que se pueden producir contradicciones con los datos ya presentes. ■

2. Hacia una combinación de herramientas

En este trabajo proponemos desarrollar un formalismo de procesamiento inteligente de flujos de datos mediante la combinación de sistemas argumentativos con técnicas de procesamiento de flujos clásico. Como se discutió en la sección anterior, la revisión de creencias y el razonamiento probabilístico son centrales para alcanzar este objetivo. Cada uno de estos temas ha sido tratado en la literatura, generando antecedentes de combinaciones entre ellos; sin embargo, no conocemos ningún antecedente que combine todos al mismo tiempo. En esta sección primero repasamos los trabajos más significativos en este sentido (secciones 2.1–2.4), y luego describimos nuestra propuesta en la sección 2.5

2.1. Argumentación estructurada probabilística

En la literatura existen diversos formalismos para argumentación estructurada; en este trabajo elegimos la programación en lógica rebatible (DeLP, en inglés); aquí haremos una breve introducción, y referimos al lector a [10] para un tratamiento más profundo. Un programa DeLP está compuesto por un conjunto de hechos, reglas estrictas, y reglas rebatibles. Este suele denotarse con $\Pi = (\Theta, \Omega, \Delta)$, donde Θ es el conjunto de hechos, Ω es el conjunto de reglas estrictas, y Δ es el conjunto de reglas rebatibles. Los hechos son literales básicos (es decir, sin variables) que representan información atómica o su negación; siempre son verdaderos y no pueden ser contradictorios. Las reglas estrictas representan información de causa y efecto que siempre vale; se especifican a partir de la combinación de literales básicos de la forma $L_0 \leftarrow L_1, \dots, L_n$, donde L_0 es un literal básico y $\{L_i\}_{i>0}$ es un conjunto de literales básicos. Por otro lado, las reglas rebatibles incluyen conocimiento que vale si no se dispone de información que lo contradiga. Estas son parecidas a las reglas estrictas, pero representan conocimiento más débil. Se especifican de la forma $L_0 \prec L_1, \dots, L_n$, donde L_0 es un literal básico y $\{L_i\}_{i>0}$ es un conjunto de literales básico. La negación fuerte está permitida tanto para reglas estrictas como rebatibles con el fin de representar información contradictoria. A partir de estos conceptos elementales ya se puede hablar de la noción de argumento: un *argumento* para un literal L es un par $\langle A, L \rangle$, donde $A \subseteq \Delta$ consiste de una prueba para L de manera que: (1) L es derivado rebatiblemente a partir de A , (2) $A \cup \Omega$ es no contradictorio, y (3) A es un subconjunto mínimo de Δ que satisface (1) y (2), denotado $\langle A, L \rangle$.

El literal L se denomina la *conclusión* soportada por el argumento, y A es el *sopORTE*. Un argumento $\langle B, L \rangle$ es un *subargumento* de $\langle A, L' \rangle$ si y sólo si $B \subseteq A$.

Para manejar incertidumbre probabilística se puede tomar el modelo planteado en [21] donde se combina DeLP con variables de comportamiento aleatorio. De esta manera se pueden definir programas DeLP3E denotados como $P = (\Pi_{EM}, \Pi_{AM}, af)$, donde Π_{EM} permite representar conocimiento incierto sujeto a eventos probabilísticos, Π_{AM} permite representar conocimiento estricto y rebatible (programa DeLP), y af es una función, conocida como *función de anotación*, tal que su dominio es el conjunto Π_{AM} y su imagen es un subconjunto de las formulas lógicas fijas que pueden formarse a partir de los elementos definidos en Π_{EM} . A su vez, el conjunto de todas las fórmulas lógicas compuestas por átomos básicos de Π_{EM} define un conjunto de *mundos posibles*. En cada uno de estos mundos existe un subconjunto de elementos de Π_{AM} que son válidos, lo cual determina un programa DeLP (no probabilístico) que existe en ese mundo en cuestión. Esto da lugar a que un literal L pueda estar garantizado en un mundo y en otro no. Es así como la probabilidad de que un literal L esté garantizado resulta de la suma de las probabilidades de los mundos en los que L está garantizado.

Concretamente Π_{EM} debe seguir algún modelo de representación de eventos aleatorios; para este fin se podría utilizar Redes Bayesianas [19], Redes Lógicas de Markov [7], lógica probabilística de Nilsson [18], entre otros.

2.2. Revisión de creencias

El problema que se estudia en el área de revisión de creencias es el de cómo deben cambiar los estados epistémicos de un agente ante una nueva entrada epistémica; en otras palabras, cómo deben revisarse las creencias ante la presencia de información nueva que posiblemente contradice las creencias establecidas hasta el momento. Tradicionalmente, los estados epistémicos han tomado la forma o bien de conjuntos de creencias (conjuntos de fórmulas lógicas cerrados bajo consecuencia) [1,11] o bases de creencias (conjuntos no cerrados) [14,13]. Es evidente que el problema de la revisión de creencias aparece constantemente en el mundo real y, por lo tanto, cualquier sistema inteligente que funcione con datos del mismo deberá estar equipado con alguna forma de revisión. La metodología típica consiste en el desarrollo de operadores que toman la base de conocimiento actual y la entrada epistémica, y producen una nueva base de conocimiento que corresponde al resultado de revisar las creencias. Estos operadores en general se caracterizan por las propiedades que deben cumplir (expresadas en la forma de postulados); luego se proponen construcciones algorítmicas y se demuestra formalmente que las dos caracterizaciones son equivalentes: este tipo de resultado lleva el nombre de teorema de representación. Como se mencionó en la sección anterior, es necesario investigar operadores que sean especialmente propicios para funcionar en un entorno de PIFD y, en particular, también con aspectos cuantitativos para sacar el mayor provecho posible al razonamiento probabilístico subyacente; hasta donde conocemos, el único trabajo en esta última línea es la propuesta reciente de [22].

2.3. Procesamiento inteligente de flujos de datos

Si bien cada vez se tiene más información disponible a través de la Web o dispositivos móviles, responder preguntas que en principio parecen sencillas se vuelve sumamente complicado. Por ejemplo, se puede llegar a tener la información disponible de los clicks que hace un usuario al navegar la Web, y a partir de éstos uno podría pensar que puede saber cuáles son sus intereses, qué noticias atraen su atención, y tal vez de forma más general determinar cuál es el patrón que siguen los lectores con noticias relacionadas. Sin embargo, este tipo de consultas requiere de sistemas que puedan manejar la naturaleza del mundo real que cambia de manera muy rápida y ofrece obstáculos a nivel semántico. Si bien los datos que cambian rápidamente pueden ser analizados por *sistemas de procesamiento de flujos de datos* especializados en tiempo real [2,4], estos sistemas no pueden realizar tareas de procesamiento *inteligente* (complejas). Con este fin, en la última década se ha investigado y desarrollado sistemas con tales características [5,3]; por otra parte, en [9] se analizan distintas técnicas de procesamiento de *eventos complejos* bajo incertidumbre. Los autores tuvieron en cuenta técnicas basadas en autómatas y en lógicas, considerando la incertidumbre relativa a la imperfección en las reglas que definen los eventos, y la incompletitud y errores derivados del flujo de datos. Como se puede apreciar, la diferencia principal con nuestra propuesta es que estas técnicas sólo realizan un procesamiento con un contenido de nivel semántico muy bajo. Estos desarrollos han planteado soluciones iniciales y delineado los problemas más importantes a resolver; sin embargo, el área de PIFD está todavía en sus primeras etapas de evolución.

2.4. Sistemas argumentativos estructurados en entornos dinámicos

El problema de realizar un procesamiento inteligente de datos provenientes de flujos con valor semántico significativo es sumamente complejo debido a las características de frecuencia y volumen elevado de datos. En la línea de investigación planteada en [20,17,16] los autores trabajaron sobre un formalismo para representar y razonar sobre conocimiento cambiante mediante la *activación y desactivación* de argumentos de acuerdo a la evidencia disponible. Sin embargo, si bien el modelo tiene en cuenta un contexto cambiante, la dinámica del conocimiento considerada no tiene velocidad alta ni tampoco se considera grandes cantidades de datos de entrada, por lo que la complejidad computacional no es un aspecto central. Además, tampoco consideran datos que estén afectados por la incertidumbre.

2.5. Modelo propuesto

Nuestra propuesta se resume en la figura 1, la cual muestra el funcionamiento general del modelo planteado. Se tiene un sistema de procesamiento de flujos que genera eventos y los notifica ya sea a AM_t o a EM_t . AM_p contiene hechos, presuposiciones, reglas estrictas y reglas rebatibles relativas a conocimiento que

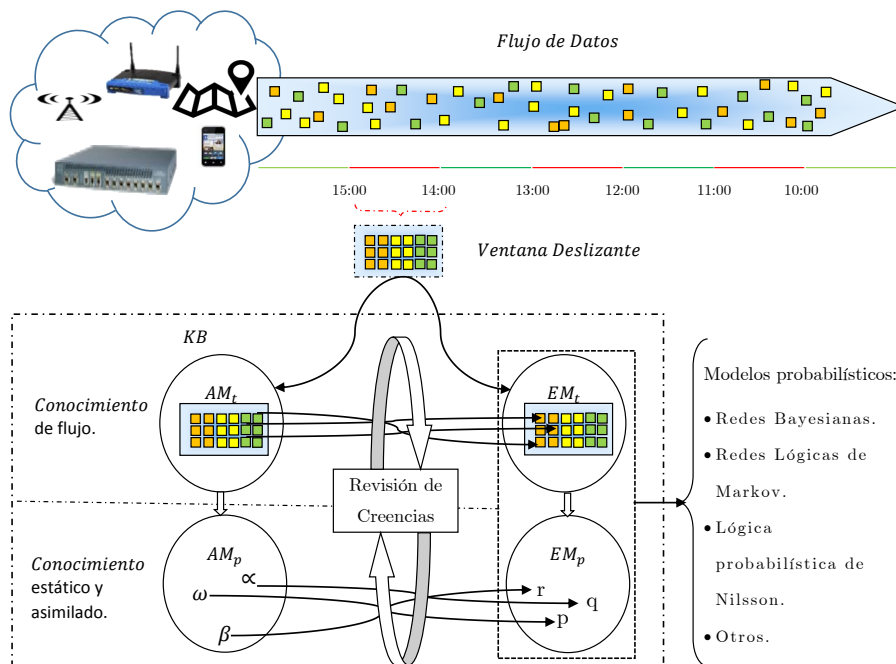


Figura 1. Esquema de un formalismo para combinar técnicas y herramientas de argumentación estructurada, revisión de creencias, razonamiento probabilístico y procesamiento de flujos.

generalmente no varía en el tiempo y ya se encuentra asimilado. Por otro lado, EM_p contiene un modelo probabilístico en relación a variables de comportamiento aleatorio cuya información contenida generalmente tampoco varía en el tiempo y ya se encuentra asimilado. AM_t contiene hechos, presuposiciones, reglas estrictas y reglas rebatibles relativas a conocimiento que generalmente varía en el tiempo y que proviene del flujo. Es importante destacar que los elementos de AM_t pueden pasar a formar parte de AM_p si adquieren una relevancia importante. EM_t contiene un modelo probabilístico en relación a variables de comportamiento aleatorio cuya información contenida generalmente varía en el tiempo y que proviene del flujo. Al igual que en el AM, algunos elementos de EM_t pueden pasar a formar parte de EM_p .

Ejemplo 2. Si se quiere aplicar el modelo planteado en la figura 1 en el ejemplo 1 se podría pensar, de manera muy simplificada, en la siguiente situación: existe un flujo de datos producido a partir de la información obtenida de los sensores del auto conducido de manera autónoma, como puede ser la ubicación de peatones y otros vehículos a una distancia considerada relevante, dirección y velocidad a la que se mueven estos últimos, cortes de calle, señales de tránsito, entre otros; esto se representa y modela en AM_t . También se dispone de conocimiento asimilado,

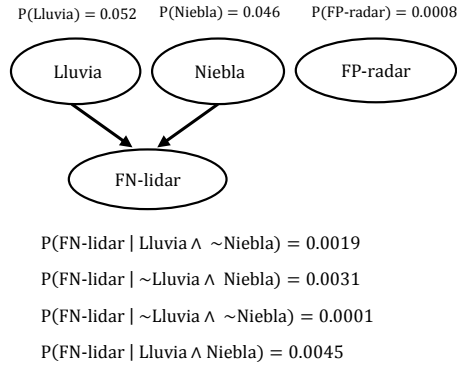


Figura 2. Red Bayesiana utilizada en los modelos EM_p y EM_t del ejemplo 2.

	AM_t	<i>Función de Anotación</i>
Θ :	$\theta_1 = \text{mi_ubicacion}(t1, \text{posicion}, \text{direccion})$	True
	$\theta_2 = \text{peaton_en_frente}$	$\sim\text{FP-radar}$
	$\theta_3 = \sim\text{peaton_en_frente}$	$\sim\text{FN-lidar}$
	AM_p	
Θ :	<i>“Conjunto de hechos que representan el mapa de la ciudad”</i>	True
Ω :	$\Omega_1 = \text{detenerse} \leftarrow \text{peaton_en_frente}$	True
Δ :	$\Delta_1 = \text{ceder_paso} \leftarrow \text{vehiculo_cruzando},$	True

Figura 3. AM_t y AM_p del ejemplo 2, junto con su función de anotación. Los conjuntos de reglas de AM_t son vacíos.

como puede ser la ubicación de cada calle, el sentido de circulación, la ubicación de los semáforos, entre otros; este conocimiento está contenido en AM_p .

Al mismo tiempo, la información provista por el LIDAR se encuentra afectada por la incertidumbre, de manera que su confiabilidad se ve influida por condiciones climáticas como la presencia de neblina y/o lluvias fuertes. Esto podría representarse por tres variables aleatorias que indiquen la probabilidad de que el LIDAR entregue información categorizada como un falso positivo, la probabilidad de que haya neblina en un determinado momento, y la probabilidad de que haya lluvias fuertes en un determinado momento. Una forma de modelar esto es con una red bayesiana como la que se presenta en la figura 2; este conocimiento es parte de EM_p . Por último, se dispone de información proveniente de eventos aleatorios pero que tiene validez acotada en un período de tiempo; por

ejemplo, información de que efectivamente hay niebla en la posición actual del vehículo; esto forma parte de EM_t .

En la figura 3 presentamos un ejemplo simple del contenido de AM_t y AM_p , junto con una función de anotación. Dicho modelo hace la suposición de que EM_t y EM_p contienen los siguientes átomos: *FN-lidar* (probabilidad de que el LIDAR produzca un falso negativo), *FP-radar* (probabilidad de que un radar de ondas de radio produzca un falso positivo), *Lluvia* (probabilidad de que llueva intensamente) y *Niebla* (probabilidad de que haya niebla). Supongamos que el auto percibe la presencia de niebla; en este caso, en EM_t se instancia la variable *Niebla* con valor *verdadero*. Puede observarse en la figura 3 que en AM_t hay dos átomos que son contradictorios: *peaton.en.frente* y \sim *peaton.en.frente*. Sin embargo su anotación es diferente, el primero existe en los mundos donde no es un falso positivo del radar de ondas de radio, y el segundo en los mundos donde no es un falso negativo del LIDAR. Así, el sistema debería decidir cuál es más probable de acuerdo a la probabilidad de que haya ocurrido un falso negativo del LIDAR (teniendo en cuenta que hay niebla) y a la probabilidad de haya ocurrido un falso positivo del radar de ondas de radio. ■

3. Discusión y trabajo futuro

En este trabajo hemos planteado las dificultades y desafíos involucrados en la generación y procesamiento de conocimiento que se presenta en forma de *flujos*; es decir información con un valor semántico en contextos donde se producen grandes volúmenes de datos con alta frecuencia. La principal contribución es la propuesta de un modelo como punto de partida para realizar procesamiento inteligente de flujo de datos, utilizando técnicas de argumentación para aprovechar el contenido semántico por medio de sus capacidades para manejar información contradictoria y para otorgar explicaciones para las respuestas brindadas, lo cual puede ser muy útil o incluso necesario en ciertos dominios. La novedad de la propuesta radica en que hasta este momento no existen desarrollos que combinen procesamiento de flujos de datos, argumentación estructurada, revisión de creencias, y razonamiento bajo incertidumbre.

Como parte de esta línea de investigación, el trabajo actual y futuro involucra el desarrollo de algoritmos que integren las herramientas existentes para el procesamiento de flujos de datos y de eventos complejos con sistemas argumentativos de manera eficiente, ya que la tratabilidad computacional es un aspecto crucial para su eficacia. Otro objetivo principal es el desarrollo de operadores de revisión de creencia adecuados para entornos con conocimiento sumamente cambiante e incierto, lo cual dependerá también del uso adecuado de modelos probabilísticos eficientes.

Agradecimientos. Este trabajo fue parcialmente financiado por fondos provistos por la Universidad Nacional del Sur y CONICET, Argentina.

Referencias

1. Alchourrón, C.E., Gärdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. *Journal of symbolic logic* 50(02), 510–530 (1985)
2. Arasu, A., Babu, S., Widom, J.: The CQL continuous query language: Semantic foundations and query execution. *VLDB Journal* 15(2), 121–142 (2006)
3. Barbieri, D.F., Braga, D., Ceri, S., VALLE, E.D., Grossniklaus, M.: C-SPARQL: A continuous query language for rdf data streams. *International Journal of Semantic Computing* 4(01), 3–25 (2010)
4. Cugola, G., Margara, A.: Processing flows of information: From data stream to complex event processing. *ACM Computing Surveys* 44(3), 15 (2012)
5. Della Valle, E., Ceri, S., Harmelen, F.v., Fensel, D.: It’s a streaming world! reasoning upon rapidly changing information. *IEEE Intell. Sys.* 24(6), 83–89 (2009)
6. Deyle, T.: Velodyne HDL-64E laser rangefinder (lidar) pseudo-disassembled. Online blog: <http://www.hizook.com/blog/2009/01/04/velodyne-hdl-64e-laser-rangefinder-lidar-pseudo-disassembled> (2009)
7. Domingos, P., Kok, S., Lowd, D., Poon, H., Richardson, M., Singla, P.: *Markov Logic*, pp. 92–117. Springer Berlin Heidelberg, Berlin, Heidelberg (2008)
8. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence* 77(2), 321–357 (1995)
9. Gal, A., Wasserkrug, S., Etzion, O.: Event processing over uncertain data. In: *Reasoning in Event-Based Distributed Systems*, pp. 279–304. Springer (2011)
10. García, A.J., Simari, G.R.: Defeasible logic programming: An argumentative approach. *Theory and practice of logic programming* 4(1+ 2), 95–138 (2004)
11. Gärdenfors, P.: *Belief revision*, vol. 29. Cambridge University Press (2003)
12. Guizzo, E.: How Google’s self-driving car works. *IEEE Spectrum Online*, October 18 (2011)
13. Hansson, S.: Semi-revision. *J. Appl. Non-Class. Log.* 7(1-2), 151–175 (1997)
14. Hansson, S.O.: Kernel contraction. *Journal of Symbolic Logic* 59, 845–859 (9 1994)
15. Martínez, M.V., García, A.J., Simari, G.R.: On the use of presumptions in structured defeasible reasoning. In: *Computational Models of Argument - Proceedings of COMMA 2012*, Vienna, Austria, September 10-12, 2012. pp. 185–196 (2012)
16. Moguillansky, M.O., Rotstein, N.D., Falappa, M.A., García, A.J., Simari, G.R.: Dynamics of knowledge in DeLP through argument theory change. *Theory and Practice of Logic Programming* 13(06), 893–957 (2013)
17. Moguillansky, M.O., Rotstein, N.D., Falappa, M.A., García, A.J., Simari, G.R.: Argument theory change through defeater activation. In: *COMMA*. pp. 359–366 (2010)
18. Nilsson, N.J.: Probabilistic logic. *Artificial intelligence* 28(1), 71–87 (1986)
19. Pearl, J.: *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. (1988)
20. Rotstein, N.D., Moguillansky, M.O., García, A.J., Simari, G.R.: A dynamic argumentation framework. *COMMA* 216, 427–438 (2010)
21. Shakarian, P., Simari, G.I., Moores, G., Paulo, D., Parsons, S., Falappa, M.A., Aleali, A.: Belief revision in structured probabilistic argumentation: Model and application to cyber security. *AMAI, to appear* (2016)
22. Simari, G.I., Shakarian, P., Falappa, M.A.: A quantitative approach to belief revision in structured probabilistic argumentation. *AMAI* 76(3-4), 375–408 (2016)

Vigilancia Tecnológica e Inteligencia Competitiva basada en Técnicas de Minería de la Web

Karina Eckert¹, Victor Alvarenga¹, Matias Barboza¹, Leandro Witzke¹, Luis Airaldi²

¹Univer. Gastón Dachary. Av. López y Planes 6519, Posadas, Misiones.

²Univ. Tecnológica Nacional, Facultad Regional Resistencia. French 414, Resistencia, Chaco.
{karinaeck, matias.mbz, leanwitzke, luislezcair}@gmail.com, alva_victor@hotmail.com

Abstract. Tomar decisiones estratégicas es un proceso complejo que requiere información fiable y actualizada. Por ello es necesario contar con herramientas que faciliten la administración de la información. La Vigilancia Tecnológica y la Inteligencia Competitiva (VT e IC) son dos disciplinas que tienen como objetivo la obtención de información precisa y actualizada. Claramente, la Web es la fuente más grande e importante de información, pero su desestructuración y desorganización requiere de herramientas que ayuden a administrarla. En este trabajo se presenta un modelo para hacer VT e IC utilizando técnicas de Minería de Web. Se describen los módulos del sistema desarrollado y las primeras pruebas realizadas en el ámbito del Clúster del Té de Misiones.

Keywords: minería de Web, vigilancia tecnológica e inteligencia competitiva, toma de decisiones.

1 Introducción

La Toma de decisiones (TD) estratégica requiere del análisis en detalle de una cantidad considerable de información; la cual no se dispone en su totalidad en el momento preciso. Si bien la falta de información para la TD es habitual en las organizaciones, las del área productiva del Té de Misiones son de las más afectadas por diversas razones: los métodos tradicionales de trabajo, la poca capacidad de inversión de las PyMEs, el poco impacto internacional de la producción, etc. Por ello se ha creado el Clúster del Té de Misiones (PyMEs, cooperativas, organismos del estado y universidades), que intenta encontrar soluciones a problemas comunes de sus integrantes. En base a esta unión, se afronta el desafío actual de contar con nuevas formas de competencia, poner a prueba en grado crítico a las organizaciones para posicionarse y articularse con otros actores del entorno y promover el dominio y ejercicio de una capacidad continua de monitoreo o vigilancia del entorno. Para ello, existen dos actividades muy utilizadas para analizar las variaciones del entorno: la Vigilancia Tecnológica (VT) y la Inteligencia Competitiva (IC). Ambas tienen el objetivo de alertar cambios en las tendencias de la tecnología y el comportamiento del mercado. La VT engloba los métodos que intentan anticipar y entender la dirección potencial, las características y efectos de los cambios tecnológicos, especialmente en lo referido a invención, innovación y uso [1]. En la IC intervienen los métodos éticos y legales que tienen como

fin, obtener información valiosa sobre los competidores [2]. Las dos estrategias de captación de datos deben integrarse a los Sistemas de Soporte de Decisiones (SSD), los cuales proveen una serie de técnicas y herramientas que sirven para asistir al decisor durante el proceso de TD [3][4][5].

Se estima que un 80% de la información estratégica no se encuentra estructurada y organizada, donde gran parte de ella está en Internet. La Minería Web (MW) es una metodología de recuperación de información que permite procesar y capturar información útil de páginas Web y documentos en Internet que pueden servir para llevar a cabo procesos de VT e IC, contribuyendo en una de sus etapas principales que es la de búsqueda y recolección de información para su posterior análisis y tratamiento. Básicamente los algoritmos de MW usan técnicas de Minería de Datos para descubrir y extraer información de los documentos y recursos disponibles de la Web [6]. La MW puede dividirse en tres categorías: MW del contenido Web; MW de la estructura de la Web; y MW del uso de la Web [7]. De estas tres, la más relevante para la VT e IC es la primera, la cual abarca el descubrimiento de recursos, categorización y clustering de documentos y extracción de información de páginas Web [8][9][10].

Este artículo describe un prototipo de software basado en técnicas de MW que permita hacer VT e IC en el ámbito del Clúster del Té en Misiones. De esta manera el documento se estructura de la siguiente manera: en la Sección 2 se exponen los principales ejes temáticos del trabajo; en la Sección 3 se describe el modelo desarrollado; los resultados obtenidos se muestran en la Sección 4 y, finalmente, en la Sección 5 se mencionan las conclusiones obtenidas y trabajos futuros.

2 Conceptos preliminares

2.1 Vigilancia Tecnológica e Inteligencia Competitiva

Con la globalización de los mercados y el desarrollo de la era digital, el valor que las empresas atribuyen a la información ha ido cambiando y hoy es más estratégica que nunca. Esto ha hecho que disciplinas como la VT e IC se hayan convertido en un pilar esencial para crear nuevos productos o servicios, para definir estrategias de comercialización, potenciar las capacidades de la organización, mejorar el servicio a los clientes, etc. [11]. Existen muchos métodos de VT los cuales pueden agruparse en las siguientes clases [12][13]: análisis de tendencias; opinión de los expertos [14][15]; métodos de monitoreo e inteligencia [16]; métodos estadísticos; modelado de escenarios; y métodos de modelado y simulación [17].

La IC es una disciplina que se encarga de la obtención, análisis, interpretación y difusión de información del entorno competitivo en el que se mueven las empresas, a través de un proceso sistemático y ético, indispensable para la TD [11][18]. Existen tres etapas [19]: recolección de información, extracción de información y contextualización de la información. Los objetivos del uso de un sistema de información de IC destacan tres aspectos: mejorar la competitividad de la empresa; predecir, con un alto nivel de confianza, la evolución del entorno y proveer un buen soporte para el proceso de decisión estratégico [20].

2.2 Minería de la Web

La cobertura de la información en la Web es muy amplia y diversa; existen datos de todo tipo: tablas estructuradas y semi-estructuradas, páginas Web, textos no estructurados y archivos multimedia. Esto demuestra que existe una amplia heterogeneidad de problemas de índole multi-disciplinaria a ser resueltos con el fin de explotar satisfactoriamente la información Web como recursos valiosos para las empresas. WM se plantea como el conjunto de técnicas que permiten abarcar la problemática de explotación de la información que se encuentra en la Web [21].

El proceso de MW puede ser definido como “el proceso global de descubrir información o conocimiento potencialmente útil y previamente desconocido a través de los datos de la Web” [22]. La MW tiene como objetivo descubrir información útil o el conocimiento tanto del contenido de documentos Web, como también de la estructura de hipervínculos Web y los datos de uso [23].

Una vez que se hallan sitios Web con contenido importante, el sistema debe ser capaz de obtener todo lo relevante utilizando mecanismos de extracción de contenido Web (Web Scraping), y al mismo tiempo adjuntar información de indexación a los registros para mantener una estructura que facilite futuros accesos a los recursos Web de una manera más directa, sin la necesidad de analizar ni ponderar contenido de una página que ya ha sido considerado relevante [21]. Para guiar la navegación de enlaces y localizar de manera eficiente las páginas de destino de gran relevancia, se utilizan técnicas de Web Crawling; donde, cada página que es escaneada es dada a un cliente que guarda las páginas, crea un índice para las páginas o guarda y analiza los contenidos de las páginas [23].

3 Modelo desarrollado

En base a la visión de VT e IC se propone un modelo que cumpla dos objetivos: el primero generar un proceso para dar soporte a la recopilación de los requerimientos de usuario y el segundo generar un proceso de búsqueda continua de recursos en la web. Ambos procesos trabajan coordinadamente en un mismo flujo interactuando constantemente. La idea del primer proceso es orientar al usuario en el armado de varias claves que serán tomadas como punto de partida de la búsqueda y la del segundo es proveer a medida al usuario de recursos que se vayan encontrando. El esquema general del sistema puede observarse en la Fig. 1. El sistema desarrollado posee dos componentes principales el Módulo de Recopilación de Requerimientos de usuario y el Módulo de MW (Fig. 1). Ambos interactúan utilizando un servicio Web con el que se administra el proceso de comunicación. El usuario comienza interactuando con el Módulo de Recopilación de Requerimientos que, mediante una serie de preguntas, lo guía en la construcción de claves de búsqueda. Esas claves se envían al Módulo de MW quien consulta a las interfaces públicas de cuatro buscadores: Google, Bing, Intelligo (un buscador de patentes) y Msmxl Excite (un metabuscador). Las primeras diez URLs de cada uno son categorizadas y ordenadas en una sola lista en base a los requerimientos de usuario (de esa lista se quitan las URLs repetidas). A partir de este punto el Módulo de MW comienza un proceso continuo de inspección de enlaces y

recursos tomando como base cada una de las URLs de la lista. A medida que se encuentran recursos que se consideran relevantes para el usuario, se los coloca en un directorio junto con los metadatos del recurso. Por las características de este proceso continuo (lo que genera un gran volumen de información), se seleccionan solamente cincuenta recursos que se le muestran al usuario para que los valore.

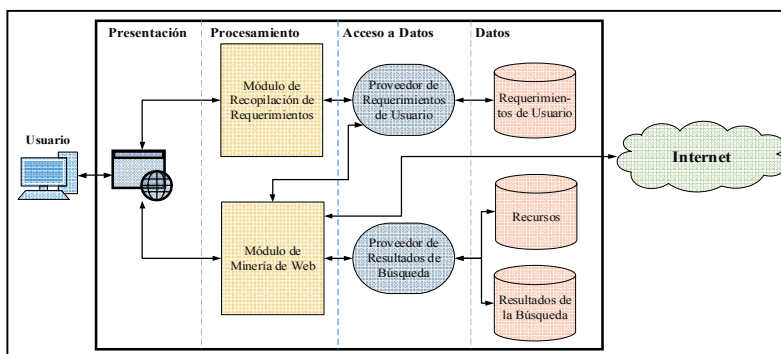


Fig. 1. Esquema general del modelo desarrollado.

3.1 Módulo de Recopilación de Requerimientos

Comúnmente los usuarios tienen dificultades para armar una clave de búsqueda que represente sus requerimientos de información y, por lo general, lo que hacen es introducir una serie de palabras relacionadas con la temática de búsqueda. Si bien estas claves devuelven recursos que pueden ser útiles, no explotan todo el potencial de los buscadores debido a que, generalmente, no se utilizan conectores lógicos ni símbolos para demarcar partes importantes de la clave.

En el modelo propuesto, se utiliza el mecanismo de preguntas guiadas para orientar al usuario en la construcción de las claves de búsqueda descrito en [24]. Inicialmente el usuario determina el tema principal y luego continúa con un proceso de refinamiento mediante preguntas estándares (Fig. 2). Debido a que las claves de búsqueda generadas se usan sobre las mismas interfaces públicas de los buscadores, se requiere utilizar los operadores lógicos: OR, AND y NOT [25][26]. Cabe aclarar que si se desea armar una estructura lógica donde una operación tenga mayor relevancia que otra, se debe estructurar de manera tal que pueda obtenerse un resultado similar a utilizar paréntesis, pero sin indicarlos explícitamente. Otra característica importante es que los navegadores otorgan a los primeros términos de la clave mayor importancia mientras que las siguientes claves refinan el proceso de exploración. Una vez finalizado la recolección de requerimientos, se deben relacionar las respuestas a través de los dos operadores lógicos básicos definidos en el sistema: AND y OR. Estas relaciones se utilizan posteriormente para construir una estructura de árbol donde cada nodo hoja contiene la respuesta provista por el usuario y cada nodo padre contiene el operador lógico que relaciona a estas respuestas entre sí Fig. 3. La secuencia de preguntas se establece por medio de un atributo de relación entre ellas y luego se las pondera [24]. Aquellas preguntas que tengan una ponderación mayor, serán por lo tanto, más importantes y sus respuestas ocuparán los primeros lugares en el conjunto de claves

de búsqueda generadas. Seguidamente, se realiza el proceso de creación del árbol teniendo en cuenta el operador lógico que relaciona a cada pregunta P_i de la lista ordenada con la próxima pregunta P_{i+1} .

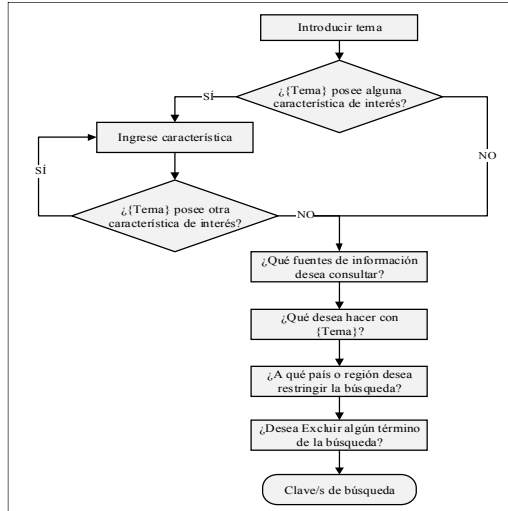


Fig. 2. Proceso de obtención de Requerimientos.

Como resultado, se obtiene una estructura de árbol como se muestra en la Fig. 3. Una respuesta puede estar compuesta por un término o un conjunto de términos separados por espacios. Estos términos se denotan con t_1, t_2, \dots, t_n .

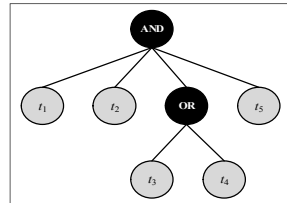


Fig. 3. Árbol de respuestas para un esquema genérico de preguntas.

Finalmente, la clave de búsqueda se define como una cadena de caracteres compuesta por los términos o palabras importantes provistas por el usuario, enlazadas entre sí mediante los operadores lógicos predefinidos. Se han implementado dos opciones para la generación de las claves. La primera construye una clave única que contiene todos los términos proporcionados por el usuario, y la segunda separa cada opción de una respuesta con múltiples opciones en claves diferentes pensadas para ser utilizadas en conjunto en el mismo proceso de búsqueda [24]. De la primera opción se obtiene la siguiente clave sobre el árbol de la Fig. 3:

$$C = t_1 \cdot t_2 \cdot (t_3 + t_4) \cdot t_5 \quad (1)$$

De la segunda opción se obtienen dos claves sobre el mismo árbol:

$$C_1 = t_1 \cdot t_2 \cdot t_3 \cdot t_5 \quad (2)$$

$$C_2 = t_1 \cdot t_2 \cdot t_4 \cdot t_5 \quad (3)$$

esta opción separa los nodos hoja de un operador de disyunción en claves distintas (Ecs. 2 y 3). Por lo tanto, se obtienen N claves, donde N es el producto de la cantidad de nodos hoja subordinados a cada nodo OR considerando el árbol completo (esto provee un mecanismo más adecuado para el inicio del proceso de búsqueda).

3.2 Módulo de Minería de Web

El proceso de minería se inicia cuando el sub-módulo llamado Web Miner (Fig. 4) recibe el conjunto de claves generadas por el Módulo de Recopilación de Requerimientos. Estas claves son enviadas al sub-módulo de Búsqueda que se encarga de obtener URLs proporcionadas por las interfaces públicas de los cuatro motores de búsquedas utilizados (los resultados obtenidos por Google y Bing se obtienen mediante sus APIs, en tanto para Msmlx Excite e Intelligo se obtienen los resultados simulando un procedimiento de búsqueda manual). A partir del conjunto de URLs obtenidas, se genera una lista única filtrada de acuerdo a la posición que ocupan en dichos buscadores y la relación con los requerimientos establecidos por el usuario. Por ejemplo, las URLs de dominio académico (".edu") tendrán más peso si el usuario determinó que quiere artículos o publicaciones relacionadas con el tema de su búsqueda; en cambio, el mayor peso lo tendrán los dominios ".com" si lo que pretende es comprar o vender. Con esta lista ordenada, el sub-módulo de Web Crawler se encarga de comenzar el proceso de descubrimiento. Este proceso es continuo e iterativo a partir de dicha lista.

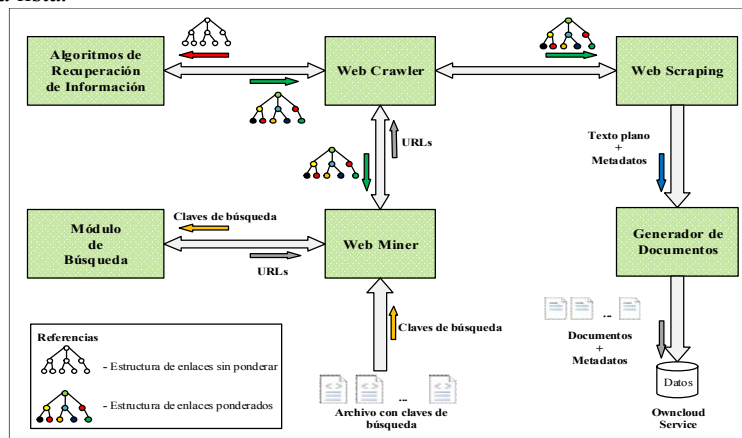


Fig. 4. Arquitectura básica de la aplicación.

Por cada URL se genera un grafo de n URLs descubiertas. Las aristas corresponden a las relaciones entre las URLs. El grafo es enviado al sub-módulo de Algoritmos de Recuperación de Información para establecer una puntuación a cada URL del grafo con el objetivo de obtener una lista ordenada según la relevancia del contenido. El grafo ponderado es enviado al sub-módulo Web Scraping que se encarga de descargar el contenido de las URLs más relevantes y generar los metadatos que contiene información sobre la ponderación realizada y la URL del documento. Esta información se envía al sub-módulo Generador de Documentos que crea los documentos dependiendo de la extensión (.pdf, .html, .asp, .php, etc.) y los asocia con los metadatos para

almacenarlo en un directorio que será sincronizado en todas las ubicaciones que requieran la información (Owncloud Service). Luego, el sub-módulo Web Crawler obtiene la siguiente URL de la lista y se reinicia el proceso.

Cabe destacar que el sub-módulo de Algoritmos de Recuperación de Información no solamente pondera las URLs del grafo recibido en cada iteración, sino que además vuelve a ponderar las URLs de los grafos que fueron procesadas anteriormente. El proceso vuelve a reajustar el peso de todas las URLs para generar una nueva lista de documentos ordenados por la relevancia del contenido. El proceso del sub-módulo Web Crawler finaliza cuando el usuario decide parar con la búsqueda de documentos o cuando no existen más URLs por explorar.

4 Pruebas y Resultados

Con el objetivo de verificar el funcionamiento del sistema desarrollado se han realizado tres pruebas, las dos primeras relacionadas con la búsqueda de información sobre el proceso productivo del té en Argentina, la exportación del mismo, relacionadas con el ámbito de aplicación del sistema; y la tercera, sobre las técnicas de machine learning para hacer análisis de datos para mostrar la extensibilidad del uso del sistema a otros dominios de aplicación. Los resultados que se presentan son los que se obtuvieron luego de 12 horas de iniciado el proceso.

4.1 Prueba 1: Proceso productivo del té

Esta primera prueba consistió en obtener recursos web respecto al proceso de producción del té. Para ello se utilizó el módulo de recopilación de requerimientos con las siguientes consignas: *proceso productivo del té, optimización, patentes, artículo, cita, reducción de costos, automatización y maquinaria, Argentina*, excluyendo *publicidad*. Puede observarse en la Tabla 1 que la gran mayoría de recursos obtenidos son de sitios académicos, especialmente presentaciones o slides que explican el proceso de producción del té.

Tabla 1. Porcentajes de sitios y recursos obtenidos para proceso productivo del té.

Tipo de sitios	Porcentaje obtenido	Tipo de recurso	Porcentaje obtenido
Comerciales	05%	Artículos de revista	24%
Gubernamentales	24%	Patentes	50%
Académicos, educativos	60%	Slides	10%
Otros	11%	Información Comercial	16%

Otro aspecto interesante es que los sitios gubernamentales de la lista corresponden en su gran mayoría al Instituto Nacional de Tecnología Agropecuaria (INTA) de Argentina donde se explican otros procesos productivos como el de la yerba mate o el algodón y, del mismo ámbito del INTA, también se ha encontrado en la lista un sitio de maquinaria agrícola y valor agregado a productos agrícolas.

4.2 Prueba 2: Exportación de té

En la segunda prueba se han utilizado las siguientes consignas para buscar información sobre exportación de té: *argentinian tea exportation, international trade,*

traditional tea y *potential markets*, excluyendo *advertising*. Los resultados se de los recursos obtenidos se pueden apreciar en la Tabla 2.

Tabla 2. Porcentajes de sitios y recursos obtenidos para exportación de té.

Tipo de sitios	Porcentaje obtenido	Tipo de recurso	Porcentaje obtenido
Comerciales	88%	Artículos de revista	06%
Gubernamentales	04%	Patentes	00%
Académicos, educativos	08%	Slides	00%
Otros	00%	Información Comercial	94%

El primer aspecto que se destaca en los resultados de la Tabla 2 es el elevado porcentaje de sitios comerciales encontrados. Entre ellos hay una gran variedad de dominios que incluyen información estadística de exportación de productos (no solamente té) de los distintos países. Uno de los sitios de la lista (theteadetective.com) aparece varias veces con distintas páginas ([TeasOfArgentina](#), [TeasOfTheWorld](#), [BlendedScentedTeasDiffer](#), [IndiaTeaGrowsAssam](#), etc), al revisar el enlace se descubrió que esta página tiene gran cantidad de información sobre los distintos aspectos acerca de la historia, producción, elaboración y comercialización del té.

4.3 Prueba 3: Machine Learning para análisis de datos

Esta prueba tuvo como objetivo revisar la potencial extensión del uso del sistema a otros dominios de aplicación. En este caso se optó por buscar información sobre algoritmos de machine learning que se utilizan para análisis de datos. Para ello se trabajó con las siguientes consignas: *data analytics*, *big data*, *paper*, *cite*, *trends* y *statistics* excluyendo *advertising*. En este caso se intentó realizar una búsqueda de recursos netamente académicos y los resultados pueden observarse en la Tabla 3.

Tabla 3. Porcentajes de sitios y recursos obtenidos para machine learning.

Tipo de sitios	Porcentaje obtenido	Tipo de recurso	Porcentaje obtenido
Comerciales	00%	Artículos de revista	86%
Gubernamentales	00%	Patentes	00%
Académicos, educativos	92%	Slides	14%
Otros	08%	Información Comercial	00%

Los resultados mostrados en la Tabla 3 indican que casi todos los sitios son académicos. Entre ellos se destacan enlaces a artículos de distintas editoriales como Springer, Elsevier y sitios de manejo de información académica como Mendeley. El resto de los sitios encontrados contienen también información académica, pero en forma de blogs personales y de organizaciones como el caso de <http://squad.compmath.fr/> o <http://deeplearning4j.org/>, ambas orientadas al desarrollo de algoritmos de machine learning bajo distintos lenguajes de programación. También, como punto destacable, entre los resultados aparece el sitio del ICML 2015, que corresponde uno de los congresos sobre machine learning más prestigiosos. Finalmente, es de destacar que en esta prueba se ha destacado la gran cantidad de resultados potencialmente utilizables respecto a la búsqueda planteada.

5 Conclusiones y Trabajos Futuros

En este artículo se ha presentado un sistema que sirve de herramienta para actividades de Vigilancia Tecnológica e Inteligencia Competitiva. El sistema cuenta con dos módulos principales, el primero sirve para guiar al usuario en la recopilación de requerimientos para el proceso de búsqueda generando claves de búsquedas más completas que las que habitualmente se utilizan en los buscadores. El segundo módulo se encarga de la minería de Web, explorando enlaces a partir de URLs devueltas por las interfaces públicas de los buscadores más utilizados.

Un aspecto de relevancia del modelo propuesto es que no hace una búsqueda tradicional de consulta-respuesta, sino que los resultados que se obtienen son refinados continuamente y categorizados en función a los requerimientos de usuario. Esto significa que, una vez iniciado el proceso de búsqueda por parte del usuario, el mismo sigue ejecutando obteniendo nuevos resultados hasta que el usuario decida finalizarlo.

Si bien el análisis realizado en este artículo ha sido cuantitativo debido a que se han presentado las primeras experiencias realizadas con el sistema, las pruebas muestran que el sistema encuentra información acorde al tipo de respuesta esperada. En este sentido, las tres pruebas tenían objetivos distintos y los recursos obtenidos por el sistema coinciden con dichos objetivos.

Cabe mencionar que si bien se ha comenzado a probar el sistema en base a los requerimientos del Clúster del Té de Misiones como dominio de aplicación, las pruebas realizadas muestran que su uso puede ser extendido a otros dominios. Cabe mencionar que si bien se ha comenzado a probar el sistema en base a los requerimientos del Clúster del Té de Misiones como dominio de aplicación, las pruebas realizadas muestran que su uso puede ser extendido a otros dominios.

Actualmente se encuentra en desarrollo el proceso de ajuste de los algoritmos y el tratamiento e indexación de resultados. El primero permitirá mejorar la performance del sistema y el segundo facilitará la visualización de los resultados y el feedback del usuario. Además, se está trabajando en la aplicación de métricas de evaluación (“precision” y “recall”) para medir la recuperación de información.

Agradecimientos

A los miembros del Clúster del té por facilitar el dominio de aplicación. Este trabajo forma parte de los proyectos “Modelos de Análisis de Información para la Toma de Decisiones Estratégicas” (UGD) y “Análisis de Información en Grandes Volúmenes de Datos Orientado al Proceso de Toma de Decisiones Estratégicas” (cód: UTN4058).

Referencias

- [1] A. Firat, W. Woon, and S. Madnick, “Technological Forecasting – A Review,” presented at the Working Paper CISL# 2008-15, Cambridge, 2008.
- [2] R. G. Vedder, M. T. Vanecek, C. Guynes, and J. Cappel, “CEO and CIO Perspectives on Competitive Intelligence,” *Commun. ACM*, vol. 42, no. 8, 1999.
- [3] J. P. Shim and others authors; “Past, Present, and Future of Decision Support Technology,” *Decis. Support Syst.*, vol. 33, no. 2, pp. 111–126, 2002.
- [4] C. W. Holsapple, “Decision Support Systems,” *Encycl. Inf. Syst.*, vol. 1, pp. 551–565, 2003.

- [5] D. J. Power and R. Sharda, "Decision support systems," Springer Handbook of Automation, pp. 1539–1548, 2009.
- [6] O. Etzioni, "The World Wide WEB: Quagmire or gold mine," Commun. ACM, vol. 39, no. 11, pp. 65–68, 1996.
- [7] R. Kosala and H. Blokeel, "WEB mining research: A survey," ACM SIGKDD Explor., vol. 2, no. 1, pp. 1–15, 2000.
- [8] S. Chakrabarti, M. van der Berg, and B. Dom, "Focused crawling: A new approach to topic-specific WEB re-source discovery," in In Proceedings of the 8th international conference on World Wide WEB (WWW8), pp. 545–562, 1999.
- [9] T. Kohonen, and others authors "Selforganization of a massive document collection," IEEE Trans. Neural Netw., vol. 11, no. 3, pp. 574–585, 2000.
- [10] C.H. Chang, C.N. Hsu, and S.C. Lui; "Automatic information extraction from semi-structured WEB pages by pattern discovery," Decis. Support Syst., vol. 35, pp. 129–147, 2003.
- [11] R. Barainka; "Modelos de Vigilancia Tecnológica e Inteligencia Competitiva". Servicio Zaintek de BAI. 2006.
- [12] V. Coates and others authors; On the future of technological forecasting, Technol. Forecast. Soc. Change, vol. 67, no. 1, pp.1-17, 2001.
- [13] T.J.Gordon and J.C. Glenn, Futures research methodology, Version 2.0 Millennium Project of the American Council for the United Nations University, 2003.
- [14] F. Parente and others authors; An examination of factors contributing to Delphi accuracy, J. Forecast; vol. 3, pp.173 – 182, 1984.
- [15] R. Levary and D. Han; Choosing a technological forecasting method. Industrial Management, vol.37, no. 1, pp.14, 1995.
- [16] S. Madnick and W.L. Woon; Technology Forecasting Using Data Mining and Semantics, MIT/MIST Collaborative Research, 2009.
- [17] T.J. Gordon; "A simple agent model of an epidemic" Technological Forecasting & Social Change, vol. 70, pp. 397-417, 2003.
- [18] Escorsa Pere; "INTEC, la Inteligencia Competitiva. Factor clave para la toma de decisiones estratégicas en las organizaciones", Plan Regional de Ciencia y Tecnología de la Comunidad de Madrid, 2008.
- [19] T. Hiltbrand; "Learning Competitive Intelligence From a Bunch of Screwballs", Business Intelligence Journal, vol: 15, no: 4, 2010.
- [20] I. Popa Anica and G. Cucui, "A Framework for Enhancing Competitive Intelligence Capabilities using Decision Support System based on Web Mining Techniques", Int. J. of Computers, Communications & Control, vol. 4, no. 4, pp. 326-334, 2009.
- [21] L. Bing; "Web Data Mining, Exploring Hyperlinks, Contents, and Usage Data", First Edition. University of Illinois, Chicago. 2007.
- [22] V. Heughes Jeria Escobar; "Minería Web de Uso y Perfiles de Usuario: Aplicaciones con Lógica Difusa", Universidad de Granada. 2007.
- [23] F. Menczer, G. Pant and P. Srinivasan; "Topical Web Crawlers: Evaluating Adaptive Algorithms". ACM Transactions on Internet Technology, vol. 5. 2004.
- [24] L. Lezcano and others authors; "Modelo de Recopilación de Requerimientos para Vigilancia Tecnológica e Inteligencia Competitiva (VTeIC)" 3er CONAIISI | 19 y 20 de Noviembre 2015. UTN, Facultad Regional Buenos Aires.
- [25] Support.google.com, 'Search operators - Search Help', Disponible en: <https://support.google.com/websearch/answer/2466433>, [último acceso: 05/07/2016].
- [26] Msdn.microsoft.com, 'Advanced Operator Reference', Disponible en: <https://msdn.microsoft.com/en-us/library/ff795620.aspx>, [último acceso: 06/07/2016].

Comparison of Different Approaches for Adapting Mutation Probabilities in Genetic Algorithms

Natalia Stark¹, Gabriela Minetti¹, Carolina Salto^{1,2}

¹Laboratorio de Investigación en Sistemas Inteligentes
Facultad de Ingeniería – UNLPam
Calle 110 N° 390 – General Pico, La Pampa
²CONICET
e-mail { nstark, minettig, saltoc }@ing.unlpam.edu.ar

Abstract- Traditionally in Genetic Algorithms, the mutation probability parameter maintains a constant value during the search. However, an important difficulty is to determine a priori which probability value is the best suited for a given problem. In this paper we compare three different adaptive algorithms that include strategies to modify the mutation probability without external control. One adaptive strategy uses the genetic diversity present in the population to update the mutation probability. Other strategy is based on the ideas of reinforcement learning and the last one varies the probabilities of mutation depending on the fitness values of the solution. All these strategies eliminate a very expensive computational phase related to the pre-tuning of the algorithmic parameters. The empirical comparisons show that if the genetic algorithm uses the genetic diversity, as the strategy for adapting the mutation probability outperforms the other two strategies.

1 Introduction

Adaptation is an alternative to determine the optimal value of the parameters to use in a genetic algorithm (GA) during the search. This is a good option, because during the design process, we need to instantiate several parameters to specific values before the run of the algorithm, in order to obtain a robust search technique. Even for an expert user, the parameter configuration for an optimal performance is hard to find.

The parameter control [9] manages information that influences the parameter values during the search and defines a set of criteria to produce changes. In this way, the user does not need to make non-trivial decisions beforehand, which are a very expensive task. Many relevant approaches have been proposed for adjusting GA parameters such as genetic operator probabilities [14,17,18,19], selection methods [5, 11], population size [4, 10, 21,23] during the search. The mutation probability (p_m) is one of the critical parameters in the GA performance. Large values of p_m transform the GA into a purely random search algorithm, while some mutation is required to prevent the premature convergence of the GA to suboptimal solutions.

In this work, we compare the adaptive method proposed by Stark et al. [22] with other two different methods, present in the literature, to adapt the p_m during the GA run. The aim of this comparison is to determine which of them allows to obtain good

quality of results with a low computational effort. In the method proposed by Stark et al. [22], the mutation probability varies depending on the genetic diversity present in the population, by using a control strategy based on the current population entropy. In the comparison the adaptive method proposed by Srinivas et al.[17] is considered, where the probabilities of mutation are changed depending on the fitness values of the solution. This proposal protects high fitness solutions, but solutions with subaverage fitness are totally disrupted. These methods have been selected since they present different ways to vary the p_m and also exhibit good performance. The other method included in the comparison is the one proposed by Riff et al. [14], which is based on the ideas of reinforcement learning: an operator receives a reward in its probability of application when the new generated individual is better than its parents. Analogously, it receives a penalty when the offspring has a worse fitness value than its parents. Both the rewards and the penalties strongly depend on the improvement/degradation of the evaluation function value.

The outline of the paper is as follows. Section 2 details the adaptive algorithms included in this work and a brief comparison among them. Section 3 addresses the optimization problem and the parameter configuration. In the next section, we analyze the results from a computational point of view. Finally, we summarize the conclusions and discuss several lines for future research in Section 5.

2 Adaptive GAs

In this section, we present the characteristics of the adaptive algorithm proposed in [22], denominated Adaptive Mutation Probability Genetic Algorithms (APmGA), which dynamically adjusts the mutation probability (p_m) during the evolutionary process. To assess their behavior we considered two other algorithms from the literature, which also include adaptive control strategies to change p_m . They are: the Adaptive Genetic Algorithms (AGA) proposed by Srinivas, et al. [17] and the Adaptive Control strategy (AcGA) proposed by Riff et al. [14]. These algorithms are the most representative in this area, showing a good performance.

All adaptive control strategies considered can be embedded as a procedure control into a traditional GA, thus not requiring any change of the algorithm itself. The generic structure of the GA used in this work is shown in Algorithm 1. The algorithm creates an initial population P of μ solutions in a random way, and then evaluates these solutions. The population goes into a cycle where it evolves by means of the operator applications such as selection, recombination, and mutation to compute a whole generation of new λ individuals. The new population is built up by selecting μ individuals from the set of $(\mu+\lambda)$ existing ones. After that, each iteration ends with the computation of the statistics of the current population and the application of the adaptive mechanism to dynamically set the mutation probability. Finally, the best solution is identified as the best individual ever found that maximizes the fitness function.

Algorithm 1 Traditional GA()

```
t = 0; {current generation}
initialize(P(t));
evaluate(P(t));
while{(t < maxgenerations) do
    P'(t) = selection(P(t));
    P'(t) = recombine(P'(t));
    P'(t) = mutate(P'(t), pm);
    evaluate (P'(t));
    P(t+1) = Replace(P'(t) U P(t));
    statistics(P(t+1));
    AdaptivePmControl(pm);
    t = t + 1;
end while
return best_solution
```

2.1 APmGA

The objective of the adaptive strategy of APmGA is to increase p_m if the genetic diversity is gradually lost in order to maintain a population distributed in the search space. Otherwise, the p_m value is reduced when an increase in the population diversity is observed. Therefore, these changes in the p_m value are also an additional source for a good balance between exploration and exploitation, intrinsic to the adaptive criterion. Furthermore, this strategy monitors the genotypic diversity present in the population using the Shannon entropy metric [20], following the ideas presented in [2]. When the entropy value is closed to 0, the population contains identical individuals; otherwise it is positive and maximal when all the individuals in the population are different.

The control strategy calculates the population entropy at the end of each *epoch* (a certain number of consecutive generations). Then, it computes the variation of the current entropy at the k epoch (H_k) respect the entropy from the previous $k-1$ epoch (H_{k-1}), denoted as $\Delta H_k = H_k - H_{k-1}$. This variation, ΔH_k , is compared with the one observed in the previous epoch ($\Delta H_{k-1} = H_{k-1} - H_{k-2}$), as shown in the Algorithm 2. In this way, if the entropy variation decreases at least in a factor of ε , this indicates the lost of diversity, in a consequence the mutation probability value will be changed by adding a constant factor (α). Otherwise, the p_m value is decreased by subtracting an α value. In order to prevent the overshooting of p_m , the strategy controls that p_m belongs to $[p_m\text{LB}, p_m\text{UB}]$; where the lower bound of p_m is $p_m\text{LB} = 0.001$ and the upper bound is $p_m\text{UB} = 0.1$. These bounds are selected considering the values suggested in literature. We consider the factor α as 0.001 and ε as 0.5. These values have been “optimized” through a previous hand-tuning process comparing it versus other different values.

Algorithm 2 AdaptivePmControl($\Delta H_k, \Delta H_{k-1}, p_m$) function

```
if  $\Delta H_k < (1+\varepsilon) \cdot \Delta H_{k-1}$  then
     $p_m = p_m + \alpha$ 
else
     $p_m = p_m - \alpha$ 
```

2.2 AGA

Srinivas et al. present an adaptive strategy to control the p_m in order to prevent the GA to be stuck at a local optimum. The value of p_m varies depending on the value of $f_{max} - \bar{f}$ (where \bar{f} is the average fitness value of the population and f_{max} the maximum fitness value of the population) and on the fitness value f of a solution. If f is close to f_{max} , p_m should be set to a smaller value in order to not disrupt good solutions. Consequently, they proposed to use different p_m values depending on the solution quality, as shown in Algorithm 3. For the solution with the maximum fitness, p_m is zero. For a solution with $f = \bar{f}$, $p_m = k$. For solutions with sub average fitness value ($f < \bar{f}$), p_m might assume values larges to 1.0, so the authors impose the following constraint: $p_m = k$ if $f < \bar{f}$, where $k \leq 1.0$. The k value has to be less than 1.0 to constrain p_m to the range $[0.0, 1.0]$ [17]. In particular, we have set k equal to 0.5, taking into the account the value used in the original work.

2.3 AcGA

This strategy computes the mutation probability value for the next generation (pm_{t+1}) considering the performance of the operator, which is measured by the average of the improvements/degradations ($\overline{Sa(O)}$) obtained after its application considering the last l generations. The improvements/degradations are considered as a proportional factor depending on the maximum improvement (max_imp) or degradation (max_deg). The success measure for an operator ($Sa(O)$) in its i^{th} application is computed as the difference between the fitness of the generated child and the average fitness of its parents. This procedure is shown in Algorithm 4, where r is included for providing smooth parameter adjustments, as it is required for any effective dynamic parameter control strategy [14]. This value is fixed at 0.5.

2.4 Comparison between the adaptive GA approaches

Both APmGA and AcGA use control strategies at a population level. Thus, the mutation probability values are the same for all of the individuals in the current

Algorithm 3: Adaptive Genetic Algorithms (f_{max}, \bar{f}, pm) function

```

if  $f \geq \bar{f}$  then
     $pm = \frac{k(f_{max} - f)}{f_{max} - \bar{f}}$ 
else
     $pm = k$ 

```

Algorithm 4: AdaptiveControl($\overline{Sa(O)}$, max_imp, max_deg, Pm_t) function

```

if  $\overline{Sa(Om)} \geq 0$  then
     $pm_{t+1} = pm_t + r * \frac{\overline{Sa(Om)}}{max\_imp}$ 
else
     $pm_{t+1} = pm_t + r * \frac{\overline{Sa(Om)}}{max\_deg}$ 

```

population. In the case of AGA, a different p_m is computed for each solution depending on its fitness value. Another difference is the way to update the p_m : AcGA and AGA consider the fitness values of individuals or population, but ApmGA uses the genetic diversity of the population.

These methods can be implemented without adding any significant overhead and without introducing any major changes to its original algorithm design. The decisions made during the search are based on the current information available from a monitoring process, allowing the algorithm to trigger changes when deemed necessary, based on the potential advantages that such changes could bring.

3 Experimental Setup

In this section we present the necessary information to reproduce the experiments that have been carried out in this work. First, we will introduce the problem used to assess the performance of our proposals: the NK landscape, which has been the center of several theoretical and empirical studies both for the statistical properties of the generated landscapes and for their GA-hardness [1, 8]. Second, we will justify the parameters that the adaptive GAs will use.

3.1 NK-Landscapes

An NK-landscape [12] is a fitness function $f: \{0,1\}^N \rightarrow R$ on binary strings, where N is the bit string length and K is the number of bits in the string that epistatically interact with each bit, i.e., K stands for the number of other genes that epistatically affect the contribution of each gene to the overall fitness value of the string. Each gene x_i , where $1 \leq x_i \leq N$, contributes to the total fitness of the genotype depending on the value of its allele and on those of each of the K other genes to which it is linked. Thus K must fall between 0 and $N-1$. For $K = 0$, there are no interaction among genes and a single-peak landscape is obtained; in the other extreme (for $K = N-1$), all genes interact each other in constructing the fitness landscape, so a completely random landscape is obtained (a maximally rugged landscape). Varying K from 0 to $N-1$ gives a family of increasingly rugged multi-peaked landscapes.

The fitness value for the entire genotype is the average of the fitness contribution of each locus f_i by Equation 1:

$$f(x) = \frac{1}{N} \sum_{i=1}^N f_i(x_i, x_{i_1}, \dots, x_{i_k}) \quad (1)$$

where $\{x_{i_1}, \dots, x_{i_k}\} \subseteq \{x_1, \dots, x_{i-1}, x_{i+1}, \dots, N\}$ are the K genes interacting with gene x_i in the genotype x . The other K epistatic genes could be chosen in any number of ways from the N genes in the genotype. Kauffman [13] investigated two possibilities: *adjacent neighbourhoods*, where K genes nearest to gene x_i on the chromosome are chosen, particularly a gene interacts with $K/2$ left and $K/2$ right adjacent genes; and *random neighbourhoods*, where these K other genes are chosen randomly on the chromosome. In this work, we adopted the first type of neighbourhood and considered circular genotypes to avoid boundary effects.

3.2 Parameters

The parameter setting applied to the GAs is the following. The whole population (μ) is composed of 32 individuals. By default, the initial population is randomly generated. In each iteration, the number of created offsprings (λ) is 64. The maximum number of generations is fixed at 10000. Each parent is selected by binary tournament selection. The uniform crossover operator is applied with a probability of 0.65. Bit-flip mutation is used. Fitness proportional selection and elitism are used to build up the next population from the set of $(\mu + \lambda)$ individuals. These parameters (population size, stop criterion, probabilities, etc.) are chosen after an examination of some values previously used with success in [1]. The initial probability mutation is $1/N$ for all algorithms. In Table 1, we summarize the whole setting of parameters values used in these experiments.

We conduct our study on NK instances with $N=96$ bits varying the epistatic relations from $K=0$ to $K=48$ in increments of 4, adding up to 13 instances. We use landscapes with adjacent epistatic patterns among genes. For each combination of N and K we have generated 30 random problem instances.

Due to the stochastic nature of the algorithms, the final results are obtained after averaging the running times of 30 independent runs. A statistical analysis has been performed in order to provide the results with statistical confidence and, therefore, obtain meaningful conclusions. We use the non-parametric Kruskal-Wallis test, to distinguish meaningful differences between the mean results of all algorithms. We have considered a level of significance of $\alpha = 0.01$, in order to indicate a 99% confidence level in the results.

The algorithms are implemented inside MALLBA [3], a C++ software library fostering rapid prototyping of hybrid and parallel algorithms. The strategy control proposed by Riff [14] and Srinivas et al. [17] were implemented by replicating the ideas and configurations proposed in their works. Our computing system is an Intel CI7 2600 at 3.40GHZ and 4 G RAM, under SUSE Linux with 3.1.0-1.2 kernel version.

4 Experimental Results

In this section, we analyze the performance of the three adaptive GA considered in the comparison: APmGA [22], AGA [17] and AcGA [14]. Our aim is to offer an analysis from the accuracy (quality) and performance points of view. For this

Table 1: Parametric values used for the different GAs

<i>Parameter</i>	<i>Value</i>
Parents selection	Binary tournament
μ	32
λ	64
Crossover Operator	UX
Crossover Probability	0.65
Mutation Operator	Bit-Flip
Replacement Selection	The best $(\mu + \lambda)$
Stop conditions	10.000 generations

purpose, this section consists of two parts. First, we analyze the solution quality obtained by the studied algorithms. In the second part, we study the computational effort of each algorithm.

Table 2 presents the solution quality that is measured by the percentage gap, i.e., the relative distance to the best solution obtained by APmGA ($best_sol_{APmGA}$) and the best solution of each of the other algorithms, as described in Equation 2.

$$gap_{best} = \frac{best_sol_{APmGA} - GA_i}{best_sol_{APmGA}} \times 100 \quad (2)$$

From the results, APmGA is better than the other two adaptive GAs, from low to high levels of epistasis, which suggests that APmGA obtains solutions with higher fitness than the other two approaches. The non-parametric tests suggest that there are statistical differences between the APmGA and the rest of the algorithms (p -value less than $2.2e-16$). Boxplot summaries for each algorithm can be found in Figure 1, where each boxplot contains the best found solutions from 30 different runs for all instances. APmGA obtains a median value lower than the rest of the algorithm with an interquartile amplitude smaller, indicating that the best solutions are not disperse (homogeneity in the data set). Previous observations suggest that the adaptive strategy used by APmGA improves the behaviour of the GA, avoiding the premature convergence and the lost of genetic diversity.

Finally, Figure 2 shows boxplots corresponding to the evaluations to obtain the best solution for the three adaptive approaches. AGA is the algorithm with the highest numerical effort, in order to find their best solutions. The differences of the computational effort between AcGA and APmGA are not statically significant. Considering these results and the improvement in the solution quality shown by the APmGA, we can conclude that this algorithm is the best strategy to adapt the mutation probability.

Table 2: Average values of the best found solution for each instance (the best values are bolded).

Inst	APmGA	AGA	AcGA
96-0	0,04	0,08	0,04
96-4	0,04	0,12	0,07
96-8	0,06	0,15	0,08
96-12	0,04	0,12	0,06
96-16	0,06	0,11	0,07
96-20	0,03	0,09	0,05
96-24	0,05	0,09	0,06
96-28	0,05	0,07	0,05
96-32	0,04	0,07	0,05
96-36	0,06	0,08	0,06
96-40	0,03	0,04	0,04
96-44	0,05	0,06	0,05

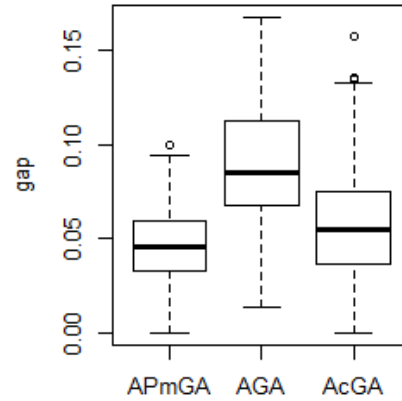


Fig. 1. Boxplot of the gap values for each algorithm

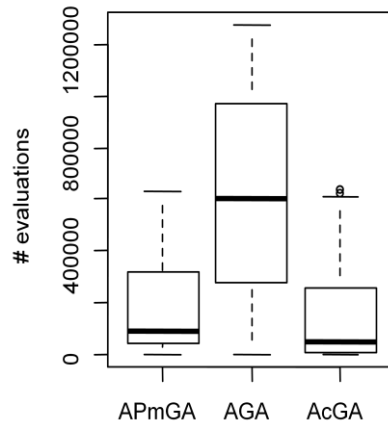


Fig. 2. Boxplot of the evaluations needed to obtain the best solution for each algorithm

Summarizing, the best results are obtained when the mutation probability adaptation is based on the genotypic diversity, as implemented in APmGA. The application of this strategy allows to increase or to keep the population diversity according to the entropy value. Furthermore, no increments in the computational effort are observed when this algorithm is run.

5 Conclusions

In this paper, we analyzed different adaptive evolutionary algorithms which add a control strategy to modify the mutation probability value during the evolution without external control. The control strategies considered differ on the information of the search considered to update the probability value. The different proposals reduce the pre-tuning time to determinate the most appropriate mutation probability value to enhance the algorithm performance.

The results obtained by APmGA compared with AcGA and AGA are very encouraging from low to high levels of epistasis, since it can obtain high quality solutions with a competitive computational effort. Thus, the control strategy using the genotypic diversity as the base to introduce changes in the mutation probability outperforms the other two using the fitness solution.

As a future work, we propose to include the recombination probability in this process. Furthermore, another extension to this work should be the analysis of the effectiveness of this adaptive approach over other complex problems.

Acknowledgements

We acknowledge the Universidad Nacional de La Pampa and the ANPCYT (Argentina) from which the authors receive continuous support. C. Salto thanks CONICET of Argentina.

References

- [1] H. Aguirre and K. Tanaka. Genetic algorithms on nk-landscapes: Effects of selection, drift, mutation, and recombination, in *EvoWorkshops 2003*, LNCS 2611, pp. 131–142, 2003.

- [2] E. Alba and B. Dorronsoro. The Exploration/Exploitation Tradeoff in Dynamic Cellular Genetic Algorithms, in *Proceedings of IEEE Transactions on Evolutionary Computation*, vol. 9(2), pp. 126-142, 2005.
- [3] E. Alba, F. Almeida, M.J. Blesa, J. Cabeza, C. Cotta, M. Díaz, I. Dorta, J. Gabarró, C. León, J. Luna, L. Moreno, C. Pablos, J. Petit, A. Rojas, F. Xhafa. MALLBA: A Library of Skeletons for Combinatorial Optimisation”, in *Proceedings of the 8th International Euro-Par Conference on Parallel*, pp. 927–932, 2002.
- [4] J. Arabas, Z. Michalewicz, and J. Mulawka. GAVaPS A genetic algorithm with varying population size, in *Proceedings of the First IEEE Conference on Evolutionary Computation*, pp. 73-78, 1994.
- [5] J. E. Baker. Adaptive selection methods for genetic algorithms, in *Proceedings of the 1st International Conference on Genetic Algorithms and Their Applications*, pp. 101–111, 1985.
- [6] M. Birattari, T. Stutzle, L. Paquete, and K. Varrenttrapp. A racing algorithm for configuring metaheuristics, in *Proceedings of the 2002 Genetic and Evolutionary Computation Conference*, pp. 11–19, 2002.
- [7] K. A. De Jong. An analysis of the behavior of a class of genetic adaptive systems, Ph.D. dissertation, Univ. of Michigan, Ann Arbor, MI, 1975.
- [8] K.A De Jong, M.A. Potter, and W.M. Spears. Using problem generators to explore the effects of epistasis, in *Proceedings of 7th International Conference on Genetic Algorithms*, pp. 338-345, 1997.
- [9] A. E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter Control in Evolutionary Algorithms, in *IEEE Transactions on Evolutionary Computation.*, vol. 3(2), pp. 124–141, 1999.
- [10] A.E. Eiben E. Marchiori, and V.A. Valko. Evolutionary Algorithms with on-the-fly pop size adjustment, in *Proceedings of Parallel Problem Solving from Nature 2004 (PPSN 2004)*, vol. 3242 of LNCS , pp. 41-50, 2004.
- [11] A.E. Eiben, M.C. Schut, and A.R. de Wilde. Boosting Genetic Algorithms with Self-Adaptive Selection, in *Proceedings of the 2006 IEEE Congress on Evolutionary Computation*, pp. 16-21, 2006.
- [12] S. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, 1993.
- [13] S. A. Kauffman and S. Levin. Towards a general theory of adaptive walks on rugged landscapes, in *Journal of Theoretical Biology*, vol. 128, pp. 11-45, 1987.
- [14] E. Montero and M-C Riff. On-the-fly calibrating strategies for evolutionary algorithms, in *Information Sciences*, vol. 181(3), pp. 552-566, 2011.
- [15] V. Nannen and A. Eiben, Relevance estimation and value calibration of evolutionary algorithm parameters, in *Proceedings of the Joint International Conference for Artificial Intelligence (IJCAI)*, pp. 975–980, 2006.
- [16] E. Ridge and D. Kudenko. Tuning an algorithm using design of experiments, in *Experimental Methods for the Analysis of Optimization Algorithms*, pp. 265 –286, 2010.
- [17] M. Srinivas and L.M. Patnaik, Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms, in *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24 (4), pp. 656-667, 1994.
- [18] J.E. Smith and T.C. Fogarty. Operator and parameter adaptation in genetic algorithms, in *Soft Computing*, vol. 1(2), pp. 81–87, 1997.
- [19] D. Thierens. Adaptive strategies for operator allocation, in *Parameter Setting in Evolutionary Algorithms*, Springer, pp. 77–90, 2007.
- [20] C.E. Shannon. A mathematical theory of communication, in *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.
- [21] X.H. Shi, L.M. Wan, H. P. Lee, X. W. Yang, L. M. Wang, and Y. C. Liang. An Improved Genetic Algorithm with Variable Population-Size and a PSO-GA Based Hybrid Evolutionary Algorithm, in *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, pp. 1735-1740, 2003.

- [22] N. Stark, G. Minetti, and C. Salto. A New Strategy for Adapting the Mutation Probability in Genetic Algorithms, in Proceedings of the Congreso Argentino de Ciencias de la Computación (CACIC 2012), pp. 51-59, 2012.
- [23] D. Voosen and H. Mühlhenbein. Adaptation population size by competing subpopulations, in Proceedings of the 1996 IEEE Conference on Evolutionary Computation, pp. 330-335, 1996.

Introducing a Context-Aware Scheme in an intelligent reasoning process

Paola D. Budán^{1,4} Maximiliano C. D. Budán^{1,2,5}, Maria Vanina Martinez^{1,2,3}
and Guillermo R. Simari^{1,3}

1 - Instituto de Ciencias e Ingeniería de la Computación (UNS-CONICET)

2 - Concejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

3 - Dpto. de Ciencias e Ingeniería de la Computación

Universidad Nacional de Sur (UNS)

4 - Dpto. Informática, Universidad Nacional de Santiago del Estero (UNSE)

5 - Dpto. Matemática, Universidad Nacional de Santiago del Estero (UNSE)

`pbudan@unse.edu.ar` - `{mcd, mvm, grs}@cs.uns.edu.ar`

Abstract. Several investigations have been developed around analogies based reasoning in different domains, however the analogy between arguments has not been deeply explored. A semiformal way to express these patterns of reasoning were proposed by Walton, through argument schemes from analogy. From this, it is possible to propose computable approximations for comparing arguments. In this paper we introduce a formalism based on the comparison of arguments through *descriptors* or labels which describes an aspect that the argument refers to. This formalism allows us classifying similar arguments considering the natural descriptors of them, in a specific context.

Keywords: Analogies Based Reasoning, Argumentation Scheme, Descriptors, Context Domain.

1 Introduction

The Argument from Analogy [30, 19] represents a very common form of everyday human reasoning. Briefly speaking, two cases are analyzed for similarities and differences between them, using a form of inductive inference where the similarities between the cases lead to postulate a further not yet confirmed similarity. The argumentation from analogy allows to solve a new case based on already solved cases. Numerous investigations have been developed around analogies based reasoning, which are based on establishing the similarities and differences between a known object -that is part of knowledge base of agents- and another new but which has certain known aspects [20, 26]. In multi-agents systems domain, when an intelligent agents need to solve a certain problem or take a decision, it is common for them to seek into their knowledge base for solutions from previous similar problems, where the context information fits both the new and the known problem. This reasoning process is guided by thought patterns and involves an argumentation process, *i.e.*, a process by which reasons are given in favour of a particular conclusion.

The analogy between arguments has not been deeply explored. Any comparison process requires the definition of the context in which such comparison acquires significance. Meanwhile, as the field of Artificial Intelligence (AI) carries out research in computational argumentation to achieve useful systems based on common sense, it seems desirable and reasonable to try to formalize the analogy between arguments. In general, this context is formed by conditions or variables that govern the comparison process, all of which have different importance degrees and need to be ordered to carry out the process in an effective manner. Consider the following situation where a person needs a recommendation about edible seeds intake being beneficial to improve its health. The recommendation system compare the common properties of the edible seeds, their benefits, and contraindications. To accomplish this, the system's reasoning process must find items to compare alternatives, *i.e.*, some common descriptors to the options under consideration. This example describes a particular pattern of reasoning that is used in order to reach a goal or a conclusion. This pattern of reasoning was called *Argument from Analogy*, and have been expressed in a semiformal way by Walton [31, 30] using a set of critical questions.

In a general sense, argumentation can be associated with an interactive process where arguments for and against conclusions are offered with the purpose of determining which conclusions are acceptable [27]. Several argument-based formalisms have considered an argument like as abstract entity without internal structure [11, 9], while other works that specify concrete forms of building arguments [3, 24, 12]. In addition, there exist some argumentative formalisms that represent the attributes associated to arguments providing more information to determine arguments acceptability [2, 8, 5]. However, these formalisms do not deal with the problem of classifying similar arguments considering the natural descriptors inherent to each argument. In this work, we will propose an *Argumentation Scheme* that that allows to considerer the context of comparison between arguments, based on a set of descriptors that are common to the arguments that are being analyzed. In this way, we will determine and represent analogies between arguments. This paper is structured as follows: a brief introduction to the *analogy concept* is presented in *Section 2*. In *Section 3*, we review the Argumentation Scheme proposed by Walton and, in *Section 4*, contains the core contribution of the paper, there the *Context-Aware Scheme* is formally presented. Finally, in *Section 5* we present the relevant related work, the conclusions and future work.

2 The Concept of Analogy

The term *analogy* has been widely studied as to their meaning and usage [13, 23]. Hesse [15] argues that the word is self-explanatory and that two objects or situations are similar if they share some properties and differ in others, and put forward that the analogy is a concept inherent in the modelling process, between a portion of the real world and its model or description, as a relationship that finds similarities and differences between both. Walton [30] agrees with

this perspective adding that two things are similar when they are visibly similar or they look similar. Gentner [13] linked the concept of analogies with the representation of the agent’s knowledge through the pattern’s repetition. First, these patterns should be identified to find relations of correspondence with new situations. Then, it is important to perform a mapping of domains so that these relations of correspondence do not produce a fallacious reasoning. Under this meaning the analogy is considered as a *process* which recovers information from a known situation to compare it with a new one and evaluate whether to apply the same solution according to the structure of the both situations. As to how to determine when two arguments are similar, Walton points out that it is not easy to clearly define the comparison between arguments as this requires interpreting the similarities and differences between them at various levels. Offering another view, Carbonell [7] proposes a technique based on how we solve problems. This technique takes into account previous experience information useful for solving a new problem, as long as both occur in similar contexts. In [28], Sowa argues that it is possible to make a comparison between arguments establishing a function of similarity or correspondence between them. In a parallel effort, Cecchi *et al.* in [10] characterized and formalized relationships that capture the behavior of a preference criterion among arguments; while this does not refer specifically to arguments from analogy, it shows the usefulness in approaching the analogy between two arguments as a binary relation.

Considering the variety of concepts presented, we can say that the analogy is a process that involves the comparison between two entities, for example, objects that look similar, cases, or two arguments in which similarities and differences are interpreted. The critical component in this process is the definition of the conditions under which carry out such comparison. Clearly, the analogical comparison process have received different answers and remain the focus of different research lines. The similarity is related to the properties shared between two objects or situations being compared. However, the comparison of two arguments depends on the agent’s perception, which can be influenced by the agent’s beliefs, goals, or external variables. All these factors are considered as a context that governs the mapping of two arguments in order to establish similarities and differences between them. The intuition that we have just presented is essential to define a *context-aware scheme* which is based on the argumentation *scheme from analogy* proposed by Walton [31, 30], to which we will refer below.

3 Argumentation Schemes from Analogy

A more applied perspective referring to the pattern of reasoning which appeal to analogies addressed by Walton [31, 30], who proposed an Argumentation Scheme from Analogy consist of a set of questions, premises and conclusions that describe a pattern of human thought [29]. Specifically, the pattern of reasoning based on analogies. compares two situations C_1 and C_2 (or cases, as Walton refers to them in the setting of law) to find similarities and differences between them. In this pattern of thought, C_1 is the *source case or known case* and C_2 is the *target case*

or *new case* [31]. Two cases may be similar in a given context, but they may be different in another. The general pattern of this scheme is presented below [31]:

- *Similarity premise*: C_1 is similar to C_2 (Generally)
- *Base premise*: A proposition A is true (false) in C_1
- *Conclusion*: A is true (false) in C_2

The defeasible character is introduced by the specific differences between C_1 and C_2 . Walton defined three *critical questions* that are appropriate for using the scheme of argument from analogy:

1. Are there differences between C_1 and C_2 that would tend to undermine the force of the similarity cited?
2. Is the feature A true (false) in C_1 ?
3. Is there some other case C_3 also similar to C_1 , but in which the feature A is false (true)?

Walton [30] analyzed different possibilities for this type of schema, for example the usage of argument from classification and the argument from precedent applied in case-based reasoning by the use of a dialogue structure. Another representation is detailed by [22] that considers a relationship in which the objects are instantiated as arguments that maintain an analogical proportion, and that can be expressed as a differs from b as c differs from d , either, a is similar to b as c is similar to d , where a , b , c , and d are entities or situations of any kind including arguments. In both cases, the representation schemes refer to the comparison of situations, entities, or cases that may implicitly contain arguments, where a context constraint plays a determinant role but it is not explicitly defined. Then, we propose below a definition of a *context-aware scheme* that allows us to compare two arguments with the intention of find similarities and differences between them. This approach is based on the exhaustive definition of *constraint set* or *context constraint* inherent of the comparison process between arguments.

4 A Context-Aware Scheme

An alternative to define the analogy relation between two arguments as a computable approximation, is to make the comparison based on a *set of descriptors* associated with the arguments. The use of these descriptors will allow to establish of similarity and difference degree between the arguments into account and formalize an argumentation schemes. We will develop these insights below, considering the following motivation example previously.

Example 1 Consider the following set of arguments: $\mathcal{A} = \{A, B, C, D\}$ where:

- A *To incorporate chia seeds to your diet is a healthy choice since they are rich in vegetal fats, proteins, antioxidants, and minerals. This seeds helps to reduce conditions such as oxidative stress.*
- B *Amaranth seeds provide vitamin A, E, from the B group, calcium, iron, and phosphorus. So this seeds help us in preventing deficiency anemia.*

- C *Sesame seeds provide high quantity of calcium, antioxidants, fatty acids, and proteins. Therefore, to ingest sesame seeds is important in preventing osteoporosis.*
- D *Chia seeds are harmful to hypotensive individuals because it lowers blood pressure. Therefore, it is not always healthy to incorporate them into the daily diet.*

We want to propose a method that allows to determine the set of similar arguments among themselves, according to a given context. In general, this method consists of the following four stage: (i) *Specification of argument’s descriptors*; (ii) *Setting the context constraint based on argument’s descriptors*; (iii) *Definition of a preference relation over the context constraint*; and (iv) *Computing similarity quotient among the arguments being compared*. Next, we describe each stage of this procedure, in order to clarify the proposed reasoning process.

4.1 Specification of Argument’s Descriptors

The set of argument’s descriptors is constituted by words or a labels describing aspects that the argument is referring to. In this sense, we assume the existence of a universe of descriptors denoted with \mathcal{D} , where the set of descriptors associated with an argument A is denote with $desc(A)$ such that $desc(A) \subseteq \mathcal{D}$.

Example 2 *An example of a universe of descriptors for a nutritional/health related knowledge base that contains the argument set A of the Example 1 could be the set $\mathcal{D} = \{type_of_food, health_benefits, dietary_contribution, health_risks\}$. Given the Argument A we could specify the following descriptors: $desc(A) = \{type_of_food, health_benefits\}$ stating that the arguments refer to a particular type of food, specific contributions to a diet, and the benefits of its consumption to ones’ health. From the perspective of analogical proportion argument A is similar to argument B as the argument C is the argument D , if we consider that the first two refer to ingestion of eggs while the latter two refer to the intake of seeds.*

Find the descriptors embedded in the arguments involves technical practice of argumentation mining which are being widely investigated [17, 18, 21].

4.2 Setting the Context Constraint

A context constraint specifies conditions under which arguments can be compared. In this sense, two arguments can only be compared if they share at least some descriptors. We can present the following definition, based on the comparability of two arguments. In the rest of the paper, whenever there is no ambiguity, we will refer to context constraints simply as contexts.

Definition 1 (Context Constraint) *Let \mathcal{D} a set of descriptors. A context constraint, denoted as Δ , is a subset of \mathcal{D} that represent the relevant aspects to perform the arguments comparison in a particular domain.*

Example 3 *Continuing with the setting of the Example 2, the elements defined in \mathcal{D} could be instantiated as follows. Let $\Delta_1 = \{type_of_food, dietary_contribution\}$ be a context indicating that two arguments can be compared in this environment whenever they refer to a type of food and dietary contributions of this food. Other contexts could be $\Delta_2 = \{health_benefits\}$.*

We can observe that there is at least one item in Δ which prevails in the comparison between two arguments (in this example could be *type_of_food*), while there are other elements in Δ which indicate properties that can take the null value (for example, *health_benefits*). This implies that we must establish a preference relation among the elements of Δ to compute this type of information. We will refer to them below.

4.3 Definition of a Preference Relation over the Context

When the context has more than one descriptor, it is necessary to establish some order among elements in order to carry out the comparison of the arguments more accurately, which can be expressed as follows:

Definition 2 (Preference between Descriptors) *Let $\nu_1, \nu_2 \in \Delta$ be descriptors of arguments. Let pref be a preference relation between ν_1 and ν_2 . We say that $(\nu_1, \nu_2) \in \text{pref}$ iff ν_1 is a prevailing condition to make the comparison between arguments, i.e. ν_1 cannot take the value null, and there is no $\nu_2 \in \Delta$ such that ν_2 has the opposite meaning to ν_1 given to ν_1 is a fundamental descriptor.*

This preference relation can be specified in various ways according to the domain being modelled. For example, weights or probability values can be assigned to the descriptors, or a partial order relation between descriptors could be established. Once descriptors, the context, and the preference relation between descriptors were defined, we can compute the similarities and differences between two arguments. However, it should be noted that descriptors take values for an argument in particular. Thus, we should consider some mapping function μ that establishes the relation between each descriptor in Δ and the set of concepts to which the argument is referring to— these concepts could, for instance, be just words in natural language or more complex concepts in an ontology. We will not focus on formalizing these concepts; however, intuitively, we can say that using the mapping function it is possible obtain the set of words or concepts that a given argument $A \in \mathcal{A}$ refers to. As an example of how to formalize such a function, it could be based on the *intensional relational structure, or conceptualization* presented in [14] as a triple consisting of a universe of discourse, a set of possible worlds or values that characterize a system, and a set of conceptual relations on two previous ones. With all these elements in place, we can compute the similarities between two arguments, which will detail below.

4.4 Computing Similarity Quotient among Arguments

To decide if two arguments are analogous, we propose to evaluate the values that take their descriptors. If the descriptors with matching values exceeds the descriptors with different values, and the prevailing conditions are fulfilled, the arguments are considered analogous. Otherwise, the arguments are considered different. This involves calculating a similarity quotient between the values of argument's descriptors that are being compared.

Definition 3 (Similarity Quotient) Let $A, B \in \mathcal{A}$ be two arguments to be compared, Δ be a context orderer by a preference relation *pref*, and μ be a mapping function. We define the similarity quotient $\alpha_\mu(A, B)$ as:

$$0 \leq \frac{|\mu(A) \cap \mu(B)|}{|\overline{\mu(A) \cap \mu(B)}|} \leq 1$$

where $|\overline{\mu(A) \cap \mu(B)}|$ must be different to 0.

A null divisor in our case means that if the arguments under comparison do not have differences, it makes no sense consider an analogy between them, but rather an identity in the context used. It is important to remark that in this approach, there is no difference in calculating $\alpha_\mu(A, B)$, or $\alpha_\mu(B, A)$, *i.e.*, we assume that similarity quotient is symmetric. With these elements in place, we can now propose a notion of analogy between arguments. Intuitively, we consider that two arguments A and B are *analogous* iff the similarity quotient is greater than 0.5. According to previous definitions, this implies that the arguments being compared under a interest context and all the prevailing conditions are fulfilled. The following definition formalizes the notion analogy relation between arguments that we adhere to in this work, and it is a refining of the one presented in [6].

Definition 4 (Analogy Relation) Let \mathcal{A} be a set of arguments, Δ be a context orderer by a preference relation, and α_μ be a similarity quotient between two arguments in \mathcal{A} . An analogy relation, denoted \mathbb{R}_Δ , is defined as $\mathbb{R}_\Delta \subseteq \mathcal{A} \times \mathcal{A}$, where $(A, B) \in \mathbb{R}_\Delta$ iff $\alpha_\mu(A, B) > 0.5$, all the prevailing conditions are fulfilled, and verifies that: (i) $\mathbb{R}_\Delta(A, B) = \mathbb{R}_\Delta(B, A)$; (ii) $\mathbb{R}_\Delta(A, A) = 1$; and (iii) $\mathbb{R}_\Delta(A, B) = 0$ iff $\mu(A) \cap \mu(B) = \emptyset$.

Example 4 Picking up the Example 3, we need to center on the value that each of the descriptors takes for every argument, according to the context Δ . For this comparison, we take a context $\Delta = \{\text{dietary_contribution}\}$. In this case, the similarity quotient regarding Δ could be calculated for A, B and C in \mathcal{A} , using the number of descriptors of the arguments A and C who take the same value, over the number of descriptors of those arguments that take different values. In the same way we proceeded with the arguments B and C . Arguments A and C are analogous in this context because $\alpha_\mu(A, C) = 0.5$. However, arguments B and C are not analogous in this context, due to $\alpha_\mu(B, C) = 0.33$.

Note that we make no assumption about the relation to be transitive; this is, if $A \mathbb{R}_\Delta B$ and $B \mathbb{R}_\Delta C$, then not necessarily must the case that $A \mathbb{R}_\Delta C$. It may happen that $\alpha_\mu(A, B) > 0.5$, $\alpha_\mu(B, C) > 0.5$ and $\alpha_\mu(A, C) < 0.5$. Clarified the meaning of the analogy between arguments, we can define below an argumentation scheme based on context information.

Definition 5 (Context-Aware Scheme) Let \mathcal{A} be a set of arguments, Δ be a context orderer by a preference relation, and α_μ be a similarity quotient between two arguments A and $B \in \mathcal{A}$. The context-aware scheme is defined as:

- Similarity premise: A is analogous to B

- Base premise: $\mu(A) = P$, where P is the descriptor's value for A
- Conclusion: $\mu(B) = P$, where P is the descriptor's value for B .

In this scheme, the similarity premise implies that the analogy relation between two arguments is greater than the difference between them, considering a orderer context. The comparison between arguments is given to for the prevailing conditions and the rest of the descriptors which are interesting in the context. The defeasible character is introduced by the context under which the arguments are compared:

1. Exist Δ' that would tend to undermine the force of the analogy cited?
2. Is the descriptor $\mathbf{P} \in \Delta'$ true and is \mathbf{P} a prevailing condition?

When considering the original critical questions defined by Walton, the similarity between two cases can be questioned. In the context-aware scheme, the analogy is established considering a specific context for comparison. The first thing that it is possible to question is whether the comparison context that has been considered is the right. Then, its possible to question about the preference relation in the context.

Example 5 *Continuing the example that we have been developing, we have:*

- Similarity Premise: A is analogous to C , considering $\Delta = \{type_of_food\}$.
- Base premise: $\mu(A) = \{seeds\}$.
- Conclusion: $\mu(C) = \{seeds\}$.

We can evaluate what happens when we take the context $\Delta' = \{dietary_contributions\}$, due to $\mu(A) = \{vegetal_fats, proteins, antioxidants, minerals\}$ and $\mu(C) = \{calcium, antioxidants, proteins, fatty_acids\}$.

In addition to particularize the analogy between arguments, the context-aware scheme presented is useful to give a new meaning to relations between arguments when they are considered abstract entities without internal structure, which will be addressed in future work.

5 Related Works and Conclusions

Few studies exist formalizing the argumentation schemes proposed by Walton. Amgoud [1] put forward that argumentation from analogies has not been exploited profitably by AI, being the structure-mapping model [4] the exception. Nevertheless, any attempt to deal with the use of analogies in the argumentative process should include three aspects: the difference of an argumentative process from analogies with the argumentative process in general, the definition of attacks and the evaluation of arguments in this new approach. Prakken [24] proposed Argumentation Systems with Structured Arguments, which used the structure of arguments and external preference information to define the a defeat relation. Regarding argumentation schemes, Prakken [25] proposes that modeling reasoning using argumentation schemes necessarily involves

developing a method combining issues of non-monotonic logic and dialogue systems. Regarding to preference relation between arguments, Cecchi *et al.* [10] defined this as a binary relation considering two particular criteria, specificity and equi-specificity, together with priorities between rules, defining preferred arguments and incomparable arguments. Specifically, with regards to formalizing argumentation schemes, Hunter [16] presented a framework for meta-reasoning about object-level arguments allowing the presentation of richer criteria for determining whether an object-level argument is warranted. These criteria can use meta-information corresponding to the arguments, including the proponents and their provenances and an axiomatization using this framework for reasoning about the appropriated conduct of the experts introducing them.

We have presented a formalism based on the comparison of arguments through descriptors. A descriptor is a word or a label that describes an aspect or element that the argument refers to. The arguments can be compared if they share a set of descriptors that represent the context of comparison. In order to compare the arguments, we have defined an analogy relation that considers the similarities and differences between arguments under certain context. We have formulated a Context-Aware Scheme based on Scheme from Analogy proposed by Walton [31, 29]. The goal of this formalization is to make it more useful to use reasoning patterns based on similarities and difference arguments in the field of AI. Different instruments can be specified to establish the comparison of the arguments, for example, the descriptors. Regardless of the type of instrument used, comparisons adjusted to a particular context allow us to add common sense to the reasoning of the agent. Among the main future work to address we can mention the specification of the mapping function between arguments and the definition of a framework that makes this type of schemes more useful for AI.

References

1. Amgoud, L., Ouannani, Y., Prade, H.: Arguing by analogy—towards a formal view. *Similarity and Analogy-based Methods in AI SAMAI 2012* p. 65 (2010)
2. Bench-Capon, T.: Value based argumentation frameworks. *arXiv preprint cs/0207059* (2002)
3. Besnard, P., Hunter, A.: A Logic-Based Theory of Deductive Arguments. *AI* 128(1-2), 203–235 (2001)
4. Bohan, A., Keane, M.T.: Towards a computational model of analogical arguments. In: *Proc. of 4th Workshop on Computational Models of Natural Argument (CMNA IV)*, at *ECAI*. vol. 4, pp. 65–67 (2004)
5. Budán, M.C., Lucero, M.G., Viglizzo, I., Simari, G.R.: A labeled argumentation framework. *Journal of Applied Logic* 13(4), 534–553 (2015)
6. Budán, P.D., Martínez, M.V., Budán, M.C., Simari, G.R.: Introducing analogy in abstract argumentation. *WL4AI-2015* p. 25 (2015)
7. Carbonell, J.: *Learning by analogy: Formulating and generalizing plans from past experience*. Springer (1983)
8. Cayrol, C., Lagasque-Schiex, M.C.: Gradual valuation for bipolar argumentation frameworks. In: *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pp. 366–377. Springer (2005)

9. Cayrol, C., Lagasquie-Schiex, M.C.: On the acceptability of arguments in bipolar argumentation frameworks. In: Symbolic and quantitative approaches to reasoning with uncertainty, pp. 378–389. Springer (2005)
10. Cecchi, L., Simari, G.: Sobre la relación de preferencias entre argumentos. In: VIII Congreso Argentino de Ciencias de la Computación (2002)
11. Dung, P.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *AI* 77(2), 321–357 (1995)
12. García, A.J., Simari, G.R.: Defeasible logic programming: Delp-servers, contextual queries, and explanations for answers. *Arg. & Comp.* 5(1), 63–88 (2014)
13. Gentner, D., Holyoak, K.J., Kokinov, B.N.: The analogical mind: Perspectives from cognitive science. MIT press (2001)
14. Guarino, N., Oberle, D., Staab, S.: What is an ontology? In: Handbook on ontologies, pp. 1–17. Springer (2009)
15. Hesse, M.: Models and analogies in science, vol. 7. U. of Notre Dame Press Notre Dame (1966)
16. Hunter, A.: Reasoning about the appropriateness of proponents for arguments. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence, USA (2008)
17. Lawrence, J., Reed, C., Allen, C., McAlister, S., Ravenscroft, A., Bourget, D.: Mining arguments from 19th century philosophical texts using topic based modelling. *ACL 2014* p. 79 (2014)
18. Lippi, M., Torroni, P.: Context-independent claim detection for argument mining. In: Proceedings of the Twenty-Fourth Inter. Conference on AI. pp. 185–191 (2015)
19. Macagno, F., Walton, D.: Argument from analogy in law, the classical tradition, and recent theories. *Philosophy and rhetoric* 42(2), 154–182 (2009)
20. Nejtkovic, V., Tosic, M.: Using analogy computing for ontology development. In: 2016 IEEE 32nd International Conference on Data Engineering Workshops (ICDEW). pp. 115–120. IEEE (2016)
21. Peldszus, A., Stede, M.: Joint prediction in mst-style discourse parsing for argumentation mining. In: Proc. of the Conference on Empirical Methods in Natural Language Processing. pp. 938–948 (2015)
22. Prade, H., Richard, G.: Analogical proportion—a brief survey. *Similarity and Analogy-based Methods in AI SAMAI 2012* (2012)
23. Prade, H., Richard, G.: Computational Approaches to Analogical Reasoning: Current Trends, vol. 548. Springer (2014)
24. Prakken, H.: An abstract framework for argumentation with structured arguments. *Arg. and Comp.* 1(2), 93–124 (2010)
25. Prakken, H.: On the nature of argument schemes. *Dialectics, dialogue and argumentation. An examination of Douglas Walton’s theories of reasoning and argument* pp. 167–185 (2010)
26. Raad, E., Evermann, J.: The role of analogy in ontology alignment: A study on lisa. *Cognitive Systems Research* 33, 1–16 (2015)
27. Simari, G.R., Loui, R.P.: A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence* 53(1–2), 125–157 (1992)
28. Sowa, J., Majumdar, A.: Analogical reasoning. In: *Conceptual Structures for Knowledge Creation and Communication*, pp. 16–36. Springer (2003)
29. Walton, D.: *Fundamentals of critical argumentation*. Cambridge Univ Press (2006)
30. Walton, D.: Similarity, precedent and argument from analogy. *AI and Law* 18(3), 217–246 (2010)
31. Walton, D., Reed, C., Macagno, F.: *Argumentation Schemes*. Cambridge University Press, Cambridge, UK (2008)

Immune Algorithm for Solving the Dynamic Economic Dispatch Problem

Victoria S. Aragón^{1,2} and Susana C. Esquivel¹

¹Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)
Universidad Nacional de San Luis -Ejército de los Andes 950 - (5700) San Luis,
ARGENTINA

^{1,2} CONICET, Universidad Nacional de San Luis, Argentina

Abstract. In this paper, an algorithm inspired on the immune system is presented, IA_DED stands for *Immune Algorithm Dynamic Economic Dispatch*, it is used to solve the Dynamic Economic Dispatch problem. IA_DED uses as differentiation process a redistribution power operator and the output power are integer values. The proposed approach is validated using three problems taken from the specialized literature. Our results are compared with respect to those obtained by several other approaches.

Keywords: Artificial immune systems , dynamic economic dispatch problem , metaheuristics

1 Introduction

One of the early problems in power system optimization is the Dynamic Economic Dispatch (DED) problem. Its main objective is to determine the optimal schedule of output powers of on line generating units, over a certain period of time (T intervals), to meet power demands at minimum operating cost [10]. Besides, several constraints associated to the system have to be satisfied, such as load demands, ramp rate limits, maximum and minimum limits, and prohibited operating zones. As DED problem is nonlinear different heuristics have been used to solve it, such as, artificial immune system [8], genetic algorithm [7], particle swarm optimizer [7], algorithm bee colony [7], harmony search [14], [12], [2], imperialist competitive algorithm [11], an hybridized differential evolution [3], among others. Surveys about these techniques can be found in [16] and [9].

In this paper, we propose an algorithm to solve DED problem which is inspired on the immune system. Considering T intervals, the problem is regarded as a sequence of T problems. But, each problem (at time i) depends on its predecessor (at time $i - 1$) and it conditions to its successor (at time $i + 1$). The algorithm applies a redistribution power operator in order to improve a solution at time t with the aim of keeping such a solution feasible at a low computational cost.

The remainder of this paper is organized as follows. Section 2 defines the DED problem. In Section 3, we describe our proposed algorithm. In Section 4, we present the test problems used to validate our proposed approach and parameters settings. In Section 5, we present our results and we discuss and compare them with respect to other approaches. Finally, in Section 6, we present our conclusions and some possible paths for future research.

2 Problem Formulation

The DED problem minimizes the total production cost (TC) associated with N dispatch units for a time period:

$$TC = \sum_{t=1}^T \sum_{i=1}^N F_i(P_i^t) \quad (1)$$

where TC is the fuel cost over the whole dispatch period, T is the number of intervals in the period, N is the number of generating units in the system, P_i^t is the power of i^{th} unit at time t (in MW) and F_i is the fuel cost for the i^{th} unit (in \$/h).

A smooth fuel cost function can be expressed as a single quadratic function:

$$F_i(P_i^t) = a_i(P_i^t)^2 + b_iP_i^t + c_i \quad (2)$$

where a_i , b_i and c_i are the fuel consumption cost coefficients of the i^{th} unit. But, if the valve-point effects are taking into account, the fuel cost function of the i^{th} unit is expressed as the sum of a quadratic and a sinusoidal function in the form:

$$F_i(P_i^t) = a_i(P_i^t)^2 + b_iP_i^t + c_i + |e_i \sin(f_i(P_{min_i} - P_i^t))| \quad (3)$$

where e_i and f_i are the fuel cost coefficients of the i^{th} unit with valve-point effects.

Regardless the function considered (Eq. 2 or Eq. 3), its minimization is subjected to:

1. Power Balance Constraint: the power generated has to be equal to the power demand required. It is defined as:

$$\sum_{i=1}^N P_i^t - P_D^t - P_L^t = 0 \quad (4)$$

where $t = 1, 2, \dots, T$. P_D^t is the total power demand at time t , and P_L^t is the transmission power loss at time t (in MW). P_L^t is calculated using the B-matrix loss coefficients, and the general form of the loss formula using B-coefficients is:

$$P_{Lt} = \sum_{i=1}^N \sum_{j=1}^N P_i^t B_{ij} P_j^t + \sum_{i=1}^N B_{0i} P_i^t + B_{00} \quad (5)$$

If transmission power loss is not considered, $P_L^t = 0$.

2. Operating Limit Constraints: units have physical limits about the minimum and maximum power they can generate:

$$P_{min_i} \leq P_i^t \leq P_{max_i} \quad (6)$$

where P_{min_i} and P_{max_i} are the minimum and maximum power output of the i^{th} unit in MW, respectively.

3. Ramp Rate Limits: they restrict the operating range of all on-line units. Such limits indicate how quickly the unit's output can be changed:

$$\begin{cases} P_j^t - P_j^{(t-1)} \leq UR_j & \text{if } P_j^t > P_j^{(t-1)} \\ P_j^{(t-1)} - P_j^t \leq DR_j & \text{if } P_j^t < P_j^{(t-1)} \end{cases} \quad (7)$$

where $P_j^{(t-1)}$ is the output power of j^{th} unit at previous hour and UR_j and DR_j are the ramp-up and ramp-down limits of the j^{th} unit in MW, respectively. Due to ramp-rate constraints, Eq. 6 is modified as:

$$max(P_{min_j}^t, P_j^{(t-1)} - DR_j) \leq P_j^t \leq min(P_{max_j}^t, P_j^{(t-1)} + UR_j) \quad (8)$$

such that

$$\begin{cases} P_{min_j}^t = max(P_{min_j}, P_j^{(t-1)} - DR_j) \\ P_{max_j}^t = min(P_{max_j}, P_j^{(t-1)} + UR_j) \end{cases} \quad (9)$$

4. Prohibited Operating Zones: they restrict the operation of the units due to steam valve operation conditions or to vibrations in the shaft bearing. Thus, an unit with prohibited operating zones has a discontinuous input-output power generation characteristic which gives rise to additional constraints on the unit operating range.

$$\begin{cases} P_{min_i} \leq P_i^t \leq PZ_{i,1}^L & \text{or} \\ PZ_{i,k-1}^U \leq P_i^t \leq PZ_{i,k}^L, & \text{or} \\ PZ_{i,n_1}^U \leq P_i^t \leq P_{max_i} & k = 2, 3, \dots, n_i \end{cases} \quad (10)$$

where n_i is the number of prohibited zones of the i^{th} unit, k is the index of the prohibited operating zones of the i^{th} unit. $PZ_{i,k}^L$ and $PZ_{i,k}^U$ are the lower and upper bounds of the k^{th} prohibited operating zones of unit i .

3 Our Proposed Algorithm

In this paper, an adaptive immune system model based on the immune responses mediated by the T cells from the immune system is presented. These cells present special receptors on their surface called T cell receptors (TCR), they are responsible for recognizing antigens bound to major histocompatibility complex (MHC molecules.) [13].

The model considers some processes that T cells suffer. These are proliferation (to clone a cell) and differentiation (to change the clones so that they acquire specialized functional properties); this is the so-called activation process. IA_DED (Immune Algorithm for Dynamic Economic Dispatch problem) is an adaptation of an algorithm inspired on the activation process [1], which was proposed to solve the economic dispatch problem. IA_DED operates on one population which is composed of a set of T cells.

For each cell, the following information is kept:

1. *TCR*: it identifies the decision variables of the problem ($TCR \in \mathbb{N}^N$). Each thermal unit is represented by one decision variable, each variable is encoded by an integer value.
2. *objective*: objective function value for TCR, ($TC(TCR)$).
3. *prolif*: it is the number of clones that will be assigned to the cell, it is N for all problems.
4. *diff*: it is the number of decision variables that will be changed when the differentiation process takes place (if applicable). This level is calculated as $U(1, N)$.
5. *TP*: it is the power generated by TCR ($\sum_{i=1}^N TCR_i$).
6. P_L^t : it is the transmission loss for TCR, according to Eq. 5.
7. *ECV*: it is the equality constraint violation for TCR. At t time ($|TP - P_D - P_L|$). If $ECV > 0$, then the power generated is bigger than the demanded power, and if $ECV < 0$ then the power generated is lower than the required power.
8. *ICS*: it is the inequality constraints sum, $\sum_{i=1}^N \sum_{j=1}^{n_i} poz(TCR_i, i, j)$

$$poz(p, i, j) = \begin{cases} \min(p - PZ_{i,j}^L, PZ_{ij}^U - p) & \text{if } p \in [PZ_{i,j}^L, PZ_{ij}^U] \\ 0 & \text{otherwise} \end{cases}$$
 where n_i is the number of prohibited operating zones and $[PZ_{i,j}^L, PZ_{ij}^U]$ is the j^{th} prohibited range for the i^{th} unit.
9. *feasible*: it indicates if the cell is feasible or not. A cell is considered as feasible if: 1) $ECV = 0$ for problems without transmission network loss and $0 \leq ECV < \epsilon$ for problems with transmission loss and 2) $ICS = 0$ for problems which consider prohibited operating zones.

Differentiation for feasible cells - Redistribution Process

The idea is to take a value (called d) from one unit (say i) and assign it to another unit (say j). i^{th} and j^{th} units are modified according to: $cell.TCR_i = cell.TCR_i - d$ and $cell.TCR_j = cell.TCR_j + d$, where $d = U(1, (int)[(P_1 * \min(cell.TCR_i - P_{min_i}^t, P_{max_j}^t - cell.TCR_j))])$, $U(w_1, w_2)$ refers to a random

number with a uniform distribution in the range (w_1, w_2) and P_1 is a change factor ($P_1 \in [0, 1]$). Besides, a probability, P_2 , determines if i^{th} and j^{th} units will be modified.

Differentiation for infeasible cells

For infeasible cells, the number of decision variables to be changed is determined by their differentiation level (*diff*). Each variable to be changed is chosen in a random way and it is modified according to: $cell.TCR'_i = cell.TCR_i \pm m$, where $cell.TCR_i$ and $cell.TCR'_i$ are the original and the mutated decision variables, respectively. $m = U(1, (int)](cell.ECV + cell.ICS)$. In a random way, it decides if m will be added or subtracted to $cell.TCR_i$. If the procedure cannot find a TCR'_i in the allowable range, then a random number with a uniform distribution is assigned to it ($cell.TCR'_i = U(cell.TCR_i, P_{max_i}^t)$ if m should be added or $cell.TCR'_i = U(P_{min_i}^t, cell.TCR_i)$, otherwise). If the resulting clon is feasible then differentiation process stops, otherwise, the process is applied to the resulting clon instead the original infeasible cell. This methodology follows until *prolif* differentiations have been applied or a feasible clon has been reached.

Algorithm 1 IA_DED Algorithm

```

1:  $P \leftarrow \text{Initialize\_Population}()$ ;
2:  $\text{Evaluate\_Constraints}(P)$ ;
3:  $\text{Evaluate\_Objective\_Function}(P)$ ;
4: for  $t \leftarrow 1$  to  $T$  do
5:    $top \leftarrow 0$ ;
6:   while A predetermined number of evaluations has not been reached and  $top < 5 * 10^7$  do
7:      $\text{Proliferation\_Population}(P)$ ;
8:      $\text{Differentiation\_Population}(P)$ ;
9:      $top ++$ ;
10:  end while
11:   $best_t \leftarrow \text{Search\_best\_at\_Population}(P, t)$ ;
12:   $t ++$ ;
13:   $\text{Update\_limits}(best_t)$ ;
14:   $\text{Repair\_output\_power}(P)$ ;
15:   $\text{Update\_output\_power}(P)$ ;
16:   $\text{Evaluate\_Constraints}(P)$ ;
17:   $\text{Evaluate\_Objective\_Function}(P)$ ;
18: end for
19:  $BEST \leftarrow (best_1, best_2, \dots, best_T)$ ;

```

The algorithm works in the following way (see Algorithm 1). First, the TCRs are randomly initialized within the limits of the units (Step 1). Then, *ECV* and *ICS* are calculated for each cell (Step 2). Only if a cell is feasible, its objective function value is calculated (Step 3). Next, the following steps are repeated T times (Step 5 to 17): while a predetermined number of objective function evaluations had not been reached and $5*10^7$ iterations are not performed

the cells are proliferated and differentiated according to their feasibility. After activation process, best solution at t time is recorded. The time is increased and new operational limits are updated according Eq. 3. Those units which outputs power falling out the new operational limits are changed by random values from the valid limits. Finally, (Step 19) the whole final solution is sequence of solutions found in time 1, time 2, to time T .

4 Validation

IA_DED performance was validated with three test problems, 5-unit system [11], 15-unit system [5], 54-unit system [4]. Table 1 provides their most relevant characteristics and the maximum number of function evaluations. IA_DED was implemented in Java (version 1.6.0_24) and the experiments were performed in an Intel Q9550 Quad Core processor running at 2.83GHz and with 4GB DDR3 1333Mz in RAM.

The required parameters by IA_DED are: size of population, maximum number of objective function evaluations, change factor (P_1) and probability for redistribution operator (P_2). To analyze the effect of the first and third parameters on IA_DED's behavior, we tested it with different parameters settings. Some preliminary experiments were performed to discard some values for the population size parameter. Hence, the selected parameter levels were: a) Population size (C) has four levels: 5, 10 and 50 cells, b) Probability P_1 has three levels: 0.1, 0.5 and 0.9 and c) Probability P_2 has two levels: 0.01 and 0.1.

Thus, we have 18 parameters settings for three problems. They are identified as $C<size>-P1<Prob>-P2<Prob>$, where C , $P1$ and $P2$ indicate the population size and the probabilities, respectively. For each problem, 100 independent runs were performed. The box plot method was selected to visualize the distribution of the objective function values for each power system. This allowed us to determine the robustness of our proposed algorithm with respect to its parameters. Figure 1 show in the x-axis the parameter combinations and the y-axis indicates the objective function values for each problem. We can see for 5-unit system and 15-unit system the results are robust. For 54-unit system, to increase the change factor improve the results and to increase the probability of application of the redistribution process and size population deteriorate the results. So, the settings parameters were used to compare the results got by IA_DED with those produced by other approaches are: for 5-unit system $C=10$ and $P_1=0.1$, $P_2=0.1$, for 15-unit system $C=50$ and $P_1=0.9$, $P_2=0.1$, for 54-unit system $C=5$ and $P_1=0.9$, $P_2=0.01$. Also, we set $\epsilon=2.0$ for those problems which consider loss transmission.

5 Comparison of Results and Discussion

Several methods are compared with IA_DED. They are listed next with the maximum number of function evaluations performed (if the value was available): AIS [8] (300000), GA [7] (not found), PSO [7] (not found), ABC [7] (not found), HS [14] (50000), ICA [11], for 5-unit system, (20000), DE-SQP [3] (50000), NPAHS

Table 1. Test Problems Characteristics

Problem	Thermal Units	Objective	P_L	Prohibited Zones	P_D (MW)	Evaluations
5-unit system	5	non-smooth	Yes	No	14577	19000
15-unit system	15	smooth	Yes	No	60981	19000
54-unit system	54	non-smooth	No	Yes	111600	30000

[12] (50000) CSADHS [2] (250000), SAMF [6] (not indicated), OCD [15] (not indicated), ICA [11] (80000), for 54-unit system.

Table 2 shows: the best, worst, mean, standard deviation and running times obtained by the approaches. For IA_DED only four decimal digits are shown due to space restrictions. For all the test problems, our proposed IA_DED found feasible solutions in all the runs performed.

The running time of each algorithm is affected by both the hardware environment and the software environment. That is the reason why the main comparison criterion that we adopted for assessing efficiency was the number of objective function evaluations performed by each approach. For having a fair comparison of the running times of all the algorithms considered in our study, they should all be run in the same software and hardware environment (something that was not possible in our case, since we do not have the source code of several of them). Clearly, in our case, the emphasis is to identify which approach requires the lowest number of objective function evaluations to find solutions of a certain acceptable quality.

However, the running times are also compared in an indirect manner, to give at least a rough idea of the complexities of the different algorithms considered in our comparative study. Analyzing Table 2, IA_DED, for 5-unit system, is outperformed by ICA [11] and DE-SQP [3], but it can found quickly an acceptable solutions performing less evaluations of objective function. For 15-unit system, IA_DED outperformed all approaches which we compare it, it found the best solution both running time and total fuel cost. For 54-unit system, IA_DED outperformed all approaches which we compare it, taking into account the total fuel cost. It need 8.565s to find this solution, however it costs \$54547 less than OCD's solution.

6 Conclusions and Future Work

This paper presented an adaptation of an algorithm inspired on the T-Cell model of the immune system, called IA_DED, which was used to solve dynamic economic dispatch problems. IA_DED is able to handle the five types of constraints that are involved in this kind of problems: power balance constraint with and without transmission loss, operating limit constraints, ramp rate limit constraint and prohibited operating zones, and different types of objective function: smooth and non-smooth.

Table 2. Comparison of results. The best values are shown in **boldface**. - denotes that the value was not available in the literature.

Problem/ Algorithm	Best	Worst	Mean	Std. Time(s)
5-unit system				
IA_DED	43716.6386	46611.1333	44879.9749	649.60 2.228s
AIS [8]	44385.43	45553.7707	44758.8363	- 4min
GA [7]	44862.42	-	-	- -
PSO [7]	44253.24	-	-	- -
ABC [7]	44045.83	-	-	- -
HS [14]	44376.23	-	-	- 2.8min
ICA [11]	43117.055	43209.533	43144.472	19.821
DE-SQP [3]	43161	-	-	- -
15-unit system				
IA_DED	759385.9148	759665.1300	759478.5852	55.08 2.08s
NPAHS [12]	759603.089	759988.390	759779.467	- 250.0
CSADHS [2]	759689.220	759845.739	759766.233	- 3.36min
SAMF [6]	759406.42	-	-	- 2.951s
54-unit system				
IA_DED	1718177.0643	1718695.4568	1718424.5910	130.85 8.565s
OCD [15]	1772724.032	-	-	- 0.132s
ICA [11]	1807081.174	1811388.285	1809664.219	- -

At the beginning, the search performed by IA_DED is based on a simple differentiation operator which takes an infeasible solution and modifies some of its decision variables by taking into account their constraint violation. Once the algorithm finds a feasible solution, a redistribution power operator is applied. This operator modifies two decision variables at a time, it decreases the power in one unit, and it selects other unit to generate the power that has been taken, always integer values.

The approach was validated with three test problems having different characteristics and comparisons were provided with respect to some approaches that have been reported in the specialized literature. Our proposed approach produced competitive results in all cases, being able to outperform some approaches while performing a lower number of objective function evaluations.

As part of our future work, we are interested in redesigning the redistribution operator in order to maintain the solutions' feasibility when a problem involves prohibited operating zones.

Acknowledgements

The authors acknowledge support from Universidad Nacional de San Luis, project no. P330214/22F435.

References

1. V.S. Aragon, S.C. Esquivel, and C.A. Coello Coello. An immune algorithm with power redistribution for solving economic dispatch problems. *Information Sciences*, 295(0):609 – 632, 2015.
2. R. Arul, G. Ravi, and S. Velusami. Chaotic self-adaptive differential harmony search algorithm based dynamic economic dispatch. *International Journal of Electrical Power & Energy System*, 50:85–96, 2013.
3. A.M. Elaiw, X. Xia, and A.M. Shehata. Hybrid de-sqp and hybrid pso-sqp methods for solving dynamic economic emission dispatch problem with valve-point effects. *Electric Power Systems Research*, 103:192–200, October 2013. ISSN 0378-7796.
4. M. Shahidehpour [Online]. Available from: motor.ece.iit.edu/data/SCUC_118test.xls, March 2009. [accessed July 2016].
5. Z.L. Gaing and Ting-Chia Ou. Dynamic economic dispatch solution using fast evolutionary programming with swarm direction. In *4th IEEE Conference on Industrial Electronics and Applications. ICIEA 2009*, pages 1538–1544, 2009.
6. S. Ganesan and S. Subramani. Dynamic economic dispatch based on simple algorithm. *International Journal of Computer and Electrical Engineering*, 3(2):1793–8163, 2011.
7. S. Hemamalini and S. Simon. Dynamic economic dispatch using artificial bee colony algorithm for units with valve-point effect. *European Transactions on Electrical Power*, 21:70–81, 2011.
8. S. Hemamalini and S. Simon. Dynamic economic dispatch using artificial immune system for units with valve-point effect. *International Journal of Electrical Power & Energy System*, 33:868–874, 2011.
9. C. Kumar and T. Alwarsamy. Dynamic economic dispatch - a review of solution methodologies. *European Journal of Scientific Research*, 64(4):517–537, 2011.
10. R. Kumar, D. Sharma, and A. Sadu. A hybrid multi-agent based particle swarm optimization algorithm for economic power dispatch. *International Journal of Electrical Power and Energy Systems*, 33(1):115–123, 2011.
11. B. Mohammadi-Ivatloo, A. Rabiee A., Soroudi M., and Ehsan. Iteration {PSO} with time varying acceleration coefficients for solving non-convex economic dispatch problems. *International Journal of Electrical Power & Energy Systems*, 42(1):508 – 516, 2012.
12. Qun Niu, Hongyun Zhang, Kang Li, and George W. Irwin. An efficient harmony search with new pitch adjustment for dynamic economic dispatch. *Energy*, 65:25–43, 2014.
13. Leandro Nunes de Castro and Jonathan Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer-Verlag, New York, 2002.
14. V.R. Pandi and B.K. Panigrahi. Dynamic economic load dispatch using hybrid swarm intelligence based harmony search algorithm. *Expert Systems with Applications*, 38:8509–8514, 2011.
15. Abbas Rabiee, Behnam Mohammadi-Ivatloo, and Mohammad Moradi-Dalvand. Dynamic economic dispatch: A review. *IEEE Transactions on Power Systems*, 29(2):982–983, 2014.
16. X. Xia and A. M. Elaiw. Dynamic economic dispatch: A review. *The Online Journal on Electronics and Electrical Engineering (OJEEE)*, 2(2):234–245, 2009.

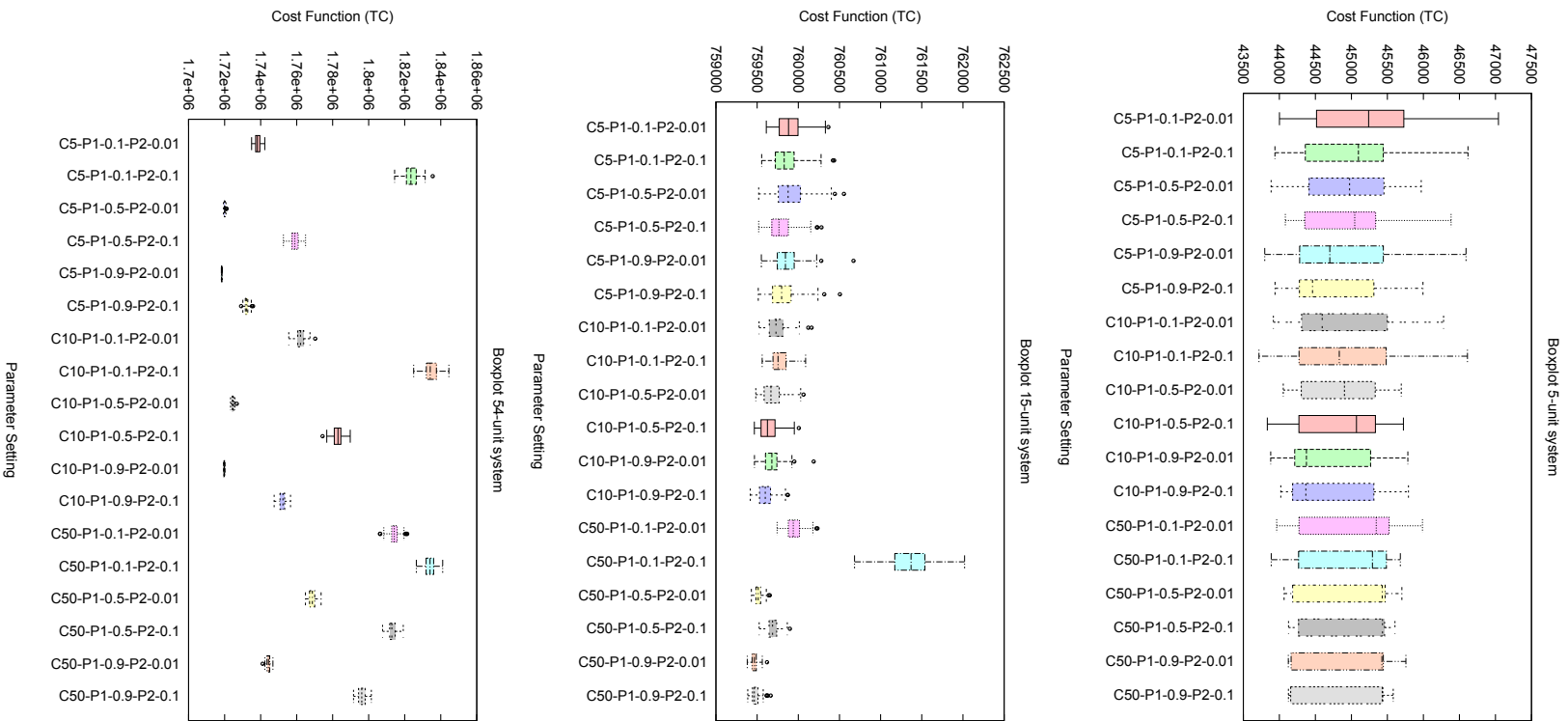


Fig. 1. Box plots for the test problems with the best parameters combination

ASrank: un análisis de robustez

Sergio Fernando Sanz Troianai Gabriela F. Minetti

Universidad Nacional de La Pampa (Argentina),
Facultad de Ingeniería,
Laboratorio de Investigación en Sistemas Inteligentes
e-mail: fstanando@gmail.com, minettig@ing.unlpam.edu.ar

Resumen La resolución de los problemas de optimización en el mundo real es dificultosa no sólo por lo que significa la búsqueda de soluciones en espacios de grandes dimensiones sino, también, por el alto grado de incertidumbre que afecta a los datos de entrada al problema. Para ello son necesarios algoritmos que encuentren soluciones de muy buena calidad en poco tiempo y que además sean robustos. Los algoritmos metaheurísticos han demostrado una y otra vez que son capaces de lograr el primer objetivo, pero no existen muchas investigaciones que estudien y analicen la robustez de los mismos. En este trabajo se analiza la resistencia al ruido (o robustez) de una variante de los algoritmos de optimización basados en colonias de hormigas, denominado *AS_{rank}*. Para ello se utiliza el problema del viajante de comercio, ya que modela un amplio espectro de problemas del mundo real. También se propone *genTSP*, un generador de instancias ruidosas del mencionado problema. El análisis de los resultados obtenidos en la experimentación arroja que el algoritmo estudiado es robusto y no necesita esfuerzo computacional extra para alcanzar este segundo objetivo.

Keywords: Optimización basada en colonia de hormigas, *ASrank*, TSP, robustez

1. Introducción

En los problemas de optimización del mundo real es común encontrar un alto grado de incertidumbre (ruido), debido a la complejidad de la generación de soluciones candidatas, a los errores en los procesos de medición, a los problemas con entornos con un alto grado de dinamismo, a las funciones objetivo ruidosas, entre otros motivos [9]. El efecto del ruido sobre el desempeño de las técnicas de optimización es un área de investigación aún poco estudiada pero de mucha utilidad en diversos campos de aplicación. En el caso de las técnicas metaheurísticas se pueden encontrar estudios sobre: optimización de simulación de sistemas estocásticos complejos [7], problemas del camino más corto estocástico [8], ensamblado de fragmentos de ADN ruidosos [11], problema oneMax [6] y detección de bordes en imágenes con ruido [1].

En particular, los algoritmos de optimización metaheurísticos basados en colonias de hormigas (*Ant Colony Optimization*, ACO) muestran una buena adaptación a los problemas de incertidumbre, ruido y dinamismo debido a su naturaleza dinámica y distribuida [3]. El desarrollo de este trabajo de investigación consiste en analizar la resistencia al ruido, también conocida como robustez, de una variante del algoritmo metaheurístico basado en la optimización de colonia de hormigas, denominado *AS_{rank}* [2]. Para llevar a cabo dicho análisis se trabajará con una de las fuentes de ruido que se presenta con mayor frecuencia en los problemas de optimización del mundo real y que es más difícil de detectar. Esta fuente de ruido surge por los errores cometidos en los procesos de medición, por los cuales se obtienen los datos de entrada a cualquier algoritmo de optimización. Consecuentemente,

se resolverá el problema del viajante de comercio (*Traveling Salesman Problem*, TSP) [13], ya que modela diferentes situaciones reales (planificación de tareas, ruteo de redes de telecomunicaciones, diseño de rutas para la recolección de residuos, diseño de circuitos impresos, etc.) y es de difícil resolución dado que pertenece al grupo de problemas NP-duros.

Para verificar si AS_{rank} es resistente al ruido son necesarias instancias del problema que presenten esta característica. Pero, las instancias de TSP publicadas en la literatura no contemplan ruido en sus datos, por ende en este trabajo también se propone un generador de instancias de TSP que simulen errores en la medición de las distancias. Por consiguiente, para estudiar la robustez de AS_{rank} se analizan las diferencias entre las soluciones encontradas para las instancias sin y con ruido. Si no se detectan diferencias (estadísticamente significativas), el algoritmo muestra un comportamiento neutro (insensible, indistinto) a pequeñas variaciones en los datos de entrada. Consecuentemente, este algoritmo se considera robusto para resolver instancias ruidosas.

El artículo se organiza de la siguiente manera. En la sección 2 se introducen los conceptos generales de AS_{rank} y se explica la versión aquí propuesta. En tanto que, en la sección 3 se describe el problema usado en la experimentación y el generador de instancias ruidosas formulado en este trabajo. En el siguiente apartado se especifica el diseño experimental, el ambiente de ejecución y la metodología de análisis utilizados, para llevar a cabo la experimentación y el análisis de los resultados. En la sección 5 se presentan los resultados obtenidos y el análisis de los mismos. En la última sección se resumen las conclusiones y se proyectan trabajos futuros.

2. Sistema de hormigas basado en jerarquización (*Rank-based Ant System*, AS_{rank})

El sistema de hormigas basado en jerarquización propuesto por Bullenheimer et al. en [2] es una extensión del sistema de hormigas desarrollado por Dorigo et al. en [4]. Para una mejor comprensión de AS_{rank} , primero se presenta una descripción de AS y luego se explica dicha extensión.

AS se caracteriza por el hecho que la actualización de la feromona se realiza una vez que todas las hormigas completaron sus soluciones. Primero, todos los caminos de feromonas son reducidos por un factor constante, implementando de esta manera la evaporación de feromona. Segundo, toda hormiga de la colonia deposita una cantidad de feromona en función de la calidad de su solución. Inicialmente AS no usa acciones de demonio, pero es aconsejable hacerlo al agregar una búsqueda local para refinar las soluciones generadas por las hormigas.

Las soluciones en AS son construidas de la siguiente manera. A cada paso de la construcción, una hormiga k en AS elige ir del nodo r al s con una probabilidad que es calculada como:

$$p_{rs}^k = \begin{cases} \frac{[\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}{\sum_{u \in N_r^k} [\tau_{ru}]^\alpha \cdot [\eta_{ru}]^\beta}, & \text{si } s \in N^k(r) \\ 0, & \text{otro caso} \end{cases}$$

Donde $N_k(r)$ es el vecino factible de la hormiga k cuando está en el nodo r , τ_{rs} es la cantidad de feromona perteneciente a la conexión rs , η_{rs} es uno sobre la distancia entre r y s , y α y $\beta \in \mathbb{R}$ son dos parámetros que pesan la importancia relativa del camino de feromona y la información heurística, respectivamente.

El depósito de feromona es realizado una vez que todas las hormigas han finalizado de construir sus soluciones. Primero, el camino de feromona asociado a cada arco es evaporado reduciendo todas las feromonas por un factor constante, conocido como tasa de evaporación, ($\rho \in (0, 1]$):

$$\tau_{rs} \leftarrow (1 - \rho) \cdot \tau_{rs}$$

Luego, cada hormiga remarca el camino que ha seguido y deposita una cantidad de feromona por cada conexión recorrida:

$$\tau_{rs} \leftarrow \tau_{rs} + \Delta\tau_{rs}^k, \quad \forall a_{rs} \in S_k$$

Donde $\Delta\tau_{rs}^k = f(C(S_k))$, es la cantidad de feromona depositada que depende de la calidad de la solución, $C(S_k)$, construída por la hormiga k .

En AS_{rank} , cada hormiga deposita una cantidad de feromona que disminuye con su rango (o jerarquía), siendo la mejor hormiga de cada iteración la que deposita la mayor cantidad de feromona. Esta actualización se realiza globalmente como sigue:

1. Se seleccionan las w hormigas para la jerarquización de acuerdo a la calidad de sus soluciones: (S'_1, \dots, S'_w) , siendo S'_1 la mejor solución generada hasta el momento.
2. Se deposita feromona en las conexiones recorridas por las mejores hormigas (denominadas hormigas elitistas). La cantidad de feromona depositada depende directamente de la clasificación de la hormiga y de la calidad de la solución. Es decir, que la feromona depositada por la hormiga clasificada en la posición r se multiplica por $(w - r)$.
3. Las conexiones recorridas por la mejor solución global reciben una cantidad adicional de feromona que depende de la calidad de la solución. Este depósito de feromona es considerado el más importante, por lo tanto se multiplica por w .

En la versión de AS_{rank} implementada en este trabajo (ver algoritmo 1), primero se inicializan los parámetros y el rastro de feromona, en un segundo paso, cada hormiga construye, iterativamente, una solución al problema. A continuación, AS_{rank} , aplica el algoritmo de búsqueda local 3 – *opt* a cada una de estas soluciones. Finalmente, aplica los tres pasos que caracterizan a AS_{rank} (descriptos en el párrafo anterior) y vuelve al paso dos hasta que el número máximo de iteraciones se haya ejecutado.

Algoritmo 1 AS_{rank}

Inicializar parámetros y rastro de feromonas (τ);

$t = 0$;

repeat

for $i = 0$ to μ **do**

 { μ es la cantidad de hormigas}

 La hormiga i construye paso a paso una solución;

 Aplicar 3 – *opt* a cada solución;

end for

 Elegir las w mejores soluciones y ordenarlas de forma decreciente (S'_1, \dots, S'_w) ;

 Actualizar el rastro de feromonas:

- 1) Depositar feromona en las conexiones recorridas por cada S'_r en función a su calidad y a su posición;
- 2) Depositar feromona en las conexiones recorridas por la mejor solución global en función a su calidad;

$t = t + 1$;

until $t = K$

return S'_1 ;

3. Generación de instancias con ruido

Como se dijo anteriormente, para determinar si AS_{rank} es robusto frente a errores en los datos de entrada del problema se requieren instancias de TSP que presenten dicha característica. Dado que las instancias de TSP publicadas en la literatura no introducen errores en sus datos, se propone un generador de instancias de TSP, denominado *genTSP*, que simule los errores en la medición de las distancias. Consecuentemente, en esta sección se introduce *genTSP* y las nuevas instancias generadas, siendo necesario introducir previamente la definición del problema del viajante de comercio, las instancias sin ruido publicadas en la literatura y la descripción de los errores que se cometen en la medición de distancias.

El problema del viajante de comercio puede ser descrito como: dado un conjunto de ciudades $C_1, C_2, C_3, \dots, C_n$ y un conjunto de distancias $d(C_i, C_j)$ para cada par de ciudades distintas (C_i, C_j) , consiste en encontrar un recorrido de distancia mínima que pase sólo una vez por cada una de las ciudades y, finalmente, regrese a la ciudad de la que partió. A fin de obtener un conjunto de prueba de significativa complejidad, en este trabajo se seleccionaron instancias de TSP de variadas características y con gran cantidad de ciudades.

Las instancias en cuestión se muestran en la tabla 1 y pueden descargarse de la librería TSPLIB¹. En dicha tabla se especifica el nombre, la dimensión y el valor óptimo para cada una de las instancias. Es importante aclarar que estas instancias serán identificadas con la etiqueta “0%”, ya que son los casos sin ruido.

<i>Nombre</i>	<i>Dimensión</i>	<i>Valor óptimo</i>
d198	198	15780
a280	280	2579
lin318	318	42029
pcb442	442	50778
rat783	783	8806
pcb1173	1173	56892
d1291	1291	50801
fl1577	1577	22249

Cuadro 1. Instancias seleccionadas de TSP de la librería TSPLIB.

Los errores en la medición surgen dependiendo del método o instrumento utilizado para determinar la distancia entre las ciudades. Existen diversos instrumentos para medir distancias [10], pero para obtener la longitud de las rutas se utilizan generalmente odómetros mecánicos o eléctricos y, también, sistemas de información geográfica (SIGs). En el proceso de medición se introducen errores ya sea por una mala calibración de los instrumentos o los denominados errores sistemáticos y aleatorios que son inevitables. Los errores sistemáticos están relacionados con la pendiente, la graduación, la temperatura, la tensión y la catenaria. En tanto que, los errores aleatorios se vinculan con la alineación y la verticalidad. Además están los errores llamados groseros, que pueden cometerse por equivocación del operador y que son impredecibles. Entre los más comunes se pueden

¹ <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/tsp/>

encontrar: la identificación incorrecta de un punto, la lectura y/o anotación errónea por transposición de números y los cálculos aritméticos mentales inexactos. Por último es necesario considerar los errores accidentales ya que son totalmente inevitables y sus causas no son conocidas.

En virtud de todos los tipos de errores presentes en la medición de distancias, en este trabajo se consideran cuatro niveles de ruido en las instancias de TSP. De esta manera, *genTSP* crea 4 nuevos grupos de instancias a partir de los casos sin ruido e incorpora a cada una de las distancias errores de hasta un 5, 10, 15 y 20%. Estos nuevos cuatro grupos de instancias son identificados con las etiquetas “5%”, “10%”, “15%” y “20%”, respectivamente. El algoritmo 2 muestra un pseudocódigo del generador *genTSP*. Este utiliza la distribución uniforme para crear en forma aleatoria un error $\epsilon \in [-l, l]$ para cada distancia entre dos ciudades. Donde $l = \gamma \times d(C_i, C_j) \forall i \neq j, i, j \leq n$ y $\gamma \in \{5\%, 10\%, 15\%, 20\%\}$, los niveles de ruido.

Algoritmo 2 *genTSP*

```

for  $i = 0$  to  $n$  do
  for  $j = 0$  to  $n$  do
    if  $i \neq j$  then
       $l = \gamma \times d(C_i, C_j)$ ;
       $\epsilon = \text{uniform.random}(-l, l)$ ;
       $d(C_i, C_j) = d(C_i, C_j) + \epsilon$ ;
    end if
  end for
end for
return

```

4. Metodología de experimentación

En esta sección se describe el diseño experimental usado en este trabajo, el ambiente de ejecución y la metodología de análisis. La variante de AS_{rank} propuesta se evaluó con el grupo de las instancias sin ruido y los cuatro grupos de instancias con ruido, sumando un total de 40 instancias ($8 \text{ instancias} \times 5 \text{ grupos}$). Además al considerarse la naturaleza estocástica de los algoritmos metaheurísticos, cada instancia fue resuelta 30 veces por esta versión de AS_{rank} .

Los valores paramétricos, $\alpha = 1$ y $\beta = 2$ utilizados para evaluar el rendimiento de esta metaheurística son determinados según lo publicado en [5]. En tanto que, los parámetros $\rho = 0,2$, $w = 8$, el tamaño de la población de hormigas $\mu = 25$, el número de iteraciones realizadas por la búsqueda local, $k = 5000$ y el número total de iteraciones llevado a cabo por AS_{rank} , $K = 10000$ surgieron de la combinación de diferentes valores que arrojó los mejores resultados en una experimentación previa. Durante la misma, se ejecutó AS_{rank} bajo todas las configuraciones que surgen al combinar los diferentes valores propuestos para cada uno de estos parámetros. El conjunto de valores considerados son: para $\rho = \{0,1,0,2,0,3\}$, para $\sigma = \{7,8,9\}$, para $\mu = \{25,100\}$, para $k = \{0,1000,5000\}$ y, finalmente, para $K = \{1000,10000\}$. En tanto que, el conjunto de prueba está formado por las instancias descritas en la sección 3.

El algoritmo fue implementado en C++. Esta experimentación se lleva a cabo sobre computadoras con procesadores AMD Phenom (64 bits) a 2.4 GHz y 2 GB de RAM, con un sistema operativo perteneciente a la distribución Slackware de Linux con una versión del kernel 2.6.27.7-smp.

Para poder comparar los resultados obtenidos para las distintas instancias se mide la desviación relativa, DR , de la distancia recorrida (costo) por las mejores soluciones encontradas por el algoritmo

con respecto al valor óptimo de cada instancia. Además de posibilitar la comparación al brindar datos normalizados, esta métrica también permite analizar la calidad de las soluciones halladas. El cálculo de los valores de DR se describe en la Ecuación 1, donde $AlgSol$ es el costo mínimo obtenido por AS_{rank} para una instancia del problema en las 30 ejecuciones y $OptSol$ es el costo óptimo para la instancia en cuestión.

$$DR = \frac{AlgSol - OptSol}{OptSol} \quad (1)$$

Con el objetivo de evaluar la robustez de AS_{rank} , se analizan las diferencias entre las soluciones encontradas para las instancias sin y con ruido. Si no se detectan diferencias (estadísticamente significativas), el algoritmo muestra un comportamiento neutro (insensible, indistinto) a pequeñas variaciones en los datos de entrada. Consecuentemente, este algoritmo se considera robusto para resolver instancias ruidosas. Este análisis se basa en los valores de DR obtenidos para las instancias sin y con ruido, que luego son usados para comparar la calidad de las soluciones halladas en ambos casos. Por ende, si estadísticamente se detectan soluciones de calidad similar ante instancias sin y con ruido se infiere que AS_{rank} es un algoritmo resistente al ruido.

Lo anterior se traduce en la comparación estadística de las medias de dos muestras independientes. En nuestro caso sería comparar los resultados obtenidos para instancias con 0 % de ruido con los encontrados para las instancias con 5 %, luego contrastar los resultados hallados para instancias con 0 y 10 % , con 0 y 15 % y, finalmente, con 0 y 20 % de ruido. De esta forma, si para un par de muestras el test estadístico arroja que no hay diferencias significativas entre ellas significa que el algoritmo es robusto en la resolución de instancias con el nivel de ruido en cuestión.

Dado que se comparan dos muestras se emplean t-tests o el test Mann-Whitney U, según sea el tipo de distribución de datos de tales muestras. El t-test no puede aplicarse sobre cualquier distribución de datos, puesto que en el caso de una distribución no normal de datos puede ser erróneo el resultado. En este caso, lo correcto sería aplicar el test Mann-Whitney U. Para determinar si los datos siguen una distribución normal se aplican la prueba de Shapiro-Wilks y homogeneidad de varianzas (homocedasticidad). Cabe aclarar que para todos los tests estadísticos realizados en este trabajo, el valor de α se fija en 0.01, para indicar un nivel de confianza del 99 % en los resultados. Por ejemplo, si el t-test arroja un valor $p \geq \alpha$ significa que las medias de las dos muestras son estadísticamente similares, en cambio si $p < \alpha$ entonces se trata de muestras significativamente distintas.

Además de estudiar la robustez, se analiza el esfuerzo numérico realizado por el algoritmo considerando el número de iteraciones que fueron necesarias para encontrar la mejor solución. Para ello se utilizan los diagramas de caja [12], ya que facilitan la comparación entre los resultados obtenidos por un algoritmo para las instancias sin y con ruido, y permiten establecer si existen o no diferencias estadísticamente significativas entre ellos.

5. Analisis de los resultados obtenidos

En la primera parte de esta sección se analiza la robustez de AS_{rank} por medio de los resultados arrojados por los tests estadísticos. En tanto que, en la segunda se compara el esfuerzo computacional realizado por el algoritmo al solucionar el problema con distintos niveles de ruido.

5.1. Análisis de robustez

La figura 1 muestra los valores p arrojados por el t-test o el test Mann-Whitney U, al comparar la desviación relativa (DR) de cada una de las instancias sin ruido (0%) con sus pares en los diferentes niveles de ruido (5, 10, 15 y 20%, respectivamente). También indica el valor de $\alpha = 0,01$ con propósitos comparativos mediante una recta.

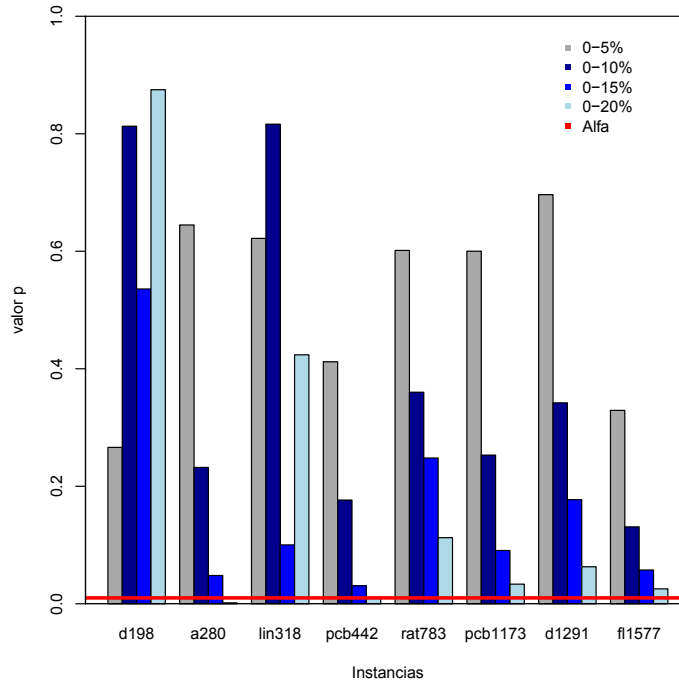


Figura 1. Valores p arrojados por las pruebas estadísticas al comparar la distancia relativa de las instancias con 0, 5, 10, 15 y 20% de ruido.

En primer lugar es fácilmente observable que hasta niveles del 15% de error y para todas las instancias, los valores p son todos mayores a α . Esto indica que los resultados obtenidos son estadísticamente similares, comprobándose la robustez de AS_{rank} para niveles de 5, 10 y 15% de ruido. En segundo plano se puede detectar que si el nivel de error es de un 20%, AS_{rank} es robusto para el 75% de las instancias. Sólo para *a280* y *pcb442*, el algoritmo no logra ser resistente al ruido con este alto nivel de ruido.

Del análisis realizado se desprende que AS_{rank} es un método de optimización robusto, ya que la resistencia al ruido es estadísticamente corroborada en instancias del problema de diferente complejidad y con diversos niveles de ruido. Estos últimos van desde niveles de error muy bajos (5%) a muy altos (20%).

5.2. Comparación del esfuerzo computacional

En el apartado anterior se verificó la característica de robustez de AS_{rank} , a continuación se analiza si esta importante ventaja del algoritmo implicaría un mayor esfuerzo computacional. Para ello se compara la cantidad de iteraciones necesarias para que ASrank encuentre la mejor solución en cada nivel de ruido, considerando todas las instancias. Esta comparación se desprende de los datos representados en el diagrama de cajas de la figura 2, donde también puede observarse si las diferencias son o no estadísticamente significativas.

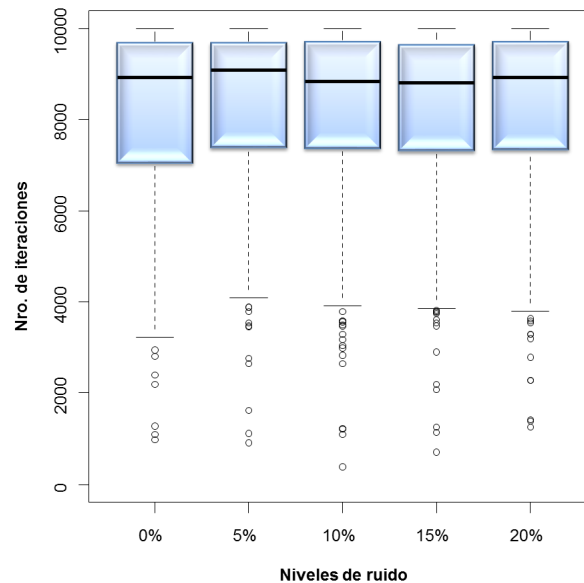


Figura 2. Diagramas de cajas (boxplots) que representan la cantidad de iteraciones utilizada por ASrank para encontrar la mejor solución frente a los diferentes niveles de ruido, comenzando por los casos sin ruido (0%).

Del análisis de la figura 2 se concluye que para todos los niveles de ruido (de 0 a 20%), el 75% de las mejores soluciones son encontradas después de las 7000 iteraciones y que, los valores promedios varían entre 8000 y 8250 iteraciones. Estos indicadores muestran fehacientemente que el esfuerzo computacional realizado por AS_{rank} para resolver los casos sin y con errores son estadísticamente similares. En consecuencia, AS_{rank} no sólo es resistente al ruido sino que además no necesita de un mayor esfuerzo numérico.

6. Conclusiones

En este trabajo se introduce un generador de instancias ruidosas correspondientes al TSP, con el cual se crean un conjuntos de instancias con cuatro distintos niveles de ruido. Estas nuevas instancias

son usadas para verificar la resistencia al ruido o robustez de un algoritmo metaheurístico basado en colonia de hormigas, AS_{rank} , frente a la variación en los datos de entrada.

El análisis realizado sobre los datos obtenidos en la experimentación arroja que AS_{rank} es un método de optimización robusto, ya que se corrobora estadísticamente la resistencia al ruido en instancias del problema de diferente complejidad y con diversos niveles de ruido. Estos últimos van desde niveles de error muy bajos (5 %) a muy altos (20 %). Pero, AS_{rank} no sólo es resistente al ruido sino que además no necesita de un mayor esfuerzo numérico, ya que que el esfuerzo computacional realizado para resolver los casos sin y con errores son estadísticamente similares.

Como trabajo futuro, en primer lugar se pretende utilizar los grupos de instancias generadas con ruido para analizar la robustez de otros algoritmos de optimización basados en colonia de hormigas. En segundo lugar se intentará mejorar la resolución de situaciones reales al aplicar el conocimiento adquirido en estos análisis.

Agradecimientos

Los autores y el trabajo de investigación son financiados por la Universidad Nacional de La Pampa y la ANPCyT por medio del contrato PICTO 2011-0278 y del programa de incentivos.

Referencias

1. Bateria, A.V., Oppus, C.: Image edge detection using ant colony optimization. WSEAS Trans. Sig. Proc. 6(2), 58–67 (Apr 2010), <http://dl.acm.org/citation.cfm?id=1853856.1853860>
2. Bullheimer, B., Hartl, R.F., Strau C.: A new rank based version of the ant system – a computational study (1997)
3. DiCaro, G.A., Ducatelle, F., Gambardella, L.M.: Reflecting Interfaces: the Complex Coevolution of Information Technology Ecosystems, chap. Theory and practice of Ant Colony Optimization for routing in dynamic telecommunications networks. Idea Group, Hershey, PA, USA (2008)
4. Dorigo, M., Maniezzo, V., Coloni, A.: The ant system: Optimization by a colony of cooperating agents. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART B 26(1), 29–41 (1996)
5. Dorigo, M., Stützle, T.: Ant Colony Optimization. Bradford Company, Scituate, MA, USA (2004)
6. Friedrich, T., Kötzing, T., Krejca, M.S., Sutton, A.M.: Robustness of ant colony optimization to noise. In: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation. pp. 17–24. GECCO '15, ACM, New York, NY, USA (2015), <http://doi.acm.org/10.1145/2739480.2754723>
7. Gosavi, A.: The effect of noise on artificial intelligence and meta-heuristic techniques. In: of Mechanical Engineering, A.S. (ed.) Proceedings of the Artificial Neural Networks in Engineering Conference (Intelligent Engineering systems through Artificial Neural Networks). vol. 12, pp. 981–988 (2002)
8. Horoba, C., Sudholt, D.: Ant colony optimization for stochastic shortest path problems. In: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation. pp. 1465–1472. GECCO '10, ACM, New York, NY, USA (2010), <http://doi.acm.org/10.1145/1830483.1830750>
9. Jin, Y., Branke, J.: Evolutionary optimization in uncertain environments—a survey. Trans. Evol. Comp 9(3), 303–317 (Jun 2005), <http://dx.doi.org/10.1109/TEVC.2005.846356>
10. Matera, L.C.: Topografía Plana. Facultad de Ingeniere la Universidad de Los Andes (2002)
11. Minetti, G., Leguizamón, G., Alba, E.: An improved trajectory-based hybrid metaheuristic applied to the noisy DNA Fragment Assembly Problem. Information Sciences 277, 273 – 283 (2014), <http://www.sciencedirect.com/science/article/pii/S0020025514001261>
12. R, M., Tukey, J.W., Larsen, W.A.: Variations of box plots. The American Statistician 32(1), 12–16 (1978)
13. Shmoys, D., Lenstra, J., Kan, A., Lawler, E.: The Traveling Salesman Problem. Wiley Interscience Series in Discrete Mathematics, John Wiley & Sons (1985)

Optimización de cadenas de adición

Fernando Aquino¹ y Guillermo Leguizamón²

¹ Universidad Tecnológica Nacional - Facultad Regional Concepción del Uruguay
fernando.aquino@hotmail.com.ar

² Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)
Universidad Nacional de San Luis
legui@unsl.edu.ar

Resumen El presente estudio aborda el problema de la computación óptima de cadenas de adición, ampliamente tratado con diferentes métodos y enfoques (tanto deterministas como estocásticos) y de interés en el ámbito de la criptografía. En este trabajo, se propone el uso del algoritmo de lobos grises o GWO (por sus siglas en inglés: Grey Wolf Optimizer) para hacer frente a este problema a fin de comparar los resultados obtenidos con otros enfoques del estado del arte. Si bien el problema en cuestión ha sido tratado mediante diferentes estrategias y para distintos tipos de exponentes, particularmente esta propuesta se centra en la optimización de cadenas de adición, asociadas a exponentes de tamaño moderado.

1. Introducción

Los algoritmos asimétricos de criptografía utilizan la exponenciación modular para cifrar y descifrar datos. El costo computacional producto de la gran cantidad de operaciones de multiplicación, que implica resolver potencias de número de orden considerable, se puede solventar mediante el uso del concepto de “cadenas de adición” [2]. Sin embargo, un exponente no posee una única cadena de adición válida asociada y por tanto hallar una cadena válida y cuya longitud resulte mínima constituye un problema NP completo. Se denomina exponenciación modular [8] a la operación cuya finalidad es obtener un valor b (entero, positivo) para satisfacer la siguiente ecuación:

$$b = A e \pmod{P} \quad (1)$$

Dado un valor de A entero, arbitrario en el rango $[1, P - 1]$ y un número e entero, positivo y arbitrario, se define exponenciación modular al problema de encontrar un valor único y entero positivo $b \in [1, P - 1]$, capaz de satisfacer la ecuación (1), siendo para el problema estudiado aquí, la función objetivo a optimizar para poder obtener un valor b mínimo (i.e., una cadena de adición de longitud mínima.) El problema radica en que a medida que se incrementa el exponente (e), aumenta el número de operaciones a realizar para resolver la ecuación (1), con lo cual se requiere una alternativa para optimizar el proceso: La alternativa es emplear cadenas de adición para poder reducir el número de

multiplicaciones para resolver una potencia. Ejemplo: dado el exponente $e = 91$, en vez de multiplicar la base 91 veces es factible contar con una cadena asociada al número 91, como ser: $C = \{1, 2, 4, 6, 8, 14, 22, 36, 58, 80, 88, 90, 91\}$ y entonces resolver la potencia consistiría en multiplicar la base por cada uno de los exponentes de C , es decir que para este ejemplo se realizan sólo 13 operaciones.

Sin embargo y teniendo en cuenta que no existe una única cadena para un exponente, hallar una cadena de longitud mínima, constituye un problema de optimización combinatoria y puede abordarse empleando una metaheurística para evitar las pruebas por fuerza bruta para hallar un valor óptimo (cadena de longitud mínima).

1.1. Antecedentes

A continuación y a modo de síntesis, se presenta un resumen del estado del arte del problema objeto de estudio en el campo de las metaheurísticas. Se resume en el Cuadro 1 los resultados de estudios anteriores en donde se abordó el problema de optimización de la función objetivo definida por la ecuación (1) mediante el uso de distintas metaheurísticas ampliamente conocidas, ordenadas de forma cronológica por año e indicando los rangos del exponente «e» empleados para cada enfoque:

Cuadro 1. Resumen del estado del arte de la problemática objeto de estudio.

MH	ACO	GA	AIS	PSO	EP	GA Híbrido	GA
	[7]	[8]	[13]	[14]	[17]	[22]	[23]
Año	2004	2005	2008	2009	2011	2011	2016
$e \in$	Rangos de exponentes						
< 128	X	X	X	-	X	X	X
[128, 512]	X	-	X	-	X		
[512, 1024]	X	-	X	-			

Considerando que la presente propuesta se centra en el uso de la metaheurística GWO como estrategia de tratamiento para la problemática de generación de cadenas de adición, resulta interesante comparar con propuestas del estado del arte también basadas en metaheurísticas. Nedjah y De Macedo-Mourelle en 2004, experimentaron con ACO [7] para resolver el problema de hallar cadenas de longitud óptima, trabajando con exponentes de hasta una longitud de 128-bits de longitud, no abordando pruebas con exponentes de longitud mayor a dicho rango, hasta un trabajo posterior donde aplicaron ACO a exponentes de hasta 1024-bits. Cruz-Cortés y otros, en 2005, emplearon GA [8] logrando hallar cadenas óptimas para exponentes de todos los rangos menores a 4096 ($e < 4096$). En 2008 Rodríguez-Henríquez y Coello Coello, propusieron AIS [13], consiguiendo obtener cadenas mínimas para $e \in [1, 4096]$ pudiendo obtenerse

resultados para todos los rangos pero mejores para aquellos de longitud acotada entre [128,512], dejando propuesto como trabajo futuro lograr mejores resultados para exponentes grandes. León-Javier en 2009 implementó PSO [14] pero sólo reportó experimentos con exponentes pequeños, dejando los otros casos como trabajo pendiente. Isidro Domínguez en 2011 en su tesis adaptó el algoritmo EP [17] logrando minimizar cadenas para exponentes de los rangos 128, 256, 512 y 1024 bits. El mismo año se propuso una alternativa de hibridación de GA con “Simulated Annealing” [22] mejorando con esto la convergencia del AG, pero sin embargo los resultados no superan a otras propuestas anteriores como el caso de PSO con la cual se comparó su desempeño. Finalmente en 2016 se propuso un GA con una serie de mejoras en los operadores de variación, una representación novedosa de soluciones y una estrategia de reparación de soluciones [23]. Se optimizó exponentes en el rango $[2^{37}-3, 2^{127}-3]$.

2. Algoritmo basado en el comportamiento de lobos grises (Grey Wolf Optimizer)

En [21] se propone una metaheurística dominada Grey Wolf Optimizer (GWO) inspirada en el comportamiento de las manadas de lobos grises (*Canis Lupus*) y su organización social para la caza de presas. Los lobos grises por naturaleza prefieren vivir en grupo de entre 5 y 12 miembros en promedio. La manada se conforma de dos líderes llamados alfa (un macho y una hembra) y el resto de la manada. En esta estructura social, los miembros alfa son los responsables de la toma de decisiones sobre la caza, lugar donde dormir, hora de despertar, y el resto de los miembros debe seguir sus órdenes. El miembro alfa de una manada no necesariamente debe ser el más fuerte, pero sí el mejor en términos de conducción del grupo y capacidad para tomar decisiones. El segundo nivel en la jerarquía del grupo son los miembros beta (subordinados de los alfa), que deben colaborar con la toma de decisiones y el cumplimiento de las órdenes de los alfa. Estos miembros son los candidatos a convertirse en alfa cuando alguno de estos fallece o envejezca. Existe otra clasificación que se encuentra más abajo en la jerarquía de las manadas de lobos grises y se denomina omega, éstos deben someterse a los alfa y betas obedeciéndoles, en muchos casos suele verse en este tipo a miembros que offician de niñeros de las crías, dentro de la manada [24]. Las principales fases de la caza en manadas de lobo gris son las siguientes:

- Seguir, perseguir, y acercarse a la presa.
- Perseguir, rodear, y acosar a la presa hasta que se “paraliza”.
- Atacar a la presa.

Esta técnica de caza fue modelada matemáticamente en [21] con la finalidad de proponer un método de optimización y emplearlo en la resolución de problemas complejos.

2.1. Modelo matemático y algoritmo

A continuación se describe el modelo matemático de la jerarquía social de las manadas del lobo gris en base a las tres acciones principales relacionadas con la actividad de la caza: seguir, rodear (acorrallar) y atacar.

Jerarquía social de las manadas:

Para modelar matemáticamente la jerarquía social de los lobos en el diseño de GWO, se considera que la solución más apta para el problema que se intenta resolver es la alfa (α). En consecuencia, la segunda y tercera mejores soluciones se denominan beta (β) y delta (δ), respectivamente. El resto de las soluciones candidatas, se supone que son omega (ω). En el algoritmo GWO la actividad de caza se traduce como proceso de optimización, y el mismo es guiado por α , β , y δ , mientras que los lobos ω siguen a los 3 tipos previamente descriptos.

Acorrallamiento de la presa:

Los lobos grises rodean la presa durante la caza. Matemáticamente este comportamiento se modela de la siguiente manera:

$$D = |C \cdot X_p(t) - X(t)| \quad (2a)$$

$$X(t+1) = X_p(t) - A \cdot D \quad (2b)$$

Donde t indica la iteración actual, A y C son vectores de coeficientes, X_p es el vector de posición de la presa, y X indica la posición vector de un determinado lobo gris. Por otro lado, los vectores A y C se calculan como sigue:

$$A = 2ar_1 - a \quad (3a)$$

$$C = 2r_2 \quad (3b)$$

Según la formulación original GWO en [21] los componentes del vector A disminuyen linealmente desde 2 hasta 0, sobre el curso de las iteraciones y r_1 , r_2 son vectores aleatorios en el rango $[0, 1]$.

El modelo matemático de GWO describe la operación de acercamiento en función de que un lobo gris en una determinada posición (en el espacio n -dimensional), podrá actualizar su posición respecto de la presa, trasladando la ubicación del lobo (o agente) mediante el ajuste del valor de los vectores A y C .

Caza (ataque a la presa):

Los lobos grises tienen la capacidad de reconocer la ubicación de la presa y rodearla. En la naturaleza la caza suele ser guiada por el miembro alfa, aunque los miembros beta y delta que le siguen en jerarquía, pueden participar eventualmente de la caza. Con el fin de simular matemáticamente el comportamiento de caza de los lobos grises, suponemos que el miembro alfa (mejor solución candidata), beta y delta tienen un mejor conocimiento sobre la posible ubicación

de la presa. Por lo tanto, guardamos las primeras tres mejores soluciones obtenidas hasta el momento y obligamos a los otros agentes de búsqueda (incluidos los omegas) a actualizar sus posiciones de acuerdo a la posición de los mejores agentes de búsqueda. En este sentido se proponen las siguientes fórmulas:

$$D_\alpha = |C_1 X_\alpha - X|, D_\beta = |C_2 X_\beta - X|, D_\delta = |C_3 X_\delta - X| \quad (4a)$$

$$X_1 = X_\alpha - A_1(D_\alpha), X_2 = X_\beta - A_2(D_\beta), X_3 = X_\delta - A_3(D_\delta) \quad (4b)$$

$$X(t+1) = (X_1 + X_2 + X_3)/3 \quad (4c)$$

En función de que cada agente actualiza su posición según: alfa, beta y delta en un espacio de búsqueda, se puede decir que el rol de alfa, beta y delta es estimar la posición de la presa y los otros lobos actualizan sus posiciones al azar alrededor de la misma.

Exploración (búsqueda) versus Explotación (ataque):

Los lobos se acercan a la presa y la atacan cuando la misma deja de moverse. El modelo matemático de tal comportamiento de acercarse a la presa, se traduce como la disminución del valor del vector A (coeficiente). Cabe destacar que el rango de fluctuación del vector A , también se reduce por el valor de “ a ” (3a). Cuando $|A| < 1$, GWO, obliga a los lobos a atacar a la presa. Esta característica está relacionada con la explotación del algoritmo.

GWO es propenso al estancamiento en las soluciones locales con estos operadores. Y si bien es cierto que el mecanismo de “rodeo” que posee genera exploración hasta cierto punto, necesita más operadores para enfatizar dicha exploración.

Los lobos tienen posiciones diferentes respecto de la presa pero convergen a la misma para atacar. El modelo matemático de esta divergencia se refleja en el uso de valores aleatorios para el vector A (mayores que 1 y menores que -1) para asegurar divergencia y que la búsqueda abarque de mejor manera el espacio de búsqueda, característica relacionada con la exploración, es decir que cuando $|A| > 1$ se está obligando a los lobos a divergir respecto de la presa.

En resumen, el proceso de búsqueda en GWO se inicia con la creación de población de lobos al azar (soluciones candidatas). En el transcurso de iteraciones, los lobos: alfa, beta y delta, estiman la posición probable de la presa. Cada solución candidata actualiza su distancia respecto de la presa. El parámetro “ a ” es disminuido desde 2 hasta 0 con el fin de enfatizar la exploración y explotación, respectivamente. Las soluciones candidatas divergen respecto de la presa cuando $|A| > 1$ y convergen hacia la presa cuando $|A| < 1$. Por último, el algoritmo GWO termina por la satisfacción de un criterio de fin. Tal como describe en [21] GWO ha sido aplicado a una serie de problemas de optimización y dentro de ellos se aplicó a un problema real del ámbito de la ingeniería óptica [21]. En el Algoritmo 1 se presenta el esquema general del método.

Algorithm 1 Pseudo código del algoritmo GWO.

```

Inicializar población de lobos  $X_i$  (con  $i = 1, 2, \dots, n$ )
Inicializar  $\alpha$ ,  $A$  y  $C$ 
Calcular fitness de cada agente de búsqueda
 $X_\alpha$  = el mejor agente.
 $X_\beta$  = el segundo mejor agente.
 $X_\delta$  = el tercer mejor agente.
while (  $t <$  número máximo de iteraciones) do
  for  $i = [1..n]$  do
    Actualizar la posición de cada agente mediante la ecuación (4c)
  end for
  Actualizar  $\alpha$ ,  $A$  y  $C$ 
  Calcular el fitness de todos los agentes de búsqueda
  Actualizar  $X_\alpha$ ,  $X_\beta$  y  $X_\delta$ 
   $t=t+1$ 
end while
Retornar  $X_\alpha$ 

```

2.2. Propuesta

El objetivo del presente trabajo es abordar el problema de optimización de generación de cadenas de adición cuya longitud sea mínima y abordarlo con una metaheurística no explorada o es mente explorada para la problemática como lo es GWO y la cual es de más reciente surgimiento que las ya utilizadas para el problema de referencia. Se empleó una función programada sobre lenguaje MATLAB©, capaz de generar cadenas de adición asociada a exponentes que recibe como parámetros de entrada [17] (ver Algoritmo 2) y la misma fue optimizada mediante GWO.

Algorithm 2 Producir cadena de adición válida.

```

Asignar a  $u_0 = 1$  y  $u_1 = 2$ 
Seleccionar de manera aleatoria 3 o 4 y asignarlo a  $u_2$ .
Completar la secuencia invocando la función  $FILL(U, Rand(3,4),e)$  // parámetros que emplea la función para "llenar" la cadena.
Retornar (U, L)

```

En el Algoritmo 2, e es el exponente para el cual se genera la cadena de adición, $U : (u_0, u_1, u_2, \dots, e)$ es la cadena de adición completa de longitud L , $Rand(3,4)$ representa el método que toma de manera aleatoria el valor 3 o 4 para generar la tercera componente de la cadena y $FILL$ es la función llenar, que se utiliza para completar la cadena y se describe en el Algoritmo 3.

Al igual que en [17] los individuos de la población serán generados aplicando las 3 reglas mencionadas a continuación, siempre con la premisa de que las dos primeras componentes de la cadena son 1 y 2, en este mismo orden:

Algorithm 3 FILL($U, \text{Rand}(3,4), e$) // completar cadena.

```

Establecer  $i = k - 1$ 
while  $U_i \neq e$  // mientras la posición de la cadena sea distinta de  $e$  do
  if flip(f) then
     $u_{i+1} = 2u_i$  // aplicar doblado (regla 1)
  else
    if FLIP(g) then
       $u_{i+1} = u_i + u_{i-1}$  // aplicar (regla 2)
    else
       $u_{i+1} = u_i + u_{rand}$  // aplicar (regla 3)
    end if
  end if
  while  $(u_{i+1}) > e$  // Para que la siguiente posición sea mayor que  $e$  do
     $j = i - 1$ 
     $u_{i+1} = u_i + u_{i-j}$ 
     $j = j - 1$  // se decrementa para no superar el valor de  $e$ .
  end while
end while
Retornar ( $U$ )

```

1. Se permite que un elemento de la cadena resulte de sumar al elemento previo por sí mismo. De este modo, en una cadena U , un componente u_{i+1} podrá ser igual a: $2u_i \equiv (u_{i+1} = u_i + u_i)$. Por ejemplo, para la cadena 1, 3, 6, 9, 12, 15, 21, el tercer elemento se compone de sumar el segundo elemento consigo mismo.
2. Suma de dos números previos, de modo que: $u_{i+1} = u_i + u_{i-1}$. Tomando como ejemplo la misma cadena que en punto 1, el cuarto elemento (9) es igual al segundo sumado al tercero.
3. Suma de un número previo más un número aleatorio de la cadena, es decir: $u_{i+1} = u_i + u_{rand}$. Siguiendo el mismo ejemplo anterior: el elemento (15) se conforma de la suma del quinto (12) y el segundo elemento (3).

Para adaptar el comportamiento de GWO para resolver la problemática objeto de estudio, vamos a establecer que nuestra población de lobos (soluciones) es una población de cadenas de adición válidas para un exponente e determinado, es decir secuencias numéricas de enteros y que responden a las reglas mencionadas anteriormente. Por otro lado, teniendo en cuenta que el propósito de optimizar cadenas para un exponente determinado consiste en obtener cadenas de longitud mínima, la función de adaptación (fitness) a utilizar, será la longitud de las cadenas.

De esta manera, la función FILL aplica las reglas anteriores con determinados valores de probabilidad. Para los experimentos se ha tomado como referencia los valores reportados por el enfoque [17] que ha demostrado resultados competitivos para varios rangos de exponentes, a fin de poder comparar nuestra propuesta bajo condiciones similares en términos de generación de soluciones y desempeño de la metaheurística propuesta. Dichos valores son:

- Tasa de determinación de un elemento de la cadena a partir de la suma del número anterior por sí mismo (regla 1) = 0.7
- Tasa de suma de elementos anteriores (regla 2) = 0.2

En la siguiente sección, se presentan los resultados vertidos de una serie de pruebas estadísticas realizadas sobre la metaheurística GWO aplicada al problema objeto de estudio. Todos los resultados obtenidos con el enfoque GWO, han sido generados aplicando la siguiente configuración de parámetros:

- SearchAgents_no= 10 (número de agentes de búsqueda).
- Max_iteration= 30 (cantidad de ejecuciones).

3. Resultados obtenidos

En esta sección se describe de qué manera se probó el desempeño de GWO. Para todos los experimentos se utilizó la metaheurística GWO sobre MATLAB® v.7.8.0 (R2009a) bajo Sistema Operativo Windows 8.1 Enterprise 64 bits, en un computador: Intel Core i-3. 2.10 GHz., 4 Gb. RAM.

3.1. Experimento 1

El primer experimento planteado radica en calcular el total de longitudes acumuladas de cadenas aditivas para conjuntos de exponentes de pequeña longitud, a fin de comparar el desempeño de algoritmo GWO con otros resultados obtenidos por métodos deterministas de la literatura y por otras metaheurísticas del estado del arte.

Cuadro 2. Promedio de resultados obtenidos por GWO sobre los siguientes rangos de exponentes con 30 ejecuciones independientes.

Exponente	Longitud acumulada	Promedio	Mediana	Desviación
[1, 128]	960	7,5	4	2,159
[1, 256]	2459	9,6	9	2,986
[1, 512]	4994	9,8	10	1,9501

El Cuadro 2 resume los mejores resultados obtenidos por enumeración para los exponentes de los rangos expresados, tomando el mejor de un lote de 30 ejecuciones independientes, a fin de poder comparar tales resultados con otras propuestas para las que se han ejecutado experimentos empleando el mismo rango de exponentes.

El Cuadro 3 muestra una comparación de los resultados para GWO respecto de los métodos deterministas del estado del arte tomados de [13] cuyos resultados fueron utilizados para comparar con el enfoque AIS [13]. Por otro lado, el Cuadro 3 brinda una comparación de los mejores resultados de cadenas de

Cuadro 3. Cadenas de adición acumuladas para todas las longitudes de exponentes $e \in [1, 512]$

e	$[1, 512]$
Optimal [13]	4924
Binary [13]	5388
Çuaternary [13]	5226

Resultados GWO

Longitudes acumuladas	4994
Promedio	9,77
Mediana	10
Desvío	1,95

Cuadro 4. Comparación de los mejores resultados acumulados para GWO con otros enfoques, para el mismo rango de exponentes.

$e \in [1, 512]$	AIS	GA	PSO	EP	GWO
	4924	4924	—	4924	4994

adición acumuladas para el mismo rango de exponentes $[1, 512]$ obtenidos por métodos estocásticos y comparados en [17], salvo PSO que no reportó resultados para este rango de exponentes, respecto de la propuesta GWO. En el Cuadro 3 se puede observar además que GWO se aproxima al resultado del método “Optimal” [13], siendo éste quien mejor desempeño logró para dicho rango de exponentes según [13]. Finalmente, en el Cuadro 4 se observa que GWO llega en su mejor ejecución igualar los mejores resultados que arrojaron AIS, GA, PSO, EP para el rango $[1, 512]$. En comparación con los resultados de los métodos deterministas se observa que en su mejor ejecución se aproxima al mejor resultado arrojado por “Optimal”, mientras que en los resultados comparativos con los demás métodos estocásticos se acerca al óptimo de las otras propuestas, no superando las mismas por un porcentaje de $\sim 0,9\%$.

3.2. Experimento 2

El experimento plantea la comparación del desempeño de GWO respecto de una propuesta reciente [22] del estado del arte.

El Cuadro 5 muestra una comparación de los resultados de GWO respecto de GA Annealing [22] para una serie de exponentes diversos y especialmente difíciles de optimizar. Se considera dentro de esta categoría a aquellos exponentes para los cuales no es posible hallar cadenas de longitud mínima mediante los métodos deterministas. Se puede observar que en los 5 casos, GWO ha obtenido cadenas de longitud igual o menor al enfoque propuesto en [22]. Si bien los métodos aplicados en [13], [14] y [17] también han sido probados con esta clase de exponentes diversos, no son los mismos que los empleados en [22]. Por esta

Cuadro 5. Comparación de GA Annealing [22] y GWO para el mismo conjunto de exponentes.

Exponente	Cadena	GA Annealing	GWO
e		Longitud	
23	1, 2, 4, 5, 10, 20, 21, 23	7	7
55	1, 2, 3, 6, 12, 24, 27, 54, 55	9	8
130	1, 2, 4, 8, 16, 32, 64, 128, 129, 130	11	9
250	1, 2, 3, 5, 10, 20, 30, 50, 100, 150, 250	13	10
768	1, 2, 3, 6, 12, 24, 48, 96, 192, 384, 768	23	10

razón, se efectúa la comparación en el Cuadro 5 y en el próximo experimento se hace respecto a los otros métodos del estado del arte.

3.3. Experimento 3

El experimento plantea la comparación del desempeño de GWO respecto de otros abordajes del estado del arte.

Cuadro 6. Mejores resultados obtenidos por AIS [13], PSO [14], EP [17] y GWO para un conjunto de exponentes diversos.

Exponente	Cadena	Longitud			
		AIS [13]	PSO [14]	EP [17]	GWO
5	1, 2, 3, 5	3	3	3	3
7	1, 2, 4, 5, 7	4	4	4	4
11	1, 2, 4, 8, 10, 11	5	5	5	5
19	1, 2, 4, 8, 16, 18, 19	6	6	6	6
29	1, 2, 3, 6, 7, 14, 28, 29	7	7	7	7
47	1, 2, 4, 8, 9, 18, 36, 45, 47	8	8	8	8
71	1, 2, 3, 6, 7, 14, 21, 35, 70, 71	9	9	9	9
127	1, 2, 3, 6, 12, 18, 30, 60, 63, 126, 127	10	10	10	10
191	1, 2, 3, 6, 12, 18, 19, 38, 57, 95, 190, 191	11	11	11	11
379	1, 2, 3, 6, 9, 18, 27, 45, 72, 117, 189, 378, 379	12	12	12	12
607	1, 2, 3, 6, 9, 15, 30, 60, 61, 121, 182, 303, 606, 607	13	13	13	13
1087	1, 2, 3, 6, 9, 18, 36, 54, 108, 216, 324, 540, 541, 1082, 1085, 1087	14	14	14	15

El Cuadro 6 resume los resultados del tercer experimento, correspondiente a los mejores resultados obtenidos por GWO para un conjunto de exponentes diversos y difíciles de optimizar al igual que los resultados del experimento mostrados en el Cuadro 5 pero comparando dichos resultados con los obtenidos por

AIS [13], PSO [14] y EP [17] para los mismos casos. GWO ha sido comparado tomando algunos exponentes de las metaheurísticas mencionadas en el párrafo anterior a excepción de [8] pues tal como se menciona en [17], todas las demás ([13], [14]) superan sus resultados. Si bien GWO no supera los resultados obtenidos para los exponentes listados mediante el uso de los otros enfoques (ver Cuadro 6) tampoco ha arrojado resultados de menor calidad en términos de longitud de cadena, salvo para el caso del exponente $e = 1083$ empleado en este lote de casos de prueba. Al margen de coincidir la mayoría de los casos tomados para el experimento con los resultados arrojados por otras metaheurísticas, dichos resultados corresponden a cadenas de longitud mínima en función del cuadro de exponentes “diversos” presentado en [13]:

Cuadro 7. Mejores resultados obtenidos por AIS [13], PSO [14], EP [17] y GWO para un conjunto de exponentes diversos.

Longitud de cadena	Exponentes (e)
1	2
2	3, 4
3	5, 6, 8
4	7, 9, 10, 12, 16
5	11, 13, 14, 15, 17, 18, 20, 24, 32
6	19, 21, 22, 23, 25, 26, 27, 28, 30, 33, 34, 36, 40, 48, 64
7	29, 31, 35, 37, 38, 39, 41, 42, 43, 44, 45, 46, 49, 50, 51, 52, 54, 56, 60, 65, 66, 68, 72, 80, 96, 128
8	47, 53, 55, 57, 58, 59, 61, 62, 63, 67, 69, 70, 73, 74, 75, 76, 77, 78, 81, 82, 83, 84, 85, 86, 88, 90, 92, 97, 98, 99, 100, 102, 104, 108, 112, 120, 129, 130, 132, 136, 144, 160, 192, 256
9	71, 79, 87, 89, 91, 93, 94, 95, 101, 103, 105, 106, 107, 109, 110, 111, 113, 114, 115, 116, 117, 118, 119, 121, 122, 123, 124, 125, 126, 131, 133, 134, 135, 137, 138, 140, 145, 146, 147, 148, 149, 150, 152, 153, 154, 156, 161, 162, 163, 164, 165, 166, 168, 170, 172, 176, 180, 184, 193, 194, 195, 196, 198, 200, 204, 208, 216, 224, 240, 257, 258, 260, 264, 272, 288, 320, 384, 512

El Cuadro 7 resume los casos de exponentes más conocidos y probados de la literatura para los que se conoce cadenas de longitud óptima, indicando a qué longitud corresponden dichos valores óptimos (primera columna del cuadro). En función de esto, se verifica que por ejemplo el exponente $e = 71$ que se utilizó para el experimento del Cuadro 6, para el cual se obtuvo la cadena 1, 2, 3, 6, 7, 14, 21, 35, 70, 71 cuya longitud es 9, ésta coincide con la longitud que indica el Cuadro 7, además de existir coincidencia en la longitud también obtenida por [13], [14] y [17] para el mismo caso.

4. Conclusión

En este trabajo se propone el uso de la heurística GWO para encontrar cadenas de adición de longitud mínima, asociadas a distintos exponentes. GWO demostró resultados competitivos para abordar la problemática, que se evidencia a partir de la comparación de sus resultados, respecto de otras propuestas del estado arte. En el Experimento 1 se observa que para el rango de exponentes [1, 512], los resultados de cadenas de adición mínimas acumuladas superan ampliamente a dos de los métodos deterministas reportados en [13] para dicho rango y sólo es superado escasamente por “Optimal” [13]. Respecto de las propuestas de abordaje con métodos estocásticos, en el segundo experimento ha sido comparado de forma directa con los mismos casos de exponentes reportados por [22], logrando igualar y superar dicha propuesta, en la mayoría de los casos. Por último, se ha comparado la propuesta para un grupo de exponentes diversos y particularmente difíciles de optimizar, con relación al desempeño de otras metaheurísticas de la literatura: AIS [13], PSO [14] y EP [17], logrando igualar a aquellos los resultados reportados a excepción del exponente de mayor tamaño.

Como trabajo futuro, se analizarán los parámetros requeridos por GWO [21] o incluso hibridaciones u otras técnicas requeridas, para ampliar los resultados de la metaheurística a exponentes de mayor longitud.

Referencias

1. Rivest, R.L.; Shamir, A.; Adleman, L. 1978. *A method for obtaining digital signatures and public-key cryptosystems*. Cambridge.
2. Bos, J.; Coster, M. 1990. *Addition Chain Heuristics*. Centrum voor Wiskunde en Informatica. Amsterdam, Netherland.
3. Kaya Koc, C. 1994. *High-speed RSA implementation*. Technical report, RSA. Laboratories, Redwood City, CA. USA.
4. Rotger, L.H.; Coma, J.R.; Tena-Ayuso, J.G. *Criptografía con Curvas Elípticas*. Universitat Oberta de Catalunya. España
5. Liu, Y.; Passino, K.M. 2002. *Biomimicry of Social Foraging Bacteria for Distributed Optimization: Models, Principles, and Emergent Behaviors*. Journal of Optimization Theory and Applications: Vol. 115, No. 3, pp. 603–628. Ohio State University, Columbus, Ohio.
6. Nicosia, G.; Cutello, V.; Bentley, P.; Timmis, J. 2004. *Artificial Immune Systems*. Third International Conference, ICARIS 2004. Springer-Verlag Berlin Heidelberg, Germany.
7. Nedjah, N.; De Macedo-Mourelle, L. 2004. *Finding Minimal Addition Chains Using Ant Colony*. Department of Systems Engineering and Computation Faculty of Engineering, State University of Rio de Janeiro. Río de Janeiro, Brasil.
8. Cruz-Cortés, N; Rodríguez-Henríquez, F.; Juárez-Morales, R.; Coello Coello, C. 2005. *Finding Optimal Addition Chains Using a Genetic Algorithm Approach*. México.
9. Karaboga, D. 2005. *An Idea Based on Honey Bee Swarm for Numerical Optimization*. (Technical Report). Erciyes University, Engineering Faculty Computer Engineering Department. Kayseri, Turquía.

10. Karaboga, D.; Basturk, B. 2007. *A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm*. Department of Computer Engineering, Erciyes University, Kayseri, Turkey.
11. Washington, L. 2008. *Elliptic Curves: Number Theory and Cryptography*. Second Edition. Discrete Mathematics and Its applications, series editor: Kenneth H. Rosen. Taylor & Francis Group. Boca Ratón, USA.
12. Mezura-Montes, E.; Hernández- Ocaña, B. 2008. *Bacterial Foraging for Engineering Design Problems: Preliminary Results*. Laboratorio Nacional de Informática Avanzada (LANIA A.C.) - Universidad Juárez Autónoma de Tabasco. México.
13. Cruz-Cortés, N.; Rodríguez-Henríquez, F.; Coello Coello, C. 2008. *An Artificial Immune System Heuristic for Generating Short Addition Chains*. México.
14. León. A.; Cruz-Cortés, N.; Moreno-Armendáriz, M.; Orantes-Jiménez, S. 2009. *Finding Minimal Addition Chains with a Particle Swarm Optimization Algorithm*. Center for Computing Research, National Polytechnic Institute. México.
15. Yang, X.S. 2010. *Nature – Inspired Metaheuristic Algorithms*, Second Edition. University of Cambridge. Luniver Press, United Kingdom.
16. Gómez Bello, M. 2011. *La aritmética modular y alguna de sus aplicaciones*. Universidad Nacional de Colombia. Bogotá, Colombia.
17. Domínguez, I. 2011. *Optimización de Cadenas de Adición en Criptografía utilizando Programación Evolutiva*. Tesis de Maestría. Laboratorio Nacional de Informática Avanzada Centro de Enseñanza LANIA, Xalapa, Veracruz, México.
18. Binitha, S.; S Siva Sathya. 2012. *A Survey of Bio inspired Optimization Algorithms*. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2. India.
19. Baro, M. 2013. *Swarming: La Comunicación en múltiples direcciones y múltiples etapas*. Razón y palabra. Primera Revista Electrónica en Iberoamérica Especializada en Comunicación. Centro Avanzado de Comunicación - 25 Aniversario Eulalio Ferrer. Número 83, Junio – Agosto
20. Tall, A.; Sanghare, A.Y. 2013. *Efficient computation of addition-subtraction chains using generalized continued Fractions*. African Institute for Mathematical Sciences. Senegal.
21. Mirjalili,S.; Mirjalili, S.M.; Lewis, A. 2014. *Grey Wolf Optimizer*. School of Information and Communication Technology, Griffith University, Nathan Campus, Brisbane QLD 4111, Australia. Department of Electrical Engineering, Faculty of Electrical and Computer Engineering, ShahidBeheshti University, G.C. 1983963113, Tehran, Iran.
22. Pogačić, M. 2014. *Evolving Minimal Addition Chain Exponentiation*. University of Zagreb - Faculty of Electrical Engineering and Computing. Zagreb, Croacia.
23. Picek, S.; Coello Coello, A.; Jakobovic, D.; Metens, N. 2016. *Evolutionary Algorithms for Finding Short Addition Chains: Going the Distance*. Volume 9595 of the series Lecture Notes in Computer Science pp 121-137.
24. Mech., D. 1999. *Alpha Status, Dominance, and Division of Labor in Wolf Packs*. Canadian Journal of Zoology. Jamestown, Dakota del Norte.

Una propuesta para mejorar la comunicación mediante persuasión

Guillermo Aguirre, Marcelo Errecalde¹, Guillermo Simari²

¹ Lab. de Investigación y Desarrollo en Inteligencia Computacional (LIDIC),
{gaguirre,merreca}@uns1.edu.ar

Departamento de Informática. Universidad Nacional de San Luis
Ejército de los Andes 950 - Local 106. (D5700HHW) - San Luis - Argentina

² Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (LIDIA),
grs@cs.uns.edu.ar

Depto. de Ciencias e Ingeniería de la Computación. Univ. Nacional del Sur,
Av. Alem 1253, (B8000CPB) Bahía Blanca, Argentina

Resumen La comunicación efectiva requiere resolver posibles conflictos de manera conjunta. Durante la comunicación puede suceder que el emisor proponga algo sobre lo cual el receptor no coincide, pero mediante las reglas establecidas en los juegos dialógicos es posible llegar a un acuerdo para superar la situación. Usando un contexto formal, es posible que dos agentes se comuniquen procurando llegar a resultados mutuamente aceptables. El sistema garantiza que habrá un ganador del conflicto. Se muestra una situación en la que el resultado de la comunicación depende de la destrezas para persuadir al interlocutor mediante las frases adecuadas.

Palabras claves Comunicación, sistemas de diálogo, argumentación.

1. Introducción

Austin es ampliamente reconocido como el creador de la *Teoría de los Actos del Habla* [1]. Él examinó el uso de los *enunciados ejecutivos* en el lenguaje e investigó las diferencias entre sentencias *declarativas* (aquellas que se pueden constatar) y las sentencias *realizables*: “Las sentencias declarativas tienen la propiedad de ser verdaderas o falsas. Las sentencias realizables, por su parte, carecen de esas dos propiedades, tienen su propia tarea especial: se usan para realizar una acción. Emitir esta clase de sentencia es realizar la acción - una acción que difícilmente se podría hacer de otra manera.” Un ejemplo de sentencia realizable es: “Llamo a este barco el *Reina Elizabeth*” (sentenciado mientras rompe un champán contra el casco). “Es claro que pronunciar esa sentencia (en circunstancias apropiadas), no es describir lo que se está haciendo... es hacerlo. Un enunciado ejecutivo indica que la declaración de la frase, es la realización de una acción - no es considerado simplemente como decir algo”.

La realización exitosa de un enunciado ejecutivo (en inglés *performative*) requiere satisfacer ciertas condiciones relativas a las circunstancias y personas

involucradas, las que deben participar del procedimiento completo y de una manera sincera (sin actuar). Estas condiciones fueron refinadas por John Rogers Searle a fin de establecer el contexto más propicio para lograr los resultados esperados. Searle profundizó el trabajo previo aportando las denominadas *condiciones de los actos del habla*, que consistió en un conjunto de reglas extraídas de las condiciones necesarias y suficientes para que un acto del habla fuese exitoso y no presentara defectos al incorporarse como parte de una frase.

La teoría de los actos del habla ha sido ampliamente estudiada en el ámbito de los sistemas multi-agente [2,10], pero no es adecuada para resolver situaciones particulares que pueden darse durante la comunicación: cuando el emisor comunica algo con lo cual el receptor puede no coincidir, se hace necesario recurrir a algún mecanismo para superar ese tipo de obstáculo. Estos conflictos se pueden interpretar como algún tipo de engaño o falacia en algunos contextos [4].

Walton en su tesis [9] plantea que la mejor forma de estudiar las falacias es mediante un juego (o sistema) dialéctico que establezca modelos de las discusiones o los diálogos, los cuales constituyen el contexto y el ambiente natural en el cual las falacias han sido tradicionalmente concebidas. Los sistemas dialécticos se pueden seguir de manera descriptiva o formal. El estudio descriptivo considera las reglas y convenciones de discusiones reales como los debates parlamentarios o los interrogatorios. La aproximación formal considera la construcción de sistemas simples, pero precisos, donde las intervenciones están controladas por reglas claramente establecidas. Estos sistemas tendrán propiedades formales que presumiblemente puedan ser comparadas con secuencias interesantes de discusiones reales y de esa manera brindar alguna luz sobre estas últimas mediante la creación de modelos de ellas.

En la teoría clásica de los actos de habla, el principio de sinceridad es considerado obligatorio. La comunicación de agentes sigue esa teoría e incorporó el requisito de sinceridad. Pero en la realidad se dificulta garantizar que no haya engaño en la interacción de los agentes o se planteen cuestionamientos de diversa índole respecto a la información intercambiada. Este trabajo muestra la conveniencia de combinar comunicación de agentes con la capacidad de razonar sobre el contenido de la misma, mediante un sistema formal.

El trabajo se organiza así, la sección 2 plantea el uso de actos del habla en planes, donde se observa la influencia del trabajo de Austin en la comunicación de agentes. La sección 3 presenta un entorno para formalizar los sistemas de diálogo, primero de manera general, luego especificando particularidades para los diálogos de persuasión y finalmente instanciando el marco general en una situación de comunicación específica. En la sección 4 se presentan algunas conclusiones y futuras líneas de investigación.

2. Uso de los actos del habla en planes

En [3] Cohen & Perrault proponen usar los actos del habla como operadores dentro de planes. Cada operador tiene establecidas las condiciones requeridas y los deseos del agente antes de aplicar el operador y los efectos producidos una

vez aplicado. Para mostrar cómo la teoría de Austin incide sobre un componente clave de la comunicación de agentes, en esta sección se analiza el operador INFORMAR. Mediante este operador, un orador hace la declaración de una proposición al oyente con el objetivo de conseguir que el oyente crea que el orador cree que esa proposición es verdadera. Este acto se planifica deseando que finalmente el oyente también crea la proposición, pero eso no se consigue con esta primer especificación:

INFORMAR(ORADOR, OYENTE, PROPOSICIÓN)
 CAPAZDE.PR:
 ORADOR CREE PROPOSICIÓN
 DESEO.PR:
 ORADOR CREE ORADOR DESEA instancia-de-informar
 EFECTO:
 OYENTE CREE ORADOR CREE PROPOSICIÓN

La precondition CAPAZDE establece que el orador debe creer la proposición. El efecto es comunicar lo que el orador cree. El oyente puede elegir rechazar la proposición sin que esto invalide la acción del operador INFORMAR que realizó el orador. Para que un agente convenza a otro de que una proposición es verdadera se define el acto intermedio CONVENCER. Este paso sirve para que AGT1 convenza a AGT sobre la verdad de la proposición y sólo es necesario que AGT crea que AGT1 cree la proposición. Con la siguiente especificación:

CONVENCER(AGT1, AGT, PROP)
 CAPAZDE.PR:
 AGT CREE AGT1 CREE PROP
 EFECTO:
 AGT CREE PROP

Que el orador crea la proposición es una condición necesaria pero claramente no es suficiente para convencer a alguien para que crea algo. Un requisito más sofisticado de CONVENCER podría ser conocer las justificaciones que tiene AGT1 para creer la proposición, lo que lleva a que AGT tenga como creencia esas justificaciones y así siguiendo hasta establecer una cadena de razonamiento que conduce a la creencia mutua.

La planificación del acto INFORMAR es simple: para lograr “H CREE PROP” hace falta el plan de la Figura 1.

Hay que destacar que existen algunas diferencias entre la propuesta original de la teoría de los actos del habla y el enfoque que propone el uso de los actos del habla en planes. La *condición de no-obviedad* establecida por Searle para el caso del INFORMAR sería que el oyente no conozca de antemano el contenido del mensaje, este requisito no se considera una precondition para C&P. La condición de sinceridad que requiere que el orador efectivamente crea la proposición, se transforma también en un requisito para aplicar INFORMAR. En la vida real, la falta de sinceridad (no cumplir con la condición de sinceridad) es reclamar falsamente estar en cierto estado psicológico. En el ámbito de los modelos con planes, por su parte, significa que una creencia o un deseo no han sido incorporados al modelo como precondiciones.

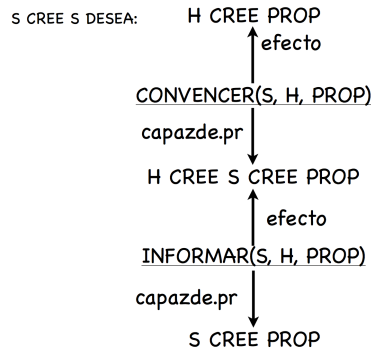


Figura 1. Plan para INFORMAR

En esta sección se analizó un plan del operador INFORMAR, donde se resalta la importancia del paso CONVENCER; en la siguiente sección se discuten los aspectos formales que debe cumplir ese paso.

3. Los sistemas de diálogo

Esta sección describe un marco formal para regular los diálogos de persuasión³, los cuales se podrían incorporar a los agentes para permitirles solucionar conflictos de opinión durante la comunicación.

Para comenzar se plantea una situación posible, conveniente para evaluar un intercambio de actos del habla y los conflictos que pueden surgir. El siguiente diálogo, entre dos soldados que deben operar en un campo minado (CM), exhibe algunas de las características típicas de persuasión y será usado para clarificar algunas ideas y para motivar la lectura.

Juan: El CM está desminado. (*hace un reclamo*)

José: ¿Por qué el CM está desminado?. (*consulta sobre fundamentos del reclamo*)

Juan: Porque trabajaron los dragaminas. (*brinda fundamentos del reclamo*)

José: Eso es verdad, (*acepta el reclamo*) pero eso no hace al CM desminado. (*establece un contrargumento*)

Juan: ¿Por qué no lo hace desminado?. (*consulta sobre fundamentos del reclamo*)

José: Porque no buscaron minas marinas en el río. (*establece un contrargumento dando sus fundamentos*)

Juan: Si, es cierto, no buscamos minas marinas; (*acepta el reclamo*) pero no hace falta porque el Alto Mando dice que aquí no existen minas marinas. (*recorta un contrargumento*)

José: De todas maneras creo que el CM no está desminado, porque el informante dijo que hay más minas. (*contrargumento alternativo*)

Juan: De acuerdo, estaba equivocado en que el CM está desminado.

³ Esta sección se basa en [7].

A continuación, el diálogo entre Juan y José será modelizado mediante un marco (en inglés *framework*) especialmente creado para resolver aquellas conversaciones iniciadas con un propósito específico [6]. La resolución, O , se consigue a partir de una secuencia de “movidas” (o jugadas) válidas, hechas en un contexto K particular. Estas secuencias son generadas por medio de un protocolo Pr , que en cada paso se aplica sobre el diálogo legal en curso y genera el conjunto de las posibles movidas a efectuar en ese punto. Las movidas son las jugadas que realizan los participantes del diálogo, quienes se agrupan en un conjunto \mathcal{A} . Para razonar, un participante a cuenta con una *base de conocimiento* Σ_a (la cual puede ser inconsistente) y los *compromisos*, C_a (las proposiciones que a hizo públicas). Conviene que la especificación de Pr incluya un par de funciones especiales: una que determina en cada paso cual es el jugador que debe mover, llamada *turno* y otra que establece cuando termina el diálogo, llamada *terminación*. El *lenguaje de comunicación*, \mathcal{L}_c , son los actos del habla que usan los participantes, el contenido de los cuales se especifica mediante el *lenguaje del tema*, \mathcal{L}_t . \mathcal{L}_t tiene una lógica L que puede o no ser monotónica y puede o no estar basada en argumentación. El conjunto de diálogos, $M^{\leq\infty}$, se crea desde \mathcal{L}_c , al igual que los diálogos finitos $M^{<\infty}$. Un *diálogo* d es una secuencia $m_1 \dots m_n \dots$ y la subsecuencia $m_1 \dots m_i$ se la identifica por d_i . El conjunto $D \subseteq M^{<\infty}$ y los elementos de D son llamados los *diálogos finitos legales*. Cada elemento del diálogo tiene un efecto sobre los compromisos, ese efecto está definido por la función C_a para el caso del jugador a .

Notar que no se asume ninguna relación entre los compromisos de los participantes y su base de conocimiento. Los compromisos son un punto de vista públicamente declarado del agente sobre una proposición, lo cual no necesita coincidir con las creencias internas del agente. El contexto K es el conocimiento acordado que se debe respetar. En el sistema legal representaría las leyes.

Lo que sigue se refiere a las particularidades del entorno para persuasión.

3.1. Sistema de persuasión

Según Walton [9] los diálogos de persuasión son diálogos donde el objetivo es resolver un conflicto entre los puntos de vista de al menos dos participantes. Un *punto de vista* en relación a una proposición puede ser positivo(a favor), negativo (en contra) o dudoso. Los participantes procuran persuadir a el(los) otro(s) participante(s) para que acepte(n) su punto de vista.

Un sistema de diálogo para persuasión requiere las siguientes instanciaciones.

El *propósito del diálogo* es resolver un conflicto de opinión a cerca de una o más proposiciones, llamadas *tópicos*, $\mathcal{T} \subseteq \mathcal{L}_t$. Este propósito de diálogo da origen a roles de participantes y reglas de resultado según se detalla a continuación.

Los participantes pueden tener los siguientes *roles*. Comenzando con $prop(t) \subseteq \mathcal{A}$, los *proponentes* del tópico $t \in \mathcal{T}$, es el conjunto (no vacío) de todos los participantes con un punto de vista positivo hacia t . De la misma manera $opon(t) \subseteq \mathcal{A}$, es el conjunto (no vacío) de todos los participantes con un punto de vista indeciso hacia el tópico t . Si fuese negativo, correspondería a una disputa. Para cualquier t , los conjuntos $prop(t)$ y $opon(t)$ son disjuntos y juntos no simepre agotan \mathcal{A} .

Las *reglas de resultado* para diálogos de persuasión definidas para un diálogo d , contexto K y tópico t son los *ganadores* y los *perdedores* de d con respecto al tópico t . Más precisamente, O consiste de dos funciones parciales w y l :

$$w : D \times Pot(\mathcal{L}_t) \times \mathcal{L}_t \rightarrow Pot(\mathcal{A})$$

$$l : D \times Pot(\mathcal{L}_t) \times \mathcal{L}_t \rightarrow Pot(\mathcal{A})$$

tal que ellas están definidas al menos para todos los diálogos terminados pero solamente para aquellos t que son un tópico de d . Estas funciones se escribirán como $w_t(d)$ y $l_t(d)$. Ellas además satisfacen las siguientes condiciones:

$$w_t(d) \cap l_t(d) = \emptyset$$

$$w_t(d) = \emptyset \text{ si, y sólo si, } l_t(d) = \emptyset$$

Si $|\mathcal{A}| = 2$, entonces $w_t(d)$ y $l_t(d)$ a lo más tienen un elemento.

Finalmente, en persuasión una de las partes del diálogo se va a dar por vencido.

$$w_t(d) \subseteq prop(t) \text{ o } w_t(d) \subseteq opon(t); \text{ y}$$

Si $a \in w_t(d)$ entonces

$$\text{Si } a \in prop(t) \text{ entonces } t \in C_a(d)$$

$$\text{Si } a \in opon(t) \text{ entonces } t \notin C_a(d)$$

Estas condiciones aseguran que un ganador no cambiará su punto de vista. Notar que ellas hacen que la persuasión entre dos personas sea un juego de suma cero.

El lenguaje de comunicación está formado por algunos actos del habla clásicos, los que generalmente vienen acompañados de una réplica típica para cada uno de ellos. La Tabla 1 presenta algunos ejemplos.

Locuciones	Réplicas
<i>reclamo</i> φ	<i>¿por qué</i> $\varphi?$, <i>reclamo</i> $\bar{\varphi}$, <i>concedo</i> φ
<i>¿por qué</i> $\varphi?$	φ <i>dado</i> S (alternativamente: <i>reclamo</i> S), <i>retiro</i> φ
<i>concedo</i> φ	
<i>retiro</i> φ	
φ <i>dado</i> S	<i>¿por qué</i> $\psi?$ ($\psi \in S$), <i>concedo</i> ψ ($\psi \in S$), φ' <i>dado</i> S'
<i>consulta</i> φ	<i>reclamo</i> φ , <i>reclamo</i> $\bar{\varphi}$, <i>retiro</i> φ

Tabla 1. Locuciones y réplicas típicas

Con este lenguaje de comunicación, el diálogo de Juan y José, sería:

P_1 : <i>reclamo</i> desminado	O_2 : <i>¿por qué</i> desminado?
P_3 : desminado <i>dado</i> barreminas	O_4 : <i>concedo</i> barreminas
P_6 : <i>¿por qué</i> \neg desminado?	O_5 : <i>reclamo</i> \neg desminado
P_8 : <i>concedo</i> minas marinas	O_7 : \neg desminado <i>dado</i> minas marinas
P_9 : <i>¿y qué?</i> <i>dado</i> Alto Mando	
P_{11} : <i>retiro</i> desminado	O_{10} : \neg desminado <i>dado</i> informante

Como se observa en el ejemplo, durante un diálogo de persuasión es posible que un mismo jugador realice sus movidas sin que intervenga el otro jugador, esto califica al sistema como de *múltiple respuesta*.

Como reglas de compromisos, las siguientes son generalmente aceptadas en la literatura. Si pl designa al orador, los cambios serían:

- Si $pl(m) = \text{reclamo } \varphi$ entonces $C_{pl}(d, m) = C_{pl}(d) \cup \varphi$
- Si $pl(m) = \text{¿por qué } \varphi?$ entonces $C_{pl}(d, m) = C_{pl}(d)$
- Si $pl(m) = \text{concedo } \varphi$ entonces $C_{pl}(d, m) = C_{pl}(d) \cup \varphi$
- Si $pl(m) = \text{retiro } \varphi$ entonces $C_{pl}(d, m) = C_{pl}(d) - \varphi$
- Si $pl(m) = \varphi \text{ dado } S$ entonces $C_{pl}(d, m) \supseteq C_{pl}(d) \cup \text{prem}(Arg)$

La regla para *dado* usa \supseteq porque tal movida puede comprometer a más que solamente las premisas del argumento movido. Por ejemplo en [5] la movida también compromete a φ , debido a que los argumentos también pueden ser movidos como contrargumentos en lugar de réplicas a desafíos de reclamos.

Continuación del ejemplo: De acuerdo a estas reglas, los conjuntos de compromisos de Juan y José al final del diálogo de ejemplo son:

- $C_P(d_{11}) \supseteq \{\text{dragaminas, minas marinas, } \neg\text{minas marinas aquí}\}$
- $C_O(d_{11}) \supseteq \{\neg\text{desminado, minas marinas, informante}\}$

3.2. Una instancia de sistema de persuasión

En esta sección se instancia el entorno de diálogos de persuasión para lograr resolver la situación planteada como ejemplo. En el marco de Prakken los participantes tienen los roles de proponente y oponente, y sus creencias son irrelevantes para los protocolos, razón por la cual, las creencias tienen una semántica pública. Los diálogos no tienen contexto. El sistema se abstrae del lenguaje de comunicación excepto que requiere la existencia de una estructura de respuesta explícita. También se abstrae del lenguaje lógico y la lógica, excepto que la lógica se asume que es basada en argumentos, que responde a semánticas fijas (en inglés *grounded*) y que los argumentos son árboles de inferencias rebatibles o *árbol de diálogos*.

El principal interés de este marco es enfocarse en los diálogos, mientras se deja suficiente libertad como para realizar réplicas alternativas y para posponer las réplicas. Esta libertad se consigue mediante dos características principales del sistema. Primero, \mathcal{L}_c tiene una estructura de respuesta explícita, donde cada movida o bien ataca o admite a la movida objetivo. Un \mathcal{L}_c de ejemplo con este formato se muestra en la tabla 2. Segundo, el ganador se define para cada uno de los diálogos, ya sea que esté terminado o no, y el ganador es definido en términos de la noción de *estado dialógico* de la movida. El estado dialógico de una movida se define recursivamente como sigue, haciendo uso de las características del árbol de diálogos generado por la estructura de respuesta del \mathcal{L}_c . Una movida está *adentro* si es admitida o si todas las réplicas que la atacan están *afuera*. (Esto implica que una movida sin réplicas está *adentro*). Y una movida está *afuera*

si tiene una réplica que esté *adentro*. Entonces un diálogo está siendo ganado (actualmente) por el proponente si su movida inicial está *adentro*, mientras que está (actualmente) siendo ganada por el oponente en otro caso.

Acto	Ataques	Admisiones
reclamo φ	¿por qué φ ?	concedo φ
φ dado S	¿por qué ψ ? ($\psi \in S$) φ' dado S' (φ' dado S' derrota φ dado S)	concedo ψ ($\psi \in S$) concedo φ
¿por qué φ ?	φ dado S	retiro φ
acepto φ		
rechazo φ		

Tabla 2. Un L_c ejemplo, en el sistema de Prakken

En combinación estas dos características del sistema sustentan la idea de *relevancia* que garantiza que el foco se mantenga en el diálogo, mientras deja cierto grado de libertad: una movida es relevante cuando deja a su objetivo *afuera* haciendo que el orador sea el actual ganador. La terminación es definida como la situación en la que un jugador debe mover pero no tiene movidas legales para hacer.

El lenguaje de comunicación propuesto es el mostrado en la Tabla 2 y está restringido por el requerimiento de que cada movida debe ser *relevante*. Esto hace que el protocolo sea de *respuesta inmediata*, lo cual implica que cada turno consista de cero o más admisiones seguidas por un ataque. Dentro de estos límites la posposición de respuesta está permitida, a veces incluso indefinidamente.

Seguidamente se discute un ejemplo, donde el \mathcal{L}_t se instancia con el lenguaje de la programación en lógica extendida priorizada basada en argumentos de Prakken y Sartor [8]. Esta lógica usa semántica fija (grounded) y soporta argumentos sobre las prioridades de las reglas rebatibles. (El ejemplo siguiente debiera hablar por sí mismo así que no se dará ninguna definición formal sobre la lógica. Las conectivas de las reglas tiene etiquetas con el nombre de la regla, lo que es necesario para expresar prioridades de las reglas.) Considere dos agentes con las siguientes bases de creencias:

$$\begin{aligned} \Sigma_P &= \{p, p \Rightarrow_{r_1} q, q \Rightarrow_{r_2} r, p \wedge s \Rightarrow_{r_3} r_2 > r_4\} \\ \Sigma_O &= \{t, t \Rightarrow_{r_4} \neg r\} \end{aligned}$$

Entonces las siguientes movidas son legales en el protocolo llamado de movida *relevante* (el objetivo de cada movida es indicado entre corchetes):

$$\begin{array}{l} P_1[-]: \text{reclamo } r \\ P_3[O_2]: r \text{ dado } q, q \Rightarrow r \\ P_5[O_4]: q \text{ dado } p, p \Rightarrow q \end{array} \quad \left\| \begin{array}{l} O_2[P_1]: \text{¿por qué } r? \\ O_4[P_3]: \text{¿por qué } q? \\ O_6[P_5]: \text{acepto } p \Rightarrow q \\ O_7[P_5]: \text{¿por qué } p? \end{array} \right.$$

Notar que los argumentos pueden ser contruidos de manera escalonada en varias movidas. Se observa que P tiene varias movidas permitidas, como ser

retractar cualquiera de las premisas de sus argumentos, o dar un argumento para p . Todas estas movidas son relevantes pero si P hace cualquier retractación entonces algún argumento para p deja de ser relevante, debido a lo cual P no puede ser el ganador actual. Además, si P retracta a r como una respuesta a P_1 , entonces el diálogo termina siendo ganado por O .

En cualquier punto después de P_3 , O podría haber movido su argumento contra r . Por ejemplo:

$$P_8[O_7]: r_2 > r_4 \text{ dado } p, s, p \wedge s \Rightarrow r_2 > r_4 \quad \parallel \quad O_7[P_3]: \neg r \text{ dado } t, t \Rightarrow \neg r$$

P_8 tiene un argumento prioritario el cual hace que P_3 derrote estrictamente a O_7 (notar que el hecho que s no se encuentre en la base de creencias de P no hace que la movida sea ilegal). En este punto, P_1 está *adentro*; O tiene varias movidas permitidas, como ser desafiar o conceder cualquier premisa de los argumentos de P , mover un contrargumento a P_5 o un segundo contrargumento a P_3 , y conceder el reclamo inicial de P .

Finalmente es posible modelizar mediante este protocolo el ejemplo considerado. La figura 2 muestra el árbol de diálogo, donde las movidas con cajas sólidas están *adentro* y las movidas con cajas de puntos están *afuera*. De acuer-

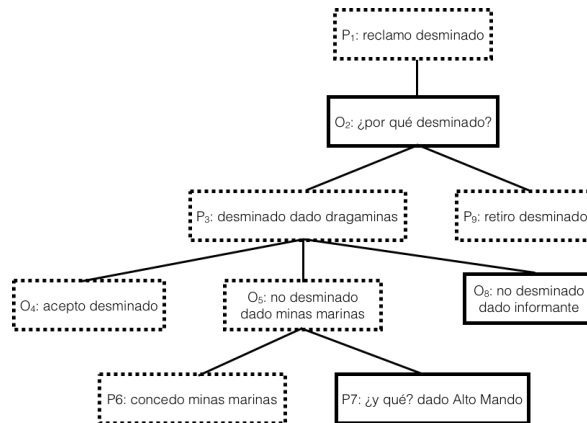


Figura 2. El ejemplo en un árbol de Prakken .

do a lo que se muestra en la Figura 2, un acto informar con el contenido “El CM está desminado” no sería realmente exitosos cuando el receptor tiene incorporado un sistema de diálogo de persuasión. Los agentes heterogéneos deben incorporar estos sistemas de diálogos para colaborar en alcanzar el objetivo del sistema multi-agente. Se pueden considerar variantes en los sistemas de diálogo formales que usen otros protocolos.

4. Consideraciones Finales

En este artículo se puede observar que la comunicación de agentes tiene algunas limitaciones, originadas en la dificultad de cumplir con el condición de sinceridad de la Teoría de los actos del habla. El caso del operador INFORMAR es analizado de manera interna dentro del agente, pero situaciones similares ocurren hacia afuera del agente, cuando se consideran dos o más agentes. Esta afirmación requiere mayor justificación, la cual es motivo actual de investigación. La propuesta central del trabajo, es superar esta limitación equipando a los agentes con un sistema de persuasión formal que use argumentación. Resulta de interés analizar la instanciación del entorno con distintos marcos argumentativos.

Referencias

1. Austin, J.L.: How to Do Things with Words. Harvard University Press, Cambridge, MA, Cambridge, MA (1962)
2. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. John Wiley & Sons, NJ (Apr 2007)
3. Cohen, P.R., Perrault, C.R.: Elements of a plan-based theory of speech acts. *Cognitive science* 3(3), 177–212 (1979)
4. Pan, Y., Cao, C., Sui, Y.: A formal system for lies based on speech acts in multi-agent systems. In: Foundations of Computational Intelligence, 2007. FOCI 2007. IEEE Symposium on. pp. 228–234. IEEE (2007)
5. Prakken, H.: Coherence and flexibility in dialogue games for argumentation. *Journal of logic and computation* 15(6), 1009–1040 (2005)
6. Prakken, H.: Formal systems for persuasion dialogue. *The knowledge engineering review* 21(02), 163–188 (2006)
7. Prakken, H.: Models of persuasion dialogue. In: Rahwan, I., Simari, G.R. (eds.) *Argumentation in artificial intelligence*, pp. 281–300. Springer (2009)
8. Prakken, H., Sartor, G.: Argument-based extended logic programming with defeasible priorities. *Journal of applied non-classical logics* 7(1-2), 25–75 (1997)
9. Walton, D.N.: *Logical Dialogue-Games and Fallacies*. University Press of America, Lanham, Maryland (1984)
10. Wooldridge, M.: *An introduction to multiagent systems*. John Wiley & Sons (2009)



XVI Workshop Procesamiento Distribuido y Paralelo (WPDP)

Comité Científico:

Ardenghi Jorge (Argentina)
Balladini, Javier (Argentina)
Bertogna, Leandro (Argentina)
Bianchini Germán (Argentina)
Cancela, Héctor (Uruguay)
CaymesScutari Paola (Argentina)
Chichizola Franco (Argentina)
De Giusti, Armando (Argentina)
De Giusti, Laura (Argentina)
Denham Mónica (Argentina)
Doallo, Ramón (España)
Fabero, Juan C. (España)
García Garino Carlos (Argentina)
Gil Costa Verónica (Argentina)
Laguna Carlos Alfonso (España)
Luque, Emilio (España)
Marín, Mauricio (Chile)
Martín, María J. (España)
Naiouf, Marcelo (Argentina)
Olcoz Herrero Katzalin (España)
Pardo Xoan (España)
Piccoli Fabiana (España)
Piñuel Luis (España)
Printista Marcela (Argentina)
Rexachs, Dolores (España)
Rodríguez León, Casiano (España)
Rodríguez de Souza, Josemar (Brasil)
Suppi, Remo (España)
Tinetti, Fernando (Argentina)
Tirado, Francisco (España)
Touríño, Juan (España)



XVI Workshop Procesamiento Distribuido y Paralelo (WPDP)

ID	Trabajo	Autores	
9169	Estructurando código paralelo para clusters heterogéneos de CPUs/GPUs	Adrián Pousa (UNLP), Victoria Sanz (UNLP), Armando De Giusti (UNLP)	139
9193	Método híbrido de reducción de incertidumbre aplicado a la predicción del comportamiento de incendios forestales	Miguel Mendez Garabetti (UNCUYO), Germán Bianchini (UTN-FRM), Paola Caymes-Scutari (UTN-FRM), Laura Tardivo (UNRC)	149
9205	Análisis de uso de un algoritmo de balanceo de carga estático en un Cluster Multi-GPU Heterogéneo	Erica Montes de Oca (UNLP), Laura De Giusti (UNLP), Franco Chichizola (UNLP), Armando E. De Giusti (UNLP), Marcelo Naiouf (UNLP)	159
9245	Un sistema de visión global paralelo para fútbol de robots con fines educativos	Rodrigo Cañibano (UNCOMA), Eduardo Grosclaude (UNCOMA), Javier Balladini (UNCOMA)	169
9280	Optimización y computación paralela aplicados a mejorar la predicción de un simulador de cauce de ríos	Adriana Angélica Gaudiani (UNGS), Emilio Luque (UAB), Pablo García (OTRA), Marcelo Naiouf (UNLP), Armando E. De Giusti (UNLP)	179
9304	Cloud Robotics: Sistema Multi-Robot conectado al Cloud público AWS	Ismael P. Rodriguez (UNLP), Juan Manuel Paniego (UNLP), Sebastián Rodriguez Eguren (UNLP), César Estrebou (UNLP), Armando E. De Giusti (UNLP)	189
9307	Análisis de la migración del cómputo intensivo a un cloud privado con OpenStack	María Antonia Murazzo (UNSJ), Federico Castiglione (UNSJ), Diego Medel (UNSJ), Fernando G. Tinetti (UNLP), Nelson R. Rodriguez (UNSJ)	199
9314	A Parallel Proposal for SEIR model using Cellular Automata	Facundo Casares (UNSL), Pablo Cristian Tissera (UNSL), María Fabiana Piccoli (UNSL)	208

Estructurando código paralelo para clusters heterogéneos de CPUs/GPUs

Adrian Pousa¹, Victoria Sanz^{1,2}, Armando De Giusti^{1,2}

¹III-LIDI, Facultad de Informática, UNLP, La Plata, Argentina

²CONICET, Ministerio de Ciencia, Tecnología e Innovación Productiva, Argentina
{apousa, vsanz, degiusti}@lidi.info.unlp.edu.ar

Resumen Los clusters de CPUs/GPUs se han vuelto habituales en HPC. Para aprovechar al máximo su potencia de cómputo, las aplicaciones deben desarrollarse combinando distintas herramientas de programación paralela, por esto el código se torna complejo y difícil de estructurar. En este trabajo describimos un esquema para estructurar código paralelo a ser ejecutado sobre un cluster de CPUs/GPUs y explotar toda su potencia de cómputo (CPUs y GPUs disponibles). En particular, nos centramos en aplicaciones desarrolladas con MPI+OpenMP+CUDA. Asimismo, explicamos los pasos a seguir para compilar estas aplicaciones híbridas. Resolvemos el problema de suma por reducción utilizando el esquema propuesto sobre un cluster de CPUs/GPUs heterogéneo, por esto la distribución de carga tiene en cuenta las capacidades de los recursos de cómputo disponibles. Comprobamos que es posible incrementar el rendimiento de la aplicación considerando todos los recursos de cómputo del cluster (CPUs y GPUs) respecto a utilizar sólo las GPUs.

Keywords: Programación híbrida, HPC, MPI, OpenMP, CUDA, Cluster, Multicore, GPU

1. Introducción

Hoy en día existen arquitecturas paralelas con gran poder de cómputo que combinan varios procesadores multi-core dentro de una misma máquina y múltiples GPUs. Para sacar provecho a esta arquitectura, las aplicaciones deben ser programadas a partir de combinar diferentes herramientas de programación, por ejemplo: MPI+CUDA, OpenMP+CUDA, Pthreads+CUDA [1,2,3,4]. En particular, en máquinas donde el número de cores supera el número de GPUs, la utilización de todos los recursos de cómputo disponibles (cores y GPUs) puede llevar a una mejora en el rendimiento con respecto a utilizar sólo un tipo de recurso.

Asimismo, varias de estas máquinas pueden conectarse a través de una red formando un cluster de CPUs/GPUs. En este caso, para explotar toda la potencia de cómputo provista por el cluster, la aplicación debe ser programada a partir de combinar herramientas de programación de memoria compartida y de paso de mensajes. Así surgen distintos patrones, como por ejemplo: MPI+OpenMP+CUDA y MPI+Pthreads+CUDA.

Aún más, las máquinas del cluster de CPUs/GPUs pueden diferir entre sí, es decir tanto en modelo y número de CPUs y GPUs, formando arquitecturas paralelas totalmente heterogéneas.

Estando en la era del "Big Data", una amplia gama de aplicaciones que procesan grandes volúmenes de datos para extraer información necesaria para la toma de decisiones en distintos ámbitos (salud, industria, finanzas, etc) pueden beneficiarse de las arquitecturas antes mencionadas para reducir su tiempo de cómputo.

Todo lo anterior supone un reto para el programador de aplicaciones, ya que no es trivial portar una aplicación heredada programada utilizando una única herramienta de programación paralela (MPI, Pthreads, OpenMP, CUDA) o combinaciones de dos de ellas, para que corra eficientemente sobre un cluster de CPUs/GPUs. Adicionalmente, la heterogeneidad del cluster de CPUs/GPUs añade un desafío adicional a la hora de distribuir la carga de trabajo.

En este trabajo describimos un esquema para estructurar código paralelo a ser ejecutado sobre un cluster de CPUs/GPUs, de modo de aprovechar toda la potencia de cómputo de esta arquitectura (CPUs y GPUs disponibles). En particular, nos centramos en aplicaciones desarrolladas con MPI+OpenMP+CUDA. Asimismo, explicamos los pasos a seguir para compilar estas aplicaciones híbridas. Resolvemos el problema de suma por reducción utilizando el esquema propuesto sobre un cluster de CPUs/GPUs heterogéneo, por esto la distribución de carga tiene en cuenta las capacidades de los recursos de cómputo disponibles. Comprobamos que es posible incrementar el rendimiento de la aplicación considerando todos los recursos de cómputo del cluster (CPUs y GPUs) respecto a utilizar sólo las GPUs.

El resto del paper está organizado de la siguiente manera. La Sección 2 presenta trabajos relacionados con la programación de GPUs, multi-GPUs y clusters de CPUs/GPUs. La Sección 3 introduce los modelos y herramientas de programación paralela. La Sección 4 introduce el esquema propuesto para estructurar código a ser ejecutado sobre un cluster de CPUs/GPUs. La Sección 5 muestra la evaluación experimental. Por último, la Sección 6 presenta las conclusiones y líneas de trabajo futuro.

2. Trabajos relacionados

En las últimas dos décadas, la evolución en las arquitecturas de cómputo trajo aparejada una revolución en la forma de programar las aplicaciones.

La integración de una GPU en una máquina ha impulsado el desarrollo de aplicaciones paralelas que explotan la potencia de ésta placa. En este caso, las herramientas de programación utilizadas son CUDA [4] u OpenCL [5], según el proveedor de la placa gráfica (NVIDIA/AMD). En particular, se pueden mencionar algoritmos de criptografía [6,7], análisis de imágenes médicas [8], simulación de dinámica de fluidos [9,10], etc, que alcanzan buena aceleración sobre una GPU.

Por otro lado, diversos autores han estudiado la paralelización de aplicaciones sobre clusters de CPUs/GPUs, debido a la atractiva relación costo/rendimiento de esta arquitectura. En general, el patrón utilizado al paralelizar se basa en lanzar un proceso MPI que controla a cada GPU disponible en el cluster. De este modo, se delega el control de la aplicación y la comunicación a MPI, mientras que se encarga a CUDA el cómputo intensivo de datos [9,11]. En particular, en [12] los autores paralelizan aplicaciones con MPI+CUDA y muestran cómo estructurar y compilar el código que combina estas dos herramientas. Para evaluar las aplicaciones, utilizan un cluster compuesto por dos máquinas heterogéneas (una de ellas integra 4 GPUs Tesla T10 y la otra 1 GPU Tesla C1060). La heterogeneidad subyacente en dicho cluster implica realizar la repartición de trabajo en base a la potencia de cómputo de las GPUs de cada máquina.

Otros autores han desarrollado aplicaciones que aprovechan la potencia de las múltiples GPUs integradas en una misma máquina. Esto da lugar a patrones híbridos (MPI+CUDA, OpenMP+CUDA, Pthreads+CUDA) que consisten en lanzar tantos procesos/threads como GPUs disponibles, encargando a cada uno la interacción con una GPU, la cual realiza el cómputo de datos [12,13,14].

Los modelos adoptados por los autores mencionados con anterioridad consisten en hacer trabajar a las GPUs disponibles, encargando la gestión de las GPUs a un número similar de cores de la arquitectura. En el caso particular que el número de cores disponibles en una máquina exceda al número de GPUs, estos modelos llevan a infrautilizar los recursos de cómputo, ya que ciertos cores de la máquina quedan ociosos.

En este sentido, en [15] los autores proponen un patrón híbrido MPI+OpenMP+CUDA (MOC) para aprovechar toda la potencia de cómputo de un cluster de CPUs/GPUs, y lo contrastan contra los patrones MPI puro y MPI+CUDA (MC). El patrón MOC se basa en lanzar un proceso MPI por cada GPU del cluster. Cada proceso a su vez genera una cantidad de threads adecuada (dada por $m \times n_c / n_g - 1$, donde m es el número de procesadores de la máquina, n_c es la cantidad de cores de cada procesador y n_g es la cantidad de GPUs de la máquina). El thread maestro del proceso se encargará de interactuar con la GPU, la cual realizará parte del cómputo sobre los datos, mientras que los demás threads computan el restante de datos. Para obtener buen rendimiento, la distribución de la carga de trabajo dentro de cada máquina la realizan teniendo en cuenta las capacidades de cómputo de la CPU y GPU. De los resultados experimentales obtenidos a partir de correr distintos tipos de aplicaciones sobre el cluster homogéneo TianHe-1A, en el cual cada nodo posee una única GPU, destacan que este patrón es efectivo especialmente para aplicaciones de cómputo intensivo.

Siguiendo una línea similar, el trabajo [16] introduce la paralelización del algoritmo de criptografía AES utilizando OpenMP+CUDA, sobre una máquina con múltiples cores y una única GPU. La estrategia propuesta por los autores consiste en lanzar varios hilos, uno de ellos se encarga de la interacción con la GPU y de invocar al kernel para computar AES sobre la GPU, mientras que el resto de los hilos computan AES sobre cores de la CPU. Para la repartición de trabajo tienen en cuenta la capacidad de la GPU y de la CPU. Comparan esta

estrategia híbrida (HPP y HPUP) contra aquellas que consisten en utilizar sólo la GPU (GPP y GPUP) y sólo los cores de la CPU (CPP y CPUP), programadas únicamente con CUDA y OpenMP respectivamente. La estrategia híbrida obtiene mejores resultados, a esta la sigue el algoritmo CUDA y por último el algoritmo OpenMP.

De lo anterior podemos destacar que, si bien en [15] proponen un patrón MPI+OpenMP+CUDA para aprovechar toda la potencia de un cluster de CPUs/GPUs, no indican la manera de estructurar el código híbrido ni tampoco muestran cómo realizar la compilación, la cual no es trivial. Además, los autores centran su experimentación sobre un cluster homogéneo, por lo tanto la distribución de carga es sencilla.

3. Modelos y Herramientas de Programación Paralela

Un modelo de programación paralela describe un sistema paralelo desde el punto de vista del programador. Una clasificación posible es según el mecanismo de interacción de los procesos:

- Modelo de memoria compartida: los procesos se comunican a través de escribir y leer la memoria compartida.
- Modelo de paso de mensajes: los procesos se comunican a través de enviar y recibir mensajes (ya que no tienen acceso a zonas de memoria compartida).

A continuación se resumen las características de las herramientas OpenMP, MPI y CUDA, hoy predominantes en el desarrollo de aplicaciones paralelas. Si bien existen otras herramientas de programación paralela, como por ejemplo Pthreads, Cilk, OpenCL, entre otras, no serán tratadas aquí, pero se discuten en detalle en [17].

3.1. OpenMP

OpenMP [2] es un estándar que define una API para la programación de aplicaciones sobre memoria compartida, disponible para los lenguajes C, C++ y Fortran. Provee un conjunto de directivas, rutinas y variables de entorno. A partir del conjunto de directivas se puede crear threads, realizar operaciones de sincronización y distribuir la carga de trabajo entre threads.

OpenMP impone una estructura jerárquica entre los threads, ya que sigue el modelo fork/join. En este modelo la ejecución del programa comienza como un único hilo (maestro), que ejecuta secuencialmente hasta encontrar una región paralela (directiva parallel). En ese momento, el maestro crea un conjunto de threads (fork). A partir de allí, cada thread ejecutará las sentencias encerradas en la región paralela en forma concurrente. Cuando todos los threads finalizaron la ejecución, se sincronizan y terminan (join), quedando sólo el thread maestro, el cual continúa la ejecución.

3.2. MPI

MPI (Message Passing Interface) [1] es un estándar que define una API para la comunicación por paso de mensajes. Existen diversas implementaciones (por ejemplo, OpenMPI y MPICH), que a su vez se adaptan a distintas arquitecturas, de este modo aprovechan las características físicas del sistema. Entre las funciones básicas que provee MPI se encuentran aquellas que posibilitan realizar comunicación punto a punto y comunicación colectiva.

3.3. CUDA

CUDA (Compute Unified Device Architecture) [4] es una extensión al lenguaje C que permite a los programadores escribir código a ser ejecutado en una GPU NVIDIA de forma sencilla.

Desde el punto de vista del programador CUDA, el sistema paralelo está formado por un host (CPU) y un device (GPU). Así, el programa CUDA tendrá fases a ejecutar en el host y fases a ejecutar en el device (codificadas a través de funciones llamadas kernels). En el device, el cómputo correspondiente a un kernel es realizado por un conjunto de hilos, que se organizan en bloques y que a su vez se agrupan en un grid. De esta manera, cuando el programador invoca a un kernel debe especificar la organización de los hilos (esto es, la cantidad de hilos por bloque y su organización, y la cantidad de bloques del grid y su organización). La ejecución del grid será tratada por una única GPU. En particular, cada Streaming Multiprocessor de la GPU será encargado de ejecutar uno o varios bloques de éste grid. Únicamente los hilos que conforman un bloque pueden cooperar, a través de la memoria compartida, y sincronizarse.

El modelo de memoria CUDA se organiza en una jerarquía y está compuesto por: la memoria global, que admite lectura/escritura por parte del host y del device; la memoria de constantes, que admite lectura/escritura por parte del host y sólo lectura por parte del device; la memoria compartida, que puede ser accedida por todos los hilos de un bloque; y por último los registros accesibles sólo por el hilo.

4. Esquema híbrido MPI+OpenMP+CUDA

En la actualidad, es posible encontrar arquitecturas paralelas integradas por dispositivos con características diferentes, por ejemplo, clusters de CPUs/GPUs compuestos por máquinas con múltiples cores y múltiples GPUs. A su vez, las CPUs y GPUs de las distintas máquinas pueden tener diferentes prestaciones. Con el objetivo de aprovechar al máximo la potencia de estas arquitecturas heterogéneas, las aplicaciones deben ser desarrolladas combinando distintas herramientas de programación paralela. El código de una aplicación con estas características en general es complejo y difícil de estructurar.

En esta sección describimos un esquema para estructurar código paralelo a ser ejecutado sobre un cluster de CPUs/GPUs, de modo de aprovechar todos los

cores y GPUs. En particular, nos centramos en aplicaciones desarrolladas con MPI+OpenMP+CUDA, en las cuales se pueden diferenciar tres tipos de código: el código MPI, donde se realiza la comunicación entre los procesos generados; el código OpenMP, encargado de crear los hilos, algunos de ellos realizarán el cómputo sobre los cores y otros gestionarán las GPUs; el código CUDA, encargado de crear los hilos que realizarán el cómputo sobre la GPU.

Para desarrollar una aplicación con estas características, la forma más sencilla es incluir todo el código dentro de un mismo archivo. Sin embargo, utilizaremos una implementación modular ya que facilita la lectura y el mantenimiento del código.

La estructura que proponemos implementa el código de cada herramienta en archivos separados, como explicamos a continuación.

El archivo principal *mpi_source.c* incluye únicamente código MPI. Al iniciar la ejecución de la aplicación, MPI genera tantos procesos como el usuario indique. En nuestro caso, generamos un proceso por máquina del cluster. Cada proceso ejecutará el código que se encuentra en dicho archivo, cuyo objetivo es el control de la aplicación, la comunicación entre máquinas y la distribución de datos. A su vez, cada proceso interactúa con el módulo que implementa el código OpenMP *omp_source.c*.

<pre> <i>mpi_source.c</i> #include <mpi.h> #include "omp_source.h" static void mpi_function(int myRank){ ... omp_function(); ... } int main(int argc, char *argv[]){ int myRank; MPI_Init(&argc, &argv); MPI_Comm_rank(MPLCOMM_WORLD,&myRank); mpi_function(myRank); MPI_Finalize(); return(0); } </pre>	<pre> <i>omp_source.h</i> #ifndef _FUNCION_OMP_ #define _FUNCION_OMP_ extern void omp_function(); #endif <i>omp_source.c</i> #include<omp.h> #include "cuda_source.h" void core_function(){ ... } void omp_function(void){ ... int tid = omp_get_thread_num(); #pragma omp parallel { if(tid < cuda_deviceCount()) cuda_function(tid); else core_function(); } ... } </pre>
---	---

En el módulo OpenMP se crean tantos hilos como cores posea la máquina. Un número determinado de estos hilos se destina a gestionar las GPUs disponibles. En particular, se asocia un hilo por GPU y, por simplicidad, estos hilos son aquellos con menor identificador. De esta forma, el identificador del hilo coincide con el identificador de la GPU con la que se relaciona. El resto de los hilos se destinan al cómputo sobre los cores de la máquina. Los hilos que ges-

tionan las GPUs interactúan con el módulo que implementa el código CUDA *cuda_source.cu*.

El módulo CUDA implementa todo lo relacionado a la ejecución sobre una GPU: selecciona la GPU; aloca espacio en memoria de la GPU; realiza las copias host-to-device; determina la cantidad de hilos a utilizar y su organización (bloques y grid); invoca el kernel CUDA; por último, recupera los resultados (copia device-to-host) y libera la memoria alocada en la GPU.

<i>cuda_source.h</i>	<i>cuda_source.cu</i>
<pre>#ifndef _FUNCION_CUDA_ #define _FUNCION_CUDA_ extern void cuda_function(int gpuID); extern int cuda_deviceCount(void); #endif</pre>	<pre>#include<cuda.h> #include<cuda_runtime.h> __global__ void kernel_function(dtype* args){ ... } extern "C" void cuda_function(int gpuID){ ... cudaSetDevice(gpuID); cudaMalloc(...); cudaMemcpy(..., cudaMemcpyHostToDevice); dim3 dimGrid(block_size); dim3 dimBlock(grid_size); kernel_function<<<<dimGrid, dimBlock>>>(args); cudaMemcpy(..., cudaMemcpyDeviceToHost); cudaFree(...); ... } extern "C" int cuda_deviceCount(void){ int nDevices; cudaGetDeviceCount(&nDevices); return nDevices; }</pre>

Este esquema puede ser aplicado en la resolución de problemas regulares, en los cuales los datos a procesar se encuentran disponibles desde el inicio de la ejecución de la aplicación.

4.1. Compilación

Generalmente, el código implementado usando una herramienta particular se compila con un compilador específico. Por ejemplo: el código MPI se compila con *mpicc*; el código OpenMP se compila con *gcc*; y el código CUDA se compila con *nvcc*. Afortunadamente, *mpicc* y *nvcc* son "wrappers" de *gcc* y es posible utilizar alguno de ellos para compilar la aplicación híbrida.

Básicamente, existen dos formas de compilar una aplicación híbrida. La primera consiste en utilizar un sólo compilador. En nuestro ejemplo, la forma más simple de hacerlo es a través del compilador de CUDA (*nvcc*), como se indica a continuación.

```
$ nvcc -Xcompiler -fopenmp -I$MPI_PATH/include
-L$MPI_PATH/lib -lmpi -o mpi_omp_cuda mpi_source.c
omp_source.c cuda_source.cu
```

La segunda forma consiste en compilar cada módulo por separado a partir del compilador correspondiente, generando código objeto, y luego enlazar el código generado utilizando alguno de los compiladores mencionados. En nuestro ejemplo, el enlazado lo realizamos con el compilador de MPI (*mpicc*).

```
$ mpicc -c mpi_source.c -o mpi_source.o
$ gcc -fopenmp -c omp_source.c -o omp_source.o
$ nvcc -c cuda_source.cu -o cuda_source.o
$ mpicc mpi_source.o omp_source.o cuda_source.o
-L$CUDA_PATH/lib64 -lcudart -lgomp -o mpi_omp_cuda
```

5. Experimentos

Utilizamos el esquema propuesto en la Sección 4 para resolver el problema de suma por reducción sobre un cluster de CPUs/GPUs. Específicamente, el algoritmo realiza la suma de los elementos de un vector. Evaluamos esta implementación, a la que llamaremos SR_MOC, sobre un cluster heterogéneo compuesto por 4 máquinas, cuyas características se muestran en la Tabla 1.

	Modelo de CPU	#CPUs	Cores	Modelo de GPU	#GPUs	GPU cores
Máquina 1	Intel i5 2310 2.9Ghz	1	4	Nvidia Tesla C2075 1150Mhz	2	448
Máquina 2	Intel i5 2310 2.9Ghz	1	4	Nvidia GTX 560Ti 1685Mhz	2	384
Máquina 3	Intel G2020 2.90GHz	1	2	Nvidia GTX 480 1401Mhz	2	480
Máquina 4	Intel i7 3770K 3.50GHz	1	4	Nvidia GTX 960 1127Mhz	1	1024

Tabla 1: Características del cluster de CPUs/GPUs heterogéneo

	Porcentaje por máquina	Porcentaje a los cores	Porcentaje a las GPUs
Máquina 1	21 %	11 %	89 %
Máquina 2	16 %	14 %	86 %
Máquina 3	24 %	0 %	100 %
Máquina 4	39 %	11 %	89 %

Tabla 2: Porcentaje de carga de trabajo asignada a cada máquina del cluster.

En un escenario heterogéneo es necesario distribuir la carga de trabajo teniendo en cuenta las capacidades de cada unidad de procesamiento. Varios autores estudiaron este tema. Por las características de nuestra aplicación, nos basamos en las propuestas de distribución estática de [18,19]. Para distribuir la carga, obtienen la potencia de cada procesador con respecto al mejor procesador de la arquitectura (*pcr* o potencia de cómputo relativa). En función de la *pcr* determinan el porcentaje de la carga de trabajo que debe computar cada unidad de procesamiento.

A partir de ejecutar la aplicación en cada unidad de procesamiento (core/GPU) del cluster, obtuvimos la potencia de cómputo relativa (*pcr*) de cada una

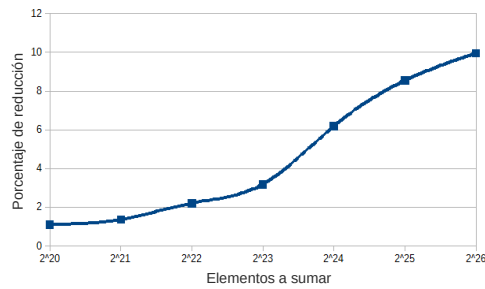


Figura 1: Reducción en el tiempo de ejecución SR_MOC vs SR_MC.

de las unidades. En función de las potencias calculadas, determinamos la *pcr* de cada máquina y, a partir de esto, el porcentaje de la carga de trabajo que recibirá cada una, el cual se muestra en la segunda columna de la Tabla 2.

Asimismo, la porción de la carga de trabajo recibida por una máquina debe distribuirse entre sus cores y GPUs de acuerdo a la potencia de cómputo relativa de cada uno de ellos dentro de esa máquina. De lo anterior, obtuvimos el porcentaje de la carga de trabajo que recibirá cada core/GPU de la máquina, los cuales se muestran en la tercera y cuarta columna de la Tabla 2.

Para ejemplificar esto último, podemos ver en la Tabla 1 que la Máquina 1 posee 4 cores y 2 GPUs: 2 cores estarán dedicados a interactuar con las GPUs y 2 cores estarán disponibles para realizar cómputo. De la porción de la carga de trabajo recibida por esta máquina, el 11 % será distribuido proporcionalmente entre los 2 cores que realizarán cómputo (5,5 % a cada uno) y el 89 % restante será procesado por las 2 GPUs (44,5 % a cada una). En el caso particular de la Máquina 3, los únicos 2 cores de la arquitectura estarán dedicados a gestionar las GPUs, por lo tanto las GPUs realizarán el 100 % del trabajo (50 % cada una).

Comparamos el algoritmo SR_MOC con un algoritmo MPI+CUDA que realiza la suma por reducción utilizando sólo las GPUs del cluster (sin tener en cuenta los cores), a la que llamaremos SR_MC. La Figura 1 muestra que el algoritmo SR_MOC reduce el tiempo de ejecución a medida que se incrementa el volumen de la carga de trabajo, con respecto al algoritmo SR_MC. Como se puede observar, el porcentaje de reducción es de hasta un 10 %.

6. Conclusiones y trabajo futuro

En este trabajo presentamos un esquema para estructurar código paralelo a ser ejecutado sobre un cluster de CPUs/GPUs, basado en MPI+OpenMP+CUDA. A diferencia de varios autores, que utilizan sólo las GPUs descartando los cores de la arquitectura, nuestra estructura permite desarrollar aplicaciones que explotan toda la potencia de cómputo del cluster (cores y GPUs).

Asimismo, explicamos los pasos a seguir para compilar estas aplicaciones híbridas, lo cual no es trivial.

Evaluamos el algoritmo de suma por reducción, desarrollado utilizando el esquema propuesto, sobre un cluster de CPUs/GPUs heterogéneo. Distribuimos la carga de trabajo de acuerdo a las capacidades de los recursos de cómputo disponibles. Concluimos que es posible incrementar el rendimiento de la aplicación hasta un 10 % si se consideran todos los recursos de cómputo del cluster (CPUs y GPUs) respecto a utilizar sólo las GPUs.

Como trabajo futuro, planeamos resolver otros problemas regulares utilizando el esquema propuesto y analizar su rendimiento sobre clusters homogéneos y heterogéneos de CPUs/GPUs.

Referencias

1. Message Passing Interface Specification. <http://www.mpi-forum.org>
2. The OpenMP API specification for parallel programming. <http://www.openmp.org>
3. Posix Threads. <http://standards.ieee.org/develop/wg/POSIX.html>
4. Nvidia CUDA. <http://www.nvidia.es/object/cuda-parallel-computing-es.html>
5. OpenCL. <https://www.khronos.org/opencl/>
6. Manavski S.: CUDA Compatible GPU as an Efficient Hardware Accelerator for AES Cryptography. Proceedings of IEEE ICSPC 2007. 65–68 (2007)
7. Deguang L et al: Parallel AES algorithm for fast Data Encryption on GPU. Proceedings of ICCET 2010. 6, 1–6 (2010)
8. Eklunda A. et al: Medical image processing on the GPU – Past, present and future. Medical Image Analysis. 1073–1094 (2013)
9. Fan Z. et al: GPU Cluster for High Performance Computing. Proceedings of the ACM - IEEE Supercomputing Conference 2004. (2004)
10. Harris M.: Fast Fluid Dynamics Simulation on the GPU. GPU Gems. Pearson (2007).
11. Kindratenko V. et al: GPU Clusters for High-Performance Computing. Proceedings of the IEEE International Conference on Cluster Computing and Workshops. 1–8 (2009)
12. Yang C. et al: Hybrid Parallel Programming on GPU Clusters. International Symposium on Parallel and Distributed Processing with Applications. 142–147 (2010)
13. Yang C. et al: Hybrid CUDA, OpenMP, and MPI parallel programming on multi-core GPU clusters. Computer Physics Communications. 266–269 (2011)
14. Noaje G. et al: MultiGPU computing using MPI or OpenMP. 347–354 (2010)
15. Fengshun L. et al: Performance evaluation of hybrid programming patterns for large CPU/GPU heterogeneous clusters. Computer Physics Communications. 1172–1181 (2012)
16. Xiongwei F. et al: A secure and efficient file protecting system based on SHA3 and parallel AES. Parallel Computing 52. 106–132 (2016)
17. Diaz J. et al: A Survey of Parallel Programming Models and Tools in the Multi and Many-Core Era. IEEE Trans. Parallel Distrib. Syst. (2012)
18. AlJaroodi J. et al: Modeling Parallel Applications Performance on Heterogeneous System. Parallel and Distributed Processing Symposium, 2003. IEEE Computer Society. (2003)
19. Tinetti F. G., Tesis Doctoral: Cómputo Paralelo en Redes Locales de Computadoras. España: Universidad Autónoma de Barcelona. <https://ddd.uab.cat/pub/tesis/2004/tdx-1027104-173002/fgt1de2.pdf>

Método híbrido de reducción de incertidumbre aplicado a la predicción del comportamiento de incendios forestales

Miguel Méndez-Garabetti^{1,2}, Germán Bianchini¹, Paola Caymes-Scutari^{1,2} y María Laura Tardivo^{1,2,3}

¹ Laboratorio de Investigación en Cómputo Paralelo/Distribuido (LICPaD), Departamento de Ingeniería en Sistemas de Información, Facultad Regional Mendoza - Universidad Tecnológica Nacional. Mendoza, Argentina. (M5502AJE) Mendoza, Argentina.

² Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

³ Departamento de Computación, Facultad de Ciencias Exactas, Físico-Químicas y Naturales (UNRC), Córdoba, Argentina.

mmendez@mendoza-conicet.gob.ar, gbianchini@frm.utn.edu.ar, pcaymesscutari@frm.utn.edu.ar, lauratardivo@dc.exa.unrc.edu.ar

Abstract. Los incendios forestales causan anualmente grandes pérdidas y daños alrededor del mundo. El pronóstico del comportamiento de un incendio puede ser una herramienta fundamental para tomar decisiones en situaciones de emergencia. Sin embargo, este proceso suele estar afectado por la existencia de incertidumbre en las variables que alimentan al modelo. Por este motivo, desde diferentes áreas, se trabaja en el desarrollo y perfeccionamiento de métodos que permiten reducir los efectos de la incertidumbre y obtener predicciones más precisas. En este trabajo se presenta un método híbrido de reducción de incertidumbre que combina las virtudes de dos metaheurísticas poblacionales evolutivas: Algoritmos Evolutivos y Evolución Diferencial. Dicho método se denomina Sistema Estadístico Evolutivo Híbrido con Modelo de Islas (HESS-IM).

Keywords: predicción de incendios, reducción de incertidumbre, algoritmos evolutivos, evolución diferencial, metaheurísticas híbridas

1 Introducción

Los incendios forestales ocasionan anualmente grandes pérdidas y daños en diferentes áreas alrededor del mundo. Las consecuencias negativas tras la ocurrencia de estos fenómenos suelen ser muy significativas. Tal es el caso del daño en el suelo, el agua, la atmósfera, la economía, la flora, la fauna y en ocasiones se debe lamentar la pérdida de vidas humanas. Debido a esto es que, desde diferentes áreas de la ciencia, se trabaja en el desarrollo de herramientas, técnicas, métodos y sistemas que permiten minimizar tales efectos, ya sea antes, durante o después de los mismos. Una de estas técnicas tiene el objetivo de predecir el

comportamiento del fenómeno, es decir, se intenta determinar, con cierto tiempo de antelación, la dirección futura de propagación del incendio. Si la predicción generada por el método es de calidad y es obtenida antes que el incendio finalice, la misma se vuelve de vital importancia ya que puede permitir mejorar el impacto de la toma de decisiones, posibilitando la optimización del uso de recursos en las tareas de extinción, evacuación, planificación, etc.

El comportamiento del fuego en un entorno forestal está determinado por diferentes variables, tales como: la velocidad y dirección del viento, la cantidad y tipo de combustible, la topografía del terreno, el tipo de vegetación, entre otros. Esta información es necesaria para poder determinar el futuro comportamiento del fenómeno. Por lo general, obtener los valores de dichas variables en tiempo real resulta una tarea compleja. Si bien las redes de sensores inalámbricos (WSN, por sus siglas en inglés) [1] pueden ser de gran ayuda, ya que permiten obtener mediciones en tiempo real de las variables de interés, es prácticamente imposible contar con instrumental de medición en toda área forestal con potencial riesgo de ignición. En [2], las WSNs son utilizadas como herramienta para el monitoreo y detección temprana de incendios forestales. Más allá de que no es económicamente viable contar con WSNs desplegadas en grandes superficies, una alternativa interesante puede ser el uso de redes de sensores inalámbricas móviles (MWSNs) [3], en las cuales los sensores no se encuentran en ubicaciones fijas, pudiendo ser reutilizados en prácticamente cualquier área afectada por incendios. Este tipo de implementaciones puede reducir la imprecisión en los valores de entrada de las variables que alimentan el modelo e incidir en mejoras en la calidad de predicción del método.

Sin embargo, en la mayoría de los incendios forestales, no es posible contar con mediciones en tiempo real de las variables de interés. Debido a esto, los valores deben ser tomados a partir de mediciones indirectas, interpolaciones o aproximaciones, lo que nos ubica ante una situación de *incertidumbre*. La incertidumbre en los parámetros de entrada afectan de forma directa la calidad de predicción del método, por lo tanto, si deseamos obtener predicciones cercanas a la realidad, dicha incertidumbre debe ser reducida a valores aceptables. Los sistemas de predicción aplicados a fenómenos naturales son de alta criticidad debido a la sensibilidad de las decisiones que pueden ser tomadas a partir de sus predicciones y del poco margen de tiempo disponible para obtener los resultados. Es debido a esto que dichos sistemas suelen ser implementados en ambientes computacionales de alto rendimiento o HPC [4].

En este trabajo se presenta un nuevo método de reducción de incertidumbre, el cual saca provecho del esquema de paralelización basado en islas del método denominado *Sistema Estadístico Evolutivo con Modelo de Islas* [5],[6] (ESS-IM por sus siglas en inglés). Esta nueva metodología, llamada *Sistema Estadístico Evolutivo Híbrido con Modelo de Islas* (Hybrid Evolutionary-Statistical System with Island Model, HESS-IM), permite que cada isla pueda operar con diferentes metaheurísticas de forma colaborativa. HESS-IM utiliza Análisis Estadístico [7], Computación Paralela [4], metaheurísticas evolutivas paralelas (Algoritmos Evo-

lutivos [8] y Evolución Diferencial [9]) como herramientas colaborativas de optimización, bajo un esquema de paralelización basado en el Modelo de Islas [8].

La siguiente sección comienza con una breve descripción de la técnica de hibridación y las metaheurísticas utilizadas: Algoritmos Evolutivos (EAs) y Evolución Diferencial (DE). Seguido a esto se comenta el esquema de paralelización implementado y el funcionamiento interno de HESS-IM. Posteriormente se brindan los detalles de la experimentación realizada y se exponen los resultados obtenidos. Finalmente, se presentan las conclusiones y el trabajo futuro.

2 Sistema Estadístico Evolutivo Híbrido con Modelo de Islas (HESS-IM)

El *Sistema Estadístico Evolutivo Híbrido con Modelo de Islas* (HESS-IM), es un método general de reducción de incertidumbre que basa su funcionamiento en un esquema colaborativo entre diferentes metaheurísticas poblacionales. En esta primera aproximación, el método opera con Algoritmos Evolutivos y Evolución Diferencial.

2.1 Hibridación de las técnicas de optimización

La eficiencia de un método de optimización puede verse incrementada mediante la combinación de múltiples metaheurísticas, debido a la utilización de diferentes estrategias de búsqueda sobre un mismo espacio de soluciones. Este tipo de combinación da lugar a lo que se conoce como metaheurísticas híbridas (Hybrid Metaheuristics, HMs). Si bien existen diferentes enfoques para clasificar a las HMs, en [10] se dividen en dos grandes grupos: metaheurísticas híbridas de *combinación colaborativa* y HMs de *combinación integrativa*. Las HMs colaborativas tienen como propósito intercambiar información entre las diferentes técnicas de optimización, ya sea que operen de forma secuencial o paralela. En otras palabras, podemos decir que cada metaheurística opera de forma casi independiente, colaborando entre sí sólo mediante el intercambio de soluciones candidatas. En cambio, en las integrativas, una metaheurística utiliza a otra, operando esta segunda como parte de la primera de forma subordinada. En este esquema la colaboración es a nivel interno del mecanismo de optimización, donde el funcionamiento de una metaheurística se ve enriquecido por las bondades de otra/s. Es decir, no intercambian soluciones ya que ambas operan sobre el mismo conjunto de soluciones.

En HESS-IM la técnica de optimización consiste en una hibridación colaborativa entre un Algoritmo Evolutivo y uno de Evolución Diferencial, ambas son metaheurísticas poblacionales o de múltiples soluciones, las cuales operan mejorando de forma iterativa un conjunto de soluciones. A continuación se proporciona una breve descripción de cada una de ellas.

Algoritmos Evolutivos. Los EAs constituyen un método de búsqueda, aplicado a la resolución de problemas de optimización, inspirado en la teoría de

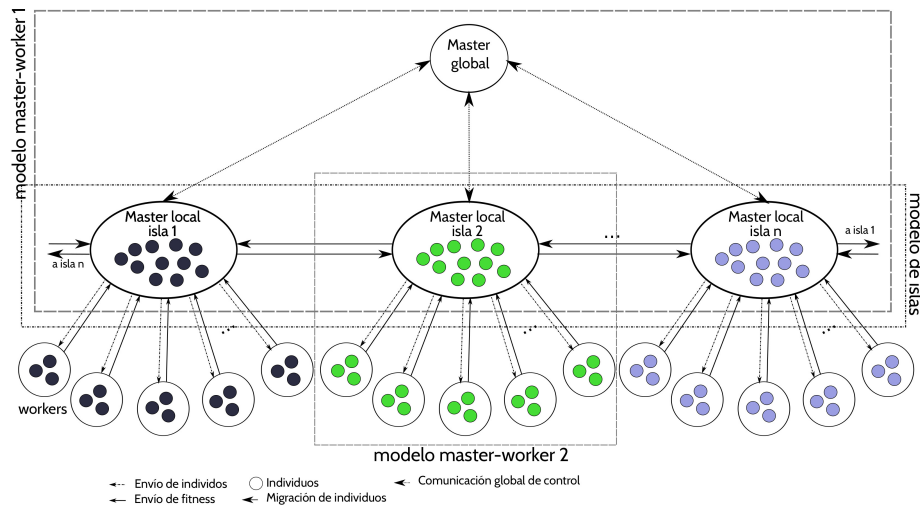


Fig. 1. Esquema de paralelización jerárquico basado en islas con doble modelo master-worker.

evolución natural de las especies [11]. El proceso se compone de cierto número de iteraciones, llamadas generaciones, donde un conjunto de soluciones candidatas, que conforman una población, evolucionan mediante la aplicación de operadores que permiten imitar los principios de selección natural y supervivencia del más apto. Cada individuo tiene asociado un valor de aptitud o fitness que determina la calidad de la solución para el problema tratado; este valor se calcula utilizando la función matemática que describe el problema. Los mecanismos que permiten evolucionar a los individuos de la población son denominados operadores evolutivos: selección, mutación, cruzamiento y reemplazo. En cada iteración, se selecciona una cantidad preestablecida de individuos, los cuales se reproducen utilizando operadores de variación y generando así nuevas soluciones candidatas. A continuación, se aplica un proceso de reemplazo para determinar qué individuos de la población sobrevivirán [12]. Al término de un número razonable de generaciones, se espera que el mejor individuo de la población represente a una solución lo suficientemente próxima a la solución óptima del problema.

Evolución Diferencial. Los DE, al igual que los EAs, son metaheurísticas del tipo poblacional estocástica. DE comienza explorando el espacio de búsqueda mediante la generación de una población de individuos, la cual se crea de manera aleatoria y uniformemente distribuida dentro del rango del problema. En términos generales DE utiliza la diferencia de vectores para modificar de forma iterativa la población mediante tres operaciones vectoriales: la mutación, el cruzamiento y la selección. Es importante remarcar que estos operadores son aplicados sobre cada uno de los individuos de la población. La operación de mutación perturba la población mediante la aplicación de diferencias de vectores

entre los miembros de la población, de esta forma se determina tanto el grado como la dirección de los nuevos individuos generados a partir de este proceso. Luego de la fase de mutación, que da como resultado el vector mutante, se somete el mismo junto con el individuo de la población actual al operador de cruce, generando un nuevo vector denominado vector de prueba. Finalmente, se lleva a cabo el proceso de selección con el objetivo de mantener en la población de la generación siguiente a aquellos individuos que posean mejores características, por lo tanto, se evalúa para cada vector de prueba el valor de su función objetivo y se compara con su correspondiente de la población actual. Si el vector de prueba tiene mejor o igual valor de la función objetivo, se reemplazará el vector actual en la siguiente generación.

2.2 Esquemas de Paralelización

Existe gran número de alternativas para paralelizar metaheurísticas: según la cantidad de poblaciones involucradas, el tratamiento de los operadores, el grado de relación entre cómputo y comunicación, entre otros. Existen ciertas estrategias de paralelización que mejoran el tiempo de resolución del algoritmo, por tratarse de implementaciones paralelas de un algoritmo secuencial, pero dicho paralelismo no influye en los valores numéricos, por lo que éstas implementaciones llegan a los mismos resultados que la implementación secuencial [13].

HESS-IM hereda de ESS-IM un esquema de paralelización basado en islas con doble jerarquía master-worker (un esquema general puede observarse en la Fig. 1). Como puede apreciarse, en el nivel superior se encuentra el *modelo master-worker 1*, donde un proceso *master global* coordina el funcionamiento de la totalidad de las islas. El master local de cada isla cumple la función de worker respecto del master global. Posteriormente en el segundo nivel de jerarquía master-worker, *modelo master-worker 2*, cada master de isla controla a un conjunto de workers dentro de cada isla, quienes son los encargados de evaluar la función de aptitud. Es importante remarcar que cada isla inicializa una población diferente, las cuales son evolucionadas mediante metaheurísticas distintas. La colaboración entre las diferentes metaheurísticas es provista mediante el operador de migración, el cual intercambia individuos entre las diferentes islas/metaheurísticas cada ciertos eventos (i.e., cantidad de generaciones realizadas, valores de aptitud de la población, detección de estancamientos en óptimos locales, etc.).

2.3 HESS-IM: Metodología de Funcionamiento

HESS-IM utiliza como técnica de optimización una metaheurística híbrida basada en Algoritmos Evolutivos y Evolución Diferencial. En esta primera aproximación del método, la hibridación se basa en un esquema colaborativo, donde las metaheurísticas operan con poblaciones diferentes y sólo intercambian individuos cada determinados eventos o cierta cantidad de generaciones.

En la Fig. 2, se presenta un esquema general del funcionamiento de HESS-IM. Tal como se puede observar, la estrategia de paralelización corresponde con el diseño presentado en la Fig. 1, donde existen tres tipos de procesos: a)

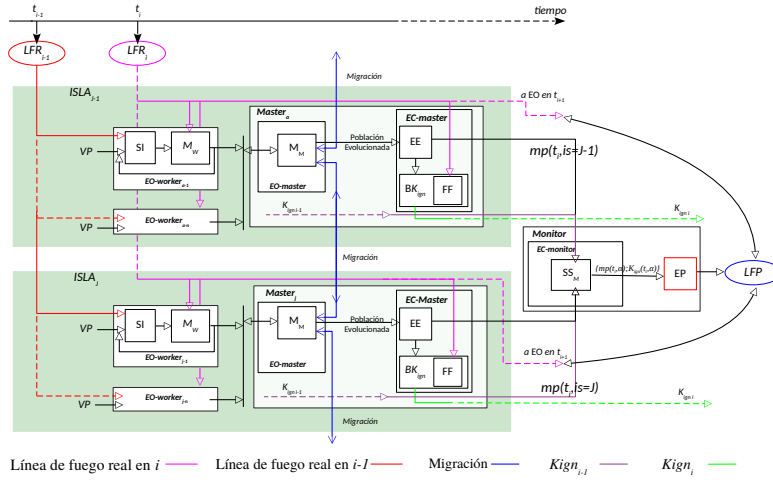


Fig. 2. Sistema Estadístico Evolutivo Híbrido con Modelo de Islas: **SI**: simulador de incendios; **M_M**: etapa metaheurística en master; **M_W**: etapa metaheurística en worker (evaluación de fitness); **EO**: etapa de optimización; **EE**: etapa estadística; **BK_{ign}**: búsqueda de K_{ign} ; **K_{ign}**: valor clave de ignición; **FF**: función de fitness; **EC**: etapa de calibración; **EP**: etapa de predicción; **LFP**: línea de fuego predicha; **LFR_x**: línea de fuego real en tiempo x ; **VP**: vector de parámetros de entrada; **EE_M**: etapa estadística en proceso monitor; **mp**: mapa de probabilidad.

proceso *monitor* (i.e., *master global* en Fig. 1), b) proceso *master* (i.e., *master local* en Fig.1), y c) los procesos de tipo *worker*. En una instancia de HESS-IM donde operen j islas, y w workers por isla, intervendrán en paralelo un total de $1 + j + w * j$ procesos, i.e., 1 proceso monitor, j procesos masters (uno por cada isla), y $w * j$ procesos workers.

Desde el punto de vista del funcionamiento del método, el proceso *monitor* es el encargado de iniciar la cadena de predicción mediante el envío de la información del incendio a ser analizado (i.e., mapa real del incendio, intervalos de tiempo a considerar, parámetros, etc.), a los *masters* de cada isla. Posteriormente, en cada *master* se ejecuta la Etapa de Optimización (EO_{master}), donde se da inicio a la metaheurística que cada isla tiene asignada (i.e., EA o DE). Tal como se puede observar, la Etapa Metaheurística se encuentra dividida en dos subetapas: por un lado la Etapa Metaheurística del proceso master (M_M), y por el otro, la Etapa Metaheurística de los procesos workers (M_W). En términos generales, diremos que la etapa M_W se encarga de la evaluación de aptitud de los individuos, y la etapa M_M se ocupa del resto de las operaciones involucradas en cada metaheurística. Los workers de cada isla evalúan la aptitud de los individuos mediante la utilización del simulador de incendios (*SI*). El *SI* se basa en el modelo definido por Rothermel [14] e implementado mediante la librería fireLib [15]. Para realizar la evaluación de aptitud en t_i es necesario contar con la línea de fuego real del incendio (LFR) en t_{i-1} (i.e., LFR_{i-1}) y los valores de

los parámetros de entrada, los cuales se almacenan en los vectores de parámetros (VP).

Una vez evaluados los individuos, éstos son enviados junto a su correspondiente valor de aptitud, a la etapa M_M . Esta etapa, además de realizar el resto de las operaciones de cada metaheurística (i.e., alteración de individuos, evolución de la población, etc.), se encarga de la migración de individuos hacia las islas vecinas. El proceso de migración se encuentra dividido en tres pasos: selección, envío-recepción y reemplazo. El proceso de *selección* consiste en escoger, de la población actual, aquellos individuos que serán enviados al resto de las islas. Se utiliza un criterio semi-elitista, donde el 50% de los individuos a migrar corresponde a los mejores y el otro 50% se selecciona de forma aleatoria. El siguiente paso, *envío-recepción*, es el mecanismo que efectúa la comunicación entre las diferentes islas, el cual envía y recibe individuos cada cierta cantidad de generaciones, también llamado frecuencia de migración, siguiendo una topología de anillo. La frecuencia de migración está establecida como un parámetro variable que depende de la cantidad máxima de generaciones. El proceso de *reemplazo* determina los individuos de la población actual que serán reemplazados por los que arriben provenientes de otras islas; el criterio utilizado es similar al proceso de selección, donde los peores individuos son reemplazados por los mejores y el resto son reemplazados de forma aleatoria.

Finalmente, cuando las poblaciones de las distintas islas han evolucionado, éstas son enviadas a la Etapa de Calibración del proceso master (EC_{master}). En esta etapa se calcula un mapa de probabilidad, en función de todos los individuos, el cual se utiliza para generar el valor clave de ignición (K_{ign} , por sus siglas en inglés). El K_{ign} representa el patrón de comportamiento del incendio y es obtenido en la etapa Búsqueda del K_{ign} (BK_{ign}).

En todo instante de tiempo i , en cada isla is , se genera un valor clave de ignición $K_{ign}(t_i, is)$ y un mapa de probabilidad $m_p(t_i, is)$, los cuales son enviados a la etapa de calibración del proceso *monitor* ($EC_{Monitor}$), donde en la etapa estadística (EE_M) genera en t_i un par $\{m_p(t_i, \alpha); K_{ign}(t_i, \alpha)\}$, donde α , indica el número de isla cuyo par de valores ha obtenido el mejor valor de fitness. Éstos valores finalmente son ingresados en la etapa de predicción (EP) la cual realiza el cálculo de predicción (i.e., línea de fuego predicha) para el próximo instante de tiempo $i + 1$, i.e., LFR_{i+1} .

3 Experimentación y Resultados

En este apartado se comentan los resultados preliminares obtenidos tras evaluar la calidad de predicción de HESS-IM en contraste con la metodología previa ESS-IM. Para ello, ambos métodos fueron aplicados a una quema real controlada realizada en Serra de Lousã (Gestosa, Portugal), en el marco del proyecto Spread [16]. La porción del incendio bajo análisis tiene una duración total de 10 minutos, a partir del minuto 2 hasta el minuto 12. Para la evaluación de la calidad de predicción se han definido pasos de tiempo discretos, los cuales representan el avance del frente de fuego. Cada intervalo tiene una duración de 2 minutos, dando

un total de 5 muestras (i.e., minuto 4, 6, 8, 10 y 12). Es importante remarcar que desde el instante t_0 (minuto 2) a t_1 (minuto 4), ambos métodos efectúan el cálculo del primer K_{ign} , lo que impide generar predicciones para el instante de tiempo inicial (t_1 , minuto 4), por estar ambos métodos en la etapa de calibración. Para ambos métodos el criterio de finalización de cada metaheurística consistió en alcanzar un umbral de la aptitud igual a 0,7, o llegar a un máximo de 200 iteraciones, y para los parámetros particulares de cada metaheurística se utilizó una configuración clásica de valores. Además, el tamaño de población para cada metaheurística fue establecido en 200 individuos.

En la Fig. 3, se puede observar la calidad de predicción alcanzada por cada método. En el eje de ordenadas se presenta la calidad de predicción o fitness, donde un valor igual a 1 equivale a una predicción perfecta, y un valor igual a 0, indica una predicción completamente errónea. El eje de abscisas contiene los instantes de tiempo para los cuales se ha calculado la calidad de predicción.

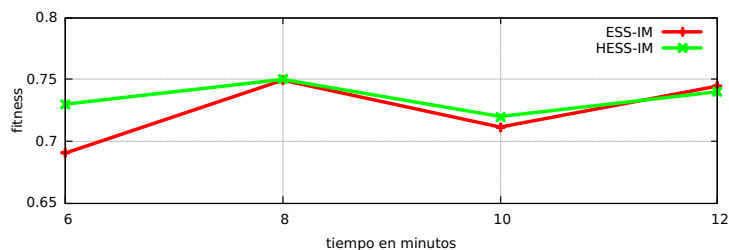


Fig. 3. Calidad de predicción obtenida por HESS-IM y ESS-IM para cada paso de predicción.

Tal como se puede observar, en la mayor cantidad de pasos de predicción (i.e., minuto 6, 8 y 10), la calidad de predicción alcanzada por HESS-IM es superior a la de ESS-IM, excepto en el último instante de tiempo (minuto 12). La mayor diferencia se observa en el primer paso de predicción y decae en los siguientes. A simple vista, puede parecer que la hibridación no impacta demasiado en la salida, pero es importante destacar que en este tipo de métodos, donde existe alta complejidad e incertidumbre en el fenómeno estudiado, suele ser difícil alcanzar incrementos mínimos de calidad de predicción, por lo que consideramos realmente positivos los resultados encontrados. Es importante remarcar que ambos métodos poseen un comportamiento no determinista, por lo que los resultados expuestos han sido calculados mediante el promedio de 30 ejecuciones.

Finalmente, es importante remarcar que debido a que este trabajo corresponde a un estudio preliminar, por el momento no se han realizado experimentaciones con mayor cantidad de casos, pero en próximos trabajos se incluirán mayor cantidad de resultados.

4 Conclusiones

En este trabajo se ha presentado HESS-IM, un nuevo método general de reducción de incertidumbre aplicado a la predicción de fenómenos naturales de propagación. Dicho método consiste en utilizar, como técnica de optimización, una metaheurística híbrida paralela basada en Algoritmos Evolutivos y Evolución Diferencial, bajo un esquema de paralelización basado en islas con doble jerarquía master-worker.

Se evaluó la calidad de predicción del método mediante su aplicación en la predicción del comportamiento de un incendio forestal controlado. Dicha calidad de predicción fue contrastada con los de una metodología previa basada en Algoritmos Evolutivos Paralelos (ESS-IM). Los resultados obtenidos demuestran que la técnica de hibridación utilizada, junto con las metaheurísticas seleccionadas, favorecen el mecanismo de búsqueda del método; permitiendo obtener resultados de mejor calidad. Dicho incremento en la calidad de los resultados es traducido en predicciones más precisas y confiables, lo que puede impactar en diferentes aspectos (optimizar el proceso de lucha contra incendios, minimizar riesgos y daños a las personas, mejorar el proceso de toma de decisiones ante situaciones de catástrofe, etc.).

Finalmente, como trabajo futuro se abre un gran abanico de temáticas y estudios pendientes. Por un lado, es preciso evaluar el método con una mayor cantidad de casos de incendios (i.e., con diferentes dimensiones, tipo de vegetación, condiciones climáticas, etc.). Por otro lado, se considera importante realizar un estudio de calibración de los parámetros evolutivos de cada metaheurística participante. Esto permitirá explotar las bondades de cada técnica de optimización y obtener así resultados de mejor calidad. Finalmente, se pretende realizar un análisis de rendimiento computacional, y evaluar la incorporación de otras metaheurísticas.

Agradecimientos

Este trabajo ha sido financiado por la Universidad Tecnológica Nacional mediante los proyectos EIUTIME0003939TC y EIUTNME0003952. Además, el primer autor agradece a CONICET por la beca doctoral otorgada.

Referencias

1. Sitharama Iyengar S., Brooks R.R.: Distributed Sensor Networks, Second Edition: Sensor Networking and Applications. Chapman and Hall/CRC, Florida (2013)
2. Aslan Y.E., Korpeoglu I., Ulusoy Ö.: A framework for use of wireless sensor networks in forest fire detection and monitoring, *Comput. Environ. Urban Syst.* 36, <http://dx.doi.org/10.1016/j.compenvurbsys.2012.03.002>, 614–625 (2012)
3. Rezazadeh J., Moradi M., Ismail A.S.: Mobile Wireless Sensor Networks Overview, *IJCCN International Journal of Computer Communications and Networks*, Volume 2, Issue 1, 17–22 (2012)

4. Buyya R.: High Performance Cluster Computing: Architectures and Systems, Prentice Hall, PTR, NJ, USA, (1999)
5. Méndez-Garabetti M., Bianchini G., Tardivo M.L., Caymes-Scutari P.: Comparative Analysis of Performance and Quality of Prediction Between ESS and ESS-IM, *Electronic Notes in Theoretical Computer Science*, Volume 314, SSN 1571-0661, <http://dx.doi.org/10.1016/j.entcs.2015.05.004>, pp. 45–60. Elsevier, Berlin (2015)
6. Méndez-Garabetti M., Bianchini G., Caymes-Scutari P., Tardivo M.L.: Increase in the quality of the prediction of a computational wildfire behavior method through the improvement of the internal metaheuristic, *Fire Safety Journal*, Volume 82, ISSN 0379-7112, <http://dx.doi.org/10.1016/j.firesaf.2016.03.002>, pp. 49–62. Elsevier, Berlin (2016)
7. Montgomery D.C, Runger G.C.: *Applied Statistics and Probability for Engineers*. 6th Edition. Limusa Wiley & Sons, New Jersey (2014)
8. Alba E., Tomassini M.: Parallelism and evolutionary algorithms, in *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 5, 443–462 (2002)
9. Das S., Mullick S.S., Suganthan P.N.: Recent advances in differential evolution – An updated survey, *Swarm and Evolutionary Computation*, Volume 27, ISSN 2210-6502, <http://dx.doi.org/10.1016/j.swevo.2016.01.004>, 1–30 (2016)
10. Blum C., Raidl G.R.: *Hybrid Metaheuristics, Powerful Tools for Optimization*, Springer International Publishing, Switzerland (2016)
11. Darwin D.: *On the Origin of Species by Means of Natural Selection*. London (1859)
12. Yu X., Gen M.: *Introduction to Evolutionary Algorithms*, Springer-Verlag, London (2010)
13. Gong Y., Chen W., Zhan Z., Zhang J., Li Y., Zhang Q., Li J.: Distributed evolutionary algorithms and their models: A survey of the state-of-the-art, *Applied Soft Computing*, Volume 34, pp. 286–300, Elsevier, Berlin (2015)
14. Rothermel R.C.: A mathematical model for predicting fire spread in wildland fuels, Res. Pap. INT-115, US Dept. of Agric., Forest Service, Intermountain Forest and Range Experiment Station. Ogden, UT., (1972)
15. Bevins C.D.: *FireLib User Manual & Technical Reference*, <http://www.fire.org/>
16. Viegas D.X.: *Project Spread – Forest Fire Spread Prevention and Mitigation*, <http://www.algosystems.gr/spread/>, (2004)

Análisis de uso de un algoritmo de balanceo de carga estático en un Cluster Multi-GPU Heterogéneo

Erica Montes de Oca¹, Laura De Giusti¹, Franco Chichizola¹,
Armando De Giusti^{1,2}, Marcelo Naiouf¹

¹ Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – Universidad Nacional de La Plata
La Plata, Buenos Aires, Argentina

² Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

{emontesdeoca, ldgiusti, francoch, degisuti, mnaiouf}@lidi.info.unlp.edu.ar

Resumen. En este trabajo se realiza un análisis de la influencia de usar un mecanismo de distribución de tareas estático en un cluster heterogéneo de GPUs, basado en la potencia de cómputo de las mismas. Se utiliza como caso de estudio el problema de atracción gravitacional de los cuerpos en el espacio (N -Body). Se presentan dos soluciones empleando una combinación de MPI-CUDA, que se diferencian en la forma de distribuir el trabajo: homogénea y heterogénea. Se detallan y analizan los resultados experimentales, mostrando un porcentaje de mejora de aproximadamente veinticinco por ciento al tener en cuenta las características de la arquitectura.

Palabras claves: GPU, Balance de Carga, Cluster Heterogéneo, MPI-CUDA, N -Body.

1. Introducción

En los últimos años, el avance de la GPU ha alcanzado a un conjunto de aplicaciones de propósito general, logrando disminuir sus tiempos de ejecución. El diseño de su arquitectura y su modo de procesamiento, han hecho de las mismas una opción a tener en cuenta a la hora de acelerar la ejecución de aquellas aplicaciones adaptables a estas plataformas [1][2]. La cantidad de transistores dedicados al cálculo intensivo de datos son la clave para que las GPUs sean más rápidas que las CPUs [3][4].

En las GPUs, la organización de los threads puede verse por nivel o jerárquicamente de la siguiente manera: grilla o grid (formado por bloques de threads), bloque (conformado por threads) y threads. En un bloque, los hilos están organizados en warps (todos los hilos de un warp son planificados juntos durante la ejecución). Por otro lado, el sistema de memoria de la GPU está compuesto por distintos tipos de memorias: Memoria Global, Memoria de Textura, Memoria Constante, Memoria Shared o Compartida, Memoria Local y Registro; los cuales se diferencian en la forma de acceso de los hilos, las limitaciones en las operaciones permitidas y su ubicación dentro del dispositivo [5][2].

En busca de lograr aún mayor potencia de cómputo pueden utilizarse más de una GPU en una misma arquitectura. En ese sentido surgen dos variantes: una solución orientada a arquitecturas de memoria compartida (múltiples GPUs en una máquina) y otra a arquitecturas distribuidas (cluster de múltiples máquinas cada una con GPU) [6]. También pueden combinarse ambas y formar un cluster de máquinas con múltiples GPUs. Este tipo de cluster en sí mismo es una arquitectura híbrida: el subsistema conformado por varios cores CPUs (y su sistema de memoria y entrada/salida) y el subsistema GPU (con sus memorias on y off chip) [7]. Estos dos subsistemas se comunican por medio de un bus PCI-E con una velocidad de ancho de banda baja comparada con las velocidades de comunicación de los sistemas de memoria. Por lo cual, el cuello de botella es la comunicación que existe entre dichos subsistemas [8] [9][10]. Si dejamos de lado el componente CPU en la arquitectura antes mencionada, podemos hablar de cluster heterogéneo cuando las características de las diferentes GPUs que intervienen son diferentes.

La aceleración que se logra con el uso de más de una GPU en la arquitectura varía dependiendo de la aplicación desarrollada. Por ejemplo, en [5] se compara el procesamiento del cifrado AES (Advanced Encryption Standard) en una GPU y en un cluster de GPU; en este caso no se obtiene una ventaja por el uso del cluster. En cambio en [11], donde se resuelve el problema N-Body, se logra una ventaja significativa al usar más de una GPU (todas con iguales características).

En este trabajo se presenta la resolución del problema N-Body utilizando un Cluster heterogéneo de máquinas con múltiples GPUs de diferentes características, y se analiza la mejora obtenida al distribuir el trabajo teniendo en cuenta la diferencia de potencia entre las GPUs. La Sección 2 plantea el problema a estudiar. En la Sección 3 se describen las soluciones implementadas en este trabajo. La Sección 4 contiene los resultados experimentales. La Sección 5 presenta las conclusiones y los trabajos futuros.

2. Problema de alta demanda computacional: *N*-Body

Como caso de estudio se presenta el problema *N*-Body, el cual consiste en simular el comportamiento (en cuanto al movimiento) de *N* cuerpos que componen el espacio de trabajo. Este problema es muy estudiado y aplicado en distintos campos de la ciencia tales como la química (por ejemplo, cálculo de la atracción de partículas subatómicas) o la astronomía (estudio de la atracción de los cuerpos en el espacio) [12][13]. En particular, el presente artículo se basa en la aplicación de la Ley de Atracción de Newton [14].

Inicialmente se cuenta con información de la masa (*m*), la velocidad (*v*) y la posición (*p*) de cada cuerpo. Posteriormente, en cada paso de simulación se debe calcular la nueva posición y velocidad de cada uno de los elementos en el espacio de simulación. Para esto es fundamental determinar la fuerza de atracción gravitacional entre cada par de cuerpos [13][15][2].

Por medio de la Ecuación (0) se calcula la *magnitud de la fuerza de atracción (F)* entre el par de cuerpos *i* y *j*. Este es el cálculo central de procesamiento.

$$F(i, j) = \frac{G \times m_i \times m_j}{r^2} \quad (0)$$

siendo r = distancia calculada a partir de p_i y p_j
G = constante gravitacional

La magnitud y la dirección (calculadas en base a la posición de ambos cuerpos) determinan la fuerza ejercida por el cuerpo *j* sobre el *i*. La *fuerza total* sobre un cuerpo *i* es la suma de las fuerzas ejercidas por cada uno de los otros cuerpos sobre él. A partir de la *fuerza total* ejercida sobre un cuerpo *i* y la información actual del mismo (velocidad, masa y posición) se calcula su nueva posición y velocidad.

El cálculo de la fuerza de atracción gravitacional es independiente para cada cuerpo en un paso de simulación, por lo que el problema es fácilmente adaptable a la arquitectura GPU [1].

En trabajos anteriores, se ha mostrado que la resolución de este problema usando una GPU se ve beneficiada de manera significativa al compararla con las versiones paralelas desarrolladas en memoria distribuida, memoria compartida y en la combinación de ambas para la arquitectura CPU [2]. Además, se ha planteado en [3][16], el uso de un cluster homogéneo de GPU obteniendo buenos resultados. En el presente trabajo se utiliza un cluster heterogéneo de GPUs, y se analiza cómo influye la forma de distribuir el trabajo en la solución.

3. Descripción de la solución

En este trabajo las soluciones implementadas se basan en el algoritmo *Fast N-Body Simulation*, el cual es apto para ser ejecutado en la GPU por la independencia de cómputo en el cálculo de la fuerza de atracción gravitacional de los cuerpos. Cada cuerpo debe resolver su fuerza gravitacional dependiendo de las posiciones de los demás sin modificar otro dato que no sea su propia fuerza. Del mismo modo, la actualización de posiciones y velocidades se realiza de manera independiente [17].

Para las implementaciones en las GPUs de este trabajo se utiliza CUDA, la cual permite definir una función denominada *kernel* que es ejecutada n veces, siendo n la cantidad de threads de la aplicación. Como el kernel se ejecuta en la GPU, la memoria en ella debe ser alocada antes de que la función sea invocada y los datos que requiera utilizar sean copiados desde la memoria de la CPU. Una vez ejecutada la función kernel, los datos de la memoria de la GPU deben ser copiados a la memoria de la CPU [8][9].

3.1. Solución con una GPU

En primera instancia, se comunican a la GPU los datos de los cuerpos previamente inicializados por la CPU a través del PCI-E. A continuación, la CPU delega la ejecución a la GPU, y queda en espera de la respuesta de la misma.

La información comunicada se almacena en la memoria global de la GPU en estructuras de datos utilizadas para mantener la velocidad, posición y masa de cada cuerpo, y para la fuerza de gravedad que se calcula.

En el kernel o función que calcula la fuerza de atracción gravitacional, cada thread calcula su posición en la memoria global que le permite acceder al cuerpo para el que tiene que calcular dicha fuerza. Se utilizan cuatro vectores con dimensión igual a la cantidad de threads por bloque, tres de los cuales son para almacenar una porción del vector de posiciones (una para cada coordenada), y el cuarto para una porción del vector de masas. Eso permite computar por bloque reduciendo el acceso a memoria global.

Además de su identificador en la memoria global, cada thread calcula su identificador dentro del vector en memoria compartida. Cada uno trae de memoria global la posición y la masa del cuerpo que le corresponde, por lo que el vector almacenado en memoria compartida es cargado por todos los threads, y la posición del cuerpo que le corresponde será almacenada en la dirección relacionada con su identificador dentro de la memoria shared. Antes de comenzar a calcular la fuerza de atracción, los threads son sincronizados para garantizar que todas las posiciones de los vectores de la memoria compartida hayan sido cargadas.

Luego, cada thread realiza el cálculo correspondiente a la fuerza de atracción para su cuerpo con las posiciones que se encuentran en los vectores de posiciones de la memoria compartida. Cuando todos los threads del bloque hayan utilizado todas las

posiciones de los vectores de la memoria shared, deben sincronizarse para volver a traer una porción más de las posiciones y de las masas de los cuerpos de la memoria global a la memoria compartida. Este ciclo se repetirá hasta que todo el vector de posiciones y masas haya sido utilizado. Finalmente, cada thread calcula la velocidad y posición del cuerpo que tiene asignado. Cuando los pasos de simulación a realizar se han completado, la GPU enviará a la CPU los resultados calculados, y le devolverá el control de la ejecución.

3.2. Solución con un cluster heterogéneo de GPUs

Basada en la solución de la Sección 3.1, se realiza una adaptación para ser ejecutada en un cluster de GPUs. Para esto se requiere la utilización de una combinación de MPI con CUDA. En este caso se utiliza un esquema master/worker de procesos que se comunican por medio de MPI. El *master* es el encargado de inicializar la información de los N cuerpos y distribuir el trabajo entre los *workers*, los cuales tienen asociada una GPU cada uno.

Cada worker (después de haber recibido del master la información necesaria) envía por medio de PCI-E dichos datos a su respectiva GPU, siendo éstas las que realmente realizan cómputo, ya que los procesos MPI se limitan a comunicar entre ellos los datos resultantes de cada paso de simulación, para que cada GPU conozca la información calculada por el resto de los procesos.

En la GPU, los datos son almacenados en la *memoria global* (off-chip) y luego son cargados a la *memoria shared* (on-chip) en porciones de 256 datos. La copia de los datos a esta última memoria, así como la reserva de espacio en la misma, es tarea del programador. Cabe destacar que al inicio del algoritmo se le debe enviar los datos de los N cuerpos a la GPU. Una vez que la GPU ha calculado la nueva posición y velocidad de los cuerpos que le corresponde, envía dichos datos a la CPU. A continuación, el proceso MPI comunicará esta información al resto de los procesos, recibiendo al mismo tiempo los datos calculados por los demás. Dicho procesamiento se repetirá t veces (donde t es la cantidad de pasos de simulación).

Un punto importante a tener en cuenta en esta solución es la forma en que se reparte el trabajo entre los *workers* (subconjunto de cuerpos que debe resolver cada uno), teniendo en cuenta que las GPUs asociadas a cada uno de ellos son diferentes. Para analizar el efecto de la distribución se utilizan dos alternativas para repartir las tareas: *homogénea* y *heterogénea*.

3.2.1. Distribución Homogénea

En el caso de la *Distribución Homogénea*, a todos los *workers* se les asigna la misma cantidad de cuerpos, es decir que cada GPU va a resolver la simulación de N/p cuerpos (siendo p la cantidad de *workers*, y suponiendo N múltiplo de p).

3.2.2. Distribución Heterogénea

En la *Distribución Heterogénea* se tiene en cuenta la potencia de cada GPU para repartir el trabajo en base a la misma, y de esta manera conseguir que todas trabajen aproximadamente el mismo tiempo en lugar de hacer la misma cantidad de tareas, buscando de esta manera reducir el desbalance de trabajo y como consecuencia el tiempo final de la aplicación.

Debido a que la potencia de cada GPU depende en gran medida de las características de la aplicación, se ejecuta la solución descrita en la Sección 3.1 en cada GPU, resolviendo un problema de tamaño reducido. A partir de esos tiempos de ejecución se calcula la *potencia de cómputo relativa* (pcr) de cada GPU con respecto a la mejor potencia por medio de la Ecuación (1). En este trabajo, la pcr de cada GPU se determina previo a la ejecución de la aplicación, pasando esta información como parámetro en el momento de ejecutarla.

$$pcr_i = \frac{Tiempo_{mejor\ GPU}}{Tiempo_i} \quad (1)$$

Cuando comienza la aplicación en el *master* se suman las potencias de cómputo relativa de cada GPU y se obtiene la *potencia de cómputo total* (pct) del cluster. En base a estos valores el *master* distribuye los cuerpos entre los *workers*. La *cantidad de cuerpos* (cc) para cada uno se calcula de acuerdo a la Ecuación (2).

$$cc_i = \frac{pcr_i \times N}{pct} \quad (2)$$

4. Resultados experimentales

En este trabajo, se utilizó un cluster heterogéneo de GPU, compuesto por 3 nodos (CPU) con 2 GPUs en cada uno. Las características de cada nodo son las siguientes:

- Quad Core Intel i5-2310 de 2,9 GHz, con dos GPUs Tesla C2050 con CUDA 4.2 y MPI 1.8.5.
- Dual Core Pentium G2020 de 2,9 GHz, con dos GPUs GeForce GTX 480 con CUDA 6.0 y MPI 1.8.4.
- Quad Core Intel i5-2310 de 2,9 GHz, con dos GPUs GeForce GTX 560 Ti con CUDA 7 y MPI 1.8.5.

En todas las pruebas realizadas se hicieron 10 pasos de simulación, y la cantidad de cuerpos (N) varió entre 1.024.000, 2.048.000 y 4.096.000. En las pruebas se utilizaron bloques de threads de 256 hilos para todas las GPUs.

En primera instancia se realizaron las pruebas utilizando una sola GPU (solución presentada en la Sección 3.1) para obtener el tiempo de ejecución de cada una de ellas y poder determinar la potencia de cómputo relativa (pcr). A partir de estas pruebas se obtuvieron las pcr que se muestran en la Tabla 1.

Tabla 1. Potencia de cómputo relativa (*pcr*) de cada GPU.

GPU	<i>Pcr</i>
GPU ₀ y GPU ₁ (Tesla C2050)	0,77
GPU ₂ y GPU ₃ (GeForce GTX 480)	1
GPU ₄ y GPU ₅ (GeForce GTX 560)	0,58

A partir de estas pruebas también se obtiene el mejor tiempo al usar una única GPU, el cual sirve de referencia para determinar la mejora conseguida al usar un cluster de GPUs. En la Tabla 2, se muestra el tiempo en segundos al usar la mejor GPU para los 3 tamaños de *N* (obtenidos como promedio de 15 ejecuciones para cada caso).

Tabla 2. Tiempo en segundos al usar la mejor GPU.

<i>N</i>	Tiempo (seg)
1.024.000	2624,64
2.048.000	10501,05
4.096.000	41980,43

En segunda instancia, se utilizó el cluster completo para realizar las pruebas correspondientes a las soluciones de la Sección 3.2.1 (*distribución homogénea*) y Sección 3.2.2 (*distribución heterogénea*). Para el segundo caso, se consideraron las *pcr* obtenidas en las pruebas anteriores (Tabla 1). En la Tabla 3 se muestran los tiempos en segundos obtenidos en estas pruebas (también como promedio de 15 ejecuciones en cada caso). En la Figura 1 se muestran estos valores.

Tabla 3. Tiempo en segundos de las pruebas en el cluster con distribución homogénea y heterogénea.

<i>N</i>	Tiempo (seg)	Tiempo (seg)
	Distribución Homogénea	Distribución Heterogénea
1.024.000	1262,20	953,52
2.048.000	4986,76	3765,59
4.096.000	19873,14	14938,82

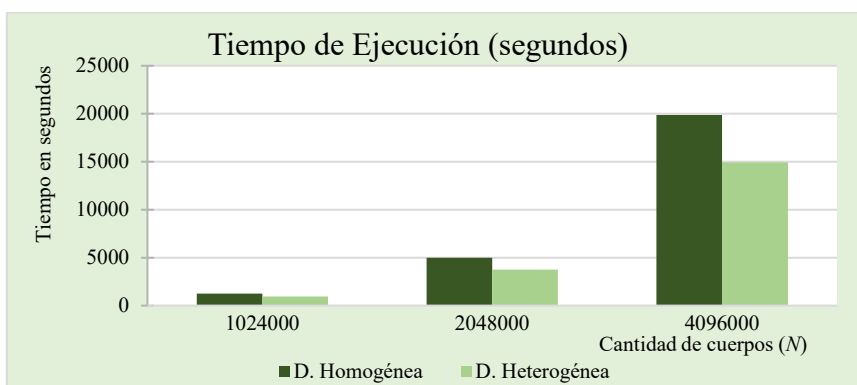


Figura 1. Tiempo en segundos de las pruebas en el cluster con distribución homogénea y heterogénea.

Al comparar los tiempos obtenidos con cada tipo de distribución se puede observar que al distribuir de acuerdo a la potencia de cada GPU (para esta aplicación en particular), se logra reducir el tiempo aproximadamente un 25%.

Se puede analizar la mejora conseguida al usar el cluster de GPUs si comparamos estos tiempos (Tabla 3) con los conseguidos al usar una sola GPU (Tabla 2). Esta mejora se calcula por medio de la Ecuación (3).

$$Mejora = \frac{Tiempo\ en\ una\ GPU}{Tiempo\ en\ Cluster} \quad (3)$$

A su vez, teniendo en cuenta que la máxima mejora que se puede obtener en el cluster está dada por la *potencia de cómputo total (pct)*, se puede estudiar la forma en que se aprovecha la arquitectura por medio de la Ecuación (4), siendo $pct = 4,7$ en esta arquitectura. En la Tabla 4 se muestra la *mejora* y el *aprovechamiento* al usar el cluster para ambos tipos de distribución.

$$Aprovechamiento = \frac{Mejora}{pct} \quad (4)$$

Tabla 4. Mejora y aprovechamiento al usar el cluster con distribución homogénea y heterogénea.

N	Distribución Homogénea		Distribución Heterogénea	
	Mejora	Aprovechamiento	Mejora	Aprovechamiento
1.024.000	2,0794	0,44	2,7526	0,58
2.048.000	2,1058	0,45	2,7887	0,59
4.096.000	2,1124	0,45	2,8102	0,60

En la Figura 2 se muestra el *aprovechamiento* conseguido con ambos tipos de distribución, donde se puede observar una mejora importante al distribuir el trabajo de acuerdo a la potencia de cada GPU.

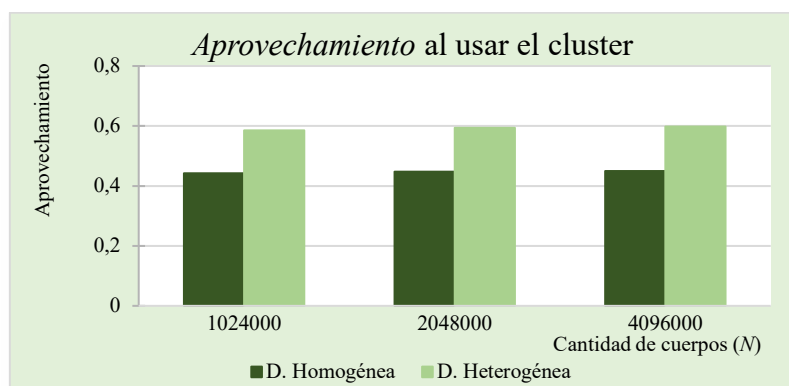


Figura 2. Aprovechamiento al usar el cluster con distribución homogénea y heterogénea.

Si bien la distribución heterogénea logra una mejora importante, la misma dista de aprovechar completamente el cluster debido a que en cada paso de simulación todos los procesos MPI deben sincronizar e intercambiar los nuevos datos de los cuerpos. Esto no sólo produce un retardo por el intercambio de mensajes entre los procesos, sino que genera un retardo mayor al tener que copiar nuevamente en la memoria de la GPU la información de todos los cuerpos.

5. Conclusiones y trabajos futuros

En este trabajo se presenta la solución a un problema con alta demanda computacional (N-Body) utilizando un cluster heterogéneo de CPUs y GPUs, analizando los resultados obtenidos al tomar en cuenta las características de las GPUs para la distribución del trabajo.

Puede observarse que la mejora alcanzada considerando las potencias de cómputo relativas, se encuentra en el orden del 25% para la solución sobre un cluster de 3 CPUs con 2 GPUs. Asimismo, el aprovechamiento de la arquitectura ronda el 60% para la solución mencionada, mientras que si no se consideran las características de las GPUs el mismo es del 45%.

Como trabajos futuros se plantea agregar más nodos en el cluster para analizar el comportamiento al escalar la arquitectura y aumentar la heterogeneidad. Y por otro lado, probar con aplicaciones que tengan características diferentes para realizar un estudio más completo de la influencia de esta distribución de trabajo en un cluster de GPUs.

Referencias

- [1] Jeroen Bédorf. *High Performance Direct Gravitational N-body Simulations on Graphics Processing Units*. Universiteit van Amsterdam (2007).
- [2] Montes de Oca Erica, De Giusti Laura, De Giusti Armando, Naiuof Marcelo. *Comparación del uso de GPU y cluster de multicores en problemas con alta demanda computacional*. XVIII Congreso Argentino de Ciencias de la Computación, pág. 267-275. (2012).
- [3] Montes de Oca Erica, De Giusti Laura, Rodriguez Ismael, De Giusti Armando, Naiuof Marcelo. *Análisis de la escalabilidad y el consumo energético en soluciones paralelas sobre cluster de multicores y GPU para un problema con alta demanda computacional*. XIX Congreso Argentino de Ciencias de la Computación (2013).
- [4] Pérez C., Piccoli M. F. *Estimación de los parámetros de rendimiento de una GPU*. Mecánica Computacional. Vol. XXIX, pp. 3155-3167 (2010).
- [5] Adrian Pousa, Victoria Sanz, Armando De Giusti. *Análisis de rendimiento de un algoritmo de criptografía simétrica sobre GPU y Cluster de GPU*. HPC La TAM 2013. (2013).

- [6] Volodymyr V. Kindratenko, Jeremy J. Enos, Guochun Shi, Michael T. Showerman, Galen W. Arnold, John E. Stone, James C. Phillips, Wen-mei Hwu. *GPU Clusters for High-Performance Computing*. Cluster Computing and Workshops, 2009. CLUSTER '09. IEEE International Conference on. 978-1-4244-5011-4. (2009).
- [7] Lingyuan Wang, Miaoging Huang, Vikram K. Narayana, Tarek El-Ghazawi. *Scaling Scientific Applications on Clusters of Hybrid Multicore/GPU Nodes*. ACM 978-1-4503-0698-0. (2011).
- [8] Chao-Tung Yang, Chih-Lin Huang, Cheng-Fang Lin, Tzu-Chieh Chang. *Hybrid Parallel Programming on GPU Clusters*. IEEE 978-0-7695-4190-7/10 (2010).
- [9] Nvidia Corporation. *NVIDIA CUDA C Programming Guide*. (2011).
- [10] Nvidia Corporation. *CUDA C Best Practices Guide*. (2012).
- [11] Montes de Oca Erica, De Giusti Laura, Chichizola Franco, De Giusti Armando, Naiuof Marcelo. *Utilización de un Cluster de GPU en HPC. Un caso de estudio*. XX Congreso Argentino de Ciencias de la Computación (2014).
- [12] Peter Berczik, Keigo Nitadori, Shiyang Zhong, Rainer Spurzem, Tsuyoshi Hamada, Xiaowei Wang, Ingo Berentzen, Alexander Veles, Wei Ge. *High performance massively parallel direct N-body simulations on large GPU clusters*. Astronomisches Rechen-Institut, ZAH, Heidelberg, Alemania, Main Astronomical Observatory, NASU, Kyiv, Ucrania, Centre for Astronomy, Heidelberg University, Heidelberg, Alemania. HPC –UA (2011).
- [13] Andrews Gregory R. *Foundations of Multithreaded, Parallel, and Distributed Programming*. Addison-Wesley (2000).
- [14] Tsuyoshi Hamada, Keigo Nitadori. *190 TFlops Astrophysical N-body Simulation on cluster of GPUs*. IEEE 978-1-4244-7558-2 (2010).
- [15] Bruzzone Sebastian. *LFN10, LFN10-OMP y el Método de Leapfrog en el Problema de los N Cuerpos*. Instituto de Física, Departamento de Astronomía, Universidad de la República y Observatorio Astronómico los Molinos, Uruguay (2011).
- [16] Pritish Jetley, Lukasz Wesolowski, Filippo Gioachin, Laxmikant V. Kalé, Thomas R. Quinn. *Scaling Hierarchical N-body Simulations on GPU clusters*. IEEE 978-1-4244-7558-2 (2010).
- [17] Lasr Nyland, Mark Harris, Jan Prins. *Fast N-body simulation with CUDA*. GPU Gem 3. cap. 31, pp. 677-695. Addison-Wesley (2007).

Un sistema de visión global paralelo para fútbol de robots con fines educativos

Rodrigo Cañibano, Eduardo Grosclaude, Javier Balladini

Universidad Nacional del Comahue
Buenos Aires 1400, Neuquén Capital, Argentina
{rcanibano, eduardo.grosclaude, javier.balladini}@fi.uncoma.edu.ar

Resumen Se presenta un nuevo sistema de visión global por computadora para el fútbol de robots de la *Small Size League*, que puede ser utilizado como herramienta educativa en una asignatura de visión por Computadora y permite explorar distintas estrategias de paralelización. El sistema procesa cuadros de vídeo y reporta la posición y orientación de los robots y la posición de la pelota en cada uno de ellos. Se aplicaron conjuntamente dos estrategias de paralelización. Una de ellas explota el paralelismo dentro de cada cuadro, dividiendo los cuadros en fragmentos que son procesados de forma independiente. La otra estrategia se basa en el procesamiento simultáneo de diferentes cuadros del vídeo. Se realizó una implementación en OpenMP para C++, basada en plugins para facilitar su modificación en un ambiente educativo. En un servidor con un procesador Intel Xeon E5-2630 (6 cores y multithreading simultáneo), la solución paralela respecto a la solución que utiliza un único núcleo (de los 6 disponibles) logró mejoras máximas de 5,27x en los FPS procesados mientras el retardo por cuadro se redujo a la mitad.

1. Introducción

La RoboCup[2] (del Inglés *Robot World Cup*) es una competencia internacional celebrada desde 1997, donde equipos de robots juegan una versión simplificada del fútbol. Su finalidad es la de poner a prueba los avances en distintas áreas de conocimiento como la inteligencia artificial, visión por computadora y robótica. En la liga de robots de tamaño pequeño denominada *Small Size League* (SSL) de la RoboCup los robots tienen capacidad de procesamiento reducida, delegando la toma de decisiones a la computadora de su equipo, que percibe el ambiente a través de un sistema de visión global centralizado compartido. Se utiliza un conjunto de cámaras montadas sobre distintas áreas del campo de juego, conectadas a una computadora donde se ejecuta el sistema de visión. El sistema detecta la posición y orientación de cada uno de los robots (en base a parches de colores) y la posición de la pelota, y reporta esta información a las computadoras que controlan los equipos.

Los criterios de calidad de este tipo de sistema son, normalmente: tasa de aciertos en la detección de objetos, precisión en la posición y orientación de los robots y pelota, cuadros por segundo (FPS, del inglés *Frames Per Second*) y el

retardo de cuadro (el tiempo desde la creación del cuadro hasta la entrega de la información extraída) máximo.

Desde sus orígenes, el tamaño de la cancha se ha ido incrementando. En el 2015 se establecieron como predeterminadas las canchas de tamaño doble para las cuales son necesarias cuatro cámaras en lugar de las dos utilizadas por las canchas de tamaño simple[1]. Duplicar la cantidad de información que se debe procesar impacta negativamente en el rendimiento de estos sistemas, debido a que se reducen los FPS procesados y se incrementa el retardo de cuadro. Para aumentar el rendimiento de estos sistemas es necesario implementar soluciones paralelas que aprovechen las múltiples unidades de procesamiento de las arquitecturas de cómputo actuales.

Actualmente el software de visión global utilizado en la SSL es el *SSL-Vision*[4]. Éste es un sistema basado en plugins, lo que le permite ser extendido fácilmente. Lamentablemente, presenta algunas desventajas en relación a la escalabilidad y al uso educativo. El sistema de captura de cuadros y el de procesamiento están fuertemente acoplados, formando parte del mismo hilo de ejecución. Esto impacta tanto en el rendimiento (sólo utiliza un núcleo de múltiples disponibles) como en la dificultad de utilizar el sistema como herramienta educativa en temas de paralelismo.

En [3] se presenta un framework destinado al uso educativo, basado en plugins organizados en pilas. El framework tiene dos hilos de búsqueda de objetos, uno para encontrar la pelota y otro para encontrar los robots, y un hilo de captura de cuadros de vídeo. Su grado de paralelismo es mejor que el de *SSL-Vision* pero aún así no permite escalar a más de tres núcleos.

El objetivo de este trabajo es el de crear un nuevo sistema de visión global por computadora para el fútbol de robots de la SSL, que pueda ser utilizado como herramienta educativa en una asignatura de visión por Computadora y permita explorar distintas estrategias de paralelización.

El resto de este trabajo está organizado de la siguiente manera. En la sección 2 se describe el sistema propuesto detallando las estrategias que se aplican, las tareas del sistema y cómo se dividen los cuadros. En la sección 3 se describen los resultados experimentales. La sección comienza explicando la metodología y plataforma experimentales; luego se presentan y analizan los resultados de los experimentos. Finalmente, en la sección 4, se discute el cumplimiento de los objetivos propuestos y se plantean líneas de trabajo futuras.

2. Descripción del sistema propuesto

El sistema debe procesar cuadros de un vídeo y reportar la posición y orientación de los robots y la posición de la pelota en cada uno de ellos. Este problema permite aplicar dos estrategias de paralelización:

- Dado que en un vídeo ya decodificado el procesamiento de cada cuadro es independiente del de los demás, se pueden procesar múltiples cuadros al mismo tiempo.

- Para realizar la búsqueda de los objetos, el cuadro puede ser dividido en varias partes que pueden ser procesadas en paralelo.

El sistema ejecuta tareas estáticas y tareas dinámicas. Las tareas estáticas son aquellas que permanecen en ejecución desde el inicio del programa hasta su finalización. Las tareas dinámicas son creadas para procesar un cuadro o fragmento específico y una vez que terminan de procesar el cuadro o fragmento finalizan.

Según su funcionalidad, las tareas son clasificadas en cuatro tipos:

Generación de cuadros: Es la tarea encargada de generar o capturar cuadros y colocarlos en la cola de cuadros a procesar. Ésta es una tarea estática, y sólo hay una en el sistema.

Generación de tareas de fragmentación de cuadro: Esta tarea crea una tarea de fragmentación de cuadro para cada cuadro en la cola. Ésta es una tarea estática, y solo hay una en el sistema.

Fragmentación de cuadro: Ésta es una tarea dinámica. Cada tarea de este tipo divide su cuadro en una cantidad predefinida de fragmentos de igual (o similar) tamaño. Luego, por cada fragmento, se crea una tarea de procesamiento de fragmento.

Procesamiento de fragmento: Ésta es una tarea dinámica. Cada una de estas tareas toma su fragmento asociado y lo procesa utilizando cada una de las pilas de plugins.

Cada una de las dos tareas estáticas tiene un hilo de ejecución asignado. Las tareas dinámicas son ejecutadas por un conjunto de N hilos de ejecución (N es un parámetro de entrada del programa). Cuando un hilo de ejecución (del conjunto) está libre, toma una nueva tarea dinámica para ejecutar. En la figura 1 se muestra la manera en que las tareas son asignadas a los hilos de ejecución.

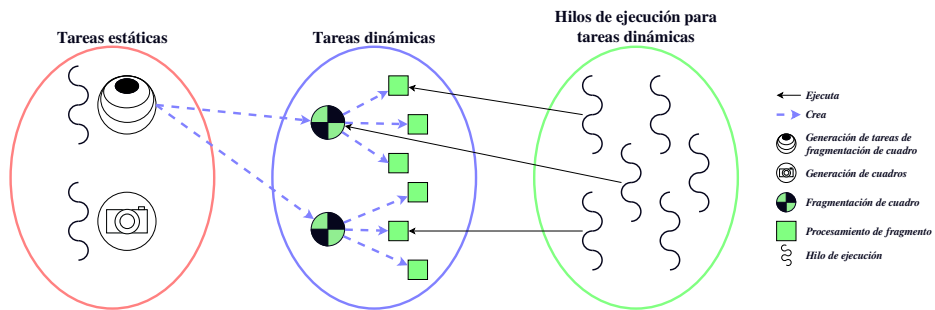


Figura 1. Asignación de tareas a los hilos de ejecución.

Si el cuadro se divide en partes sin zonas solapadas puede suceder que los parches de un robot se encuentren en fragmentos distintos. Para asegurar la detección de todos los parches, cada par de fragmentos adyacentes debe compartir una zona al menos igual al diámetro de un robot (Fig. 2).

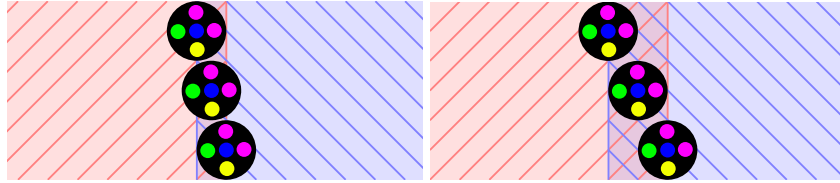


Figura 2. Izquierda: si el área compartida es demasiado pequeña y el robot se encuentra entre dos fragmentos, puede que sus parches no sean completamente visibles desde ninguno de ellos. Derecha: si el área compartida es del ancho de los robots, entonces todos los parches son visibles completamente desde por lo menos un fragmento.

El área compartida mínima resulta minimizando el perímetro de los fragmentos. Esta condición se logra cuando la relación entre su ancho y alto es lo más cercana posible a 1. Dado que el retardo de cuadro es igual al mayor tiempo de retardo de sus fragmentos, es importante balancear la carga. Por esto, todos los fragmentos serán de igual tamaño a excepción de aquellos que se encuentren en los bordes inferior y derecho (que podrían ser más pequeños).

En la figura 3 se muestra cómo se divide una imagen de 800x600 píxeles con un área compartida de 50 píxeles (igual al ancho del robot) en 1, 2, 5 y 8 fragmentos. Cuando el número de fragmentos es un número sin divisores, como en el caso de 5 fragmentos, el cuadro se divide en franjas de igual ancho o altura que el cuadro original. Este tipo de división produce fragmentos con perímetros mayores, produciendo grandes áreas compartidas.

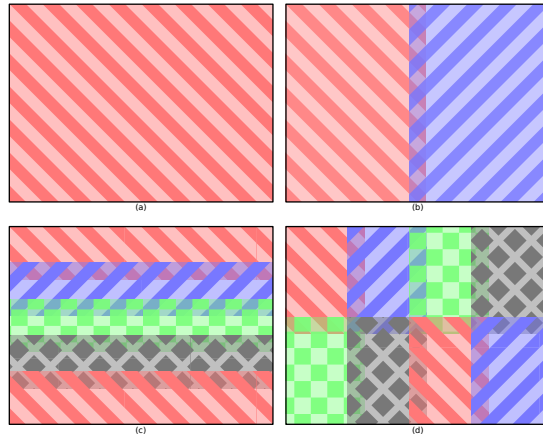


Figura 3. Resultado de dividir un cuadro de 800x600 píxeles con un área compartida de 50 píxeles en 1(a), 2(b), 5(c) y 8(d) fragmentos.

La implementación del sistema se realizó en OpenMP para C++ y está basada en plugins para facilitar su modificación en un ambiente educativo. Se reutilizaron los plugins implementados en [3], modificándolos para permitir su uso en un sistema paralelo.

3. Resultados experimentales

3.1. Metodología experimental

Se configuraron diferentes experimentos haciendo variar la cantidad de partes en las que fueron divididos los cuadros entre 1 y 24, la cantidad de hilos de ejecución para tareas dinámicas entre 1 y 12, y las resoluciones del vídeo entre 800x600 píxeles y 1280x720 píxeles. Para comparar los resultados de las diferentes configuraciones se midieron los FPS soportados por el sistema y el tiempo de retardo máximo (obtenido a los FPS soportados por el sistema bajo la misma configuración).

Como la obtención del retardo máximo depende del conocimiento previo de los FPS, cada experimento se dividió en dos partes. En la primera parte se busca la cantidad de FPS soportados por el sistema para esa configuración. Para esto, el programa procesa tantos cuadros del vídeo como pueda en un cierto intervalo de tiempo. En la segunda parte se limita la cantidad de cuadros por segundo (según el valor de FPS encontrado en la primera parte del experimento) y se registra el tiempo de retardo máximo. Dado que se trata de un sistema de tiempo real, es importante determinar las cotas inferiores del rendimiento. Por esto, se repitió cada experimento 10 veces, registrando sólo el peor valor obtenido para cada variable en cada configuración.

Dada la gran cantidad de experimentos, se buscó el tiempo mínimo de ejecución necesario para que éstos fueran representativos de una ejecución prolongada. Se realizaron tres ejecuciones de 6 minutos cada una con una misma configuración: 11 hilos y 10 fragmentos procesando el vídeo de 800x600 píxeles de resolución. El primer minuto no se contabilizó, con el fin de permitir que la ejecución se estabilizara. Los resultados se compararon con ejecuciones bajo la misma configuración, pero ejecutando de 11 a 20 segundos, y contabilizando sólo los últimos 10 segundos. En la tabla 1 se muestran los FPS obtenidos para cada ejecución. Se observa que las pruebas de 16 segundos son representativas de ejecuciones más largas. Dado que los 6 minutos de vídeo no caben en la memoria RAM del equipo de pruebas, el sistema fue modificado para reintroducir en la lista de cuadros a procesar aquellos cuadros ya procesados. Debido a esto, el rendimiento en estos experimentos es levemente inferior al resto con igual configuración.

3.2. Plataforma experimental

El equipo de pruebas cuenta un procesador Intel Xeon E5-2630. Este es un procesador de 6 núcleos con multithreading simultáneo de dos vías, una frecuencia básica de CPU de 2,3GHz y 2,8GHz en modo turbo, 15MiB de memoria

Duración	6'	11"	12"	13"	14"	15"	16"	17"	18"	19"	20"
Ejecución 1	166	171	170	170	170	169	166	166	166	166	165
Ejecución 2	166	171	170	170	170	168	167	166	166	165	166
Ejecución 3	166	171	170	170	170	169	166	166	167	166	165

Tabla 1. Búsqueda de duración mínima de los experimentos. Duración en segundos(") y minutos (').

caché L3 compartida entre todos los núcleos, y por cada núcleo hay 256KiB de caché L2 unificada, 32KiB de caché L1 para datos, y 32 KiB de caché L1 para instrucciones. El equipo posee 16GiB de memoria RAM.

3.3. Resultados

En la figura 4 se muestra la cantidades de FPS alcanzados para un vídeo de 800x600 píxeles y 1280x720 píxeles, para distintas cantidades de hilos de ejecución para tareas dinámicas y fragmentos. Note que el sistema ejecuta dos hilos más para las tareas estáticas, uno para la generación de cuadros y otro para la generación de tareas de fractura de cuadro. En el caso del vídeo de 800x600 píxeles se obtuvo un máximo de 195 FPS cuando se divide el cuadro en 15 fragmentos y se utilizan 10 u 11 hilos de ejecución para tareas dinámicas, mientras que para el vídeo de 1280x720 píxeles se alcanzó un máximo de 94 FPS cuando se divide el cuadro en 21 fragmentos y se utilizan 10 u 11 hilos de ejecución para tareas dinámicas. Con respecto a la solución que utiliza un solo hilo de ejecución para tareas dinámicas y un único fragmento, se obtuvo una mejora de 5,27x para el vídeo de 800x600 píxeles y una mejora de 11,75x para el vídeo de 1280x720 píxeles.

En la figura 4 se pueden observar dos patrones que ocurren en ambos vídeos. El primero es que si los cuadros se dividen en 7, 11, 17, 19 y 23 fragmentos, se produce una notable reducción en la cantidad de FPS con respecto a las divisiones adyacentes. Esto se produce porque los cuadros se dividieron en un número primo de fragmentos, y entonces el área total a procesar se incrementó de forma significativa con respecto a los valores circundantes. En la figura 5 se muestra la fluctuación del área para cada cantidad de fragmentos y los FPS obtenidos en divisiones de 1 hasta 100 fragmentos para un vídeo de 1280x720 píxeles de resolución, resaltando los resultados para valores primos y para los valores inmediatamente inferiores y superiores. Como puede observarse, a partir de 7 fragmentos el área total se incrementa de forma significativa cuando el número es primo con respecto a los valores cercanos con divisores. En estos casos, un pequeño incremento del número de fragmentos aumenta el paralelismo, pero los FPS se reducen debido a un gran aumento en el área a procesar.

El segundo patrón que se puede observar en la figura 4 es que cuando la cantidad de fragmentos es baja y la cantidad de hilos de ejecución para tareas dinámicas es alta, el sistema tiene menor FPS que utilizar una menor cantidad de hilos de ejecución para tareas dinámicas pero una mayor cantidad de fragmentos.

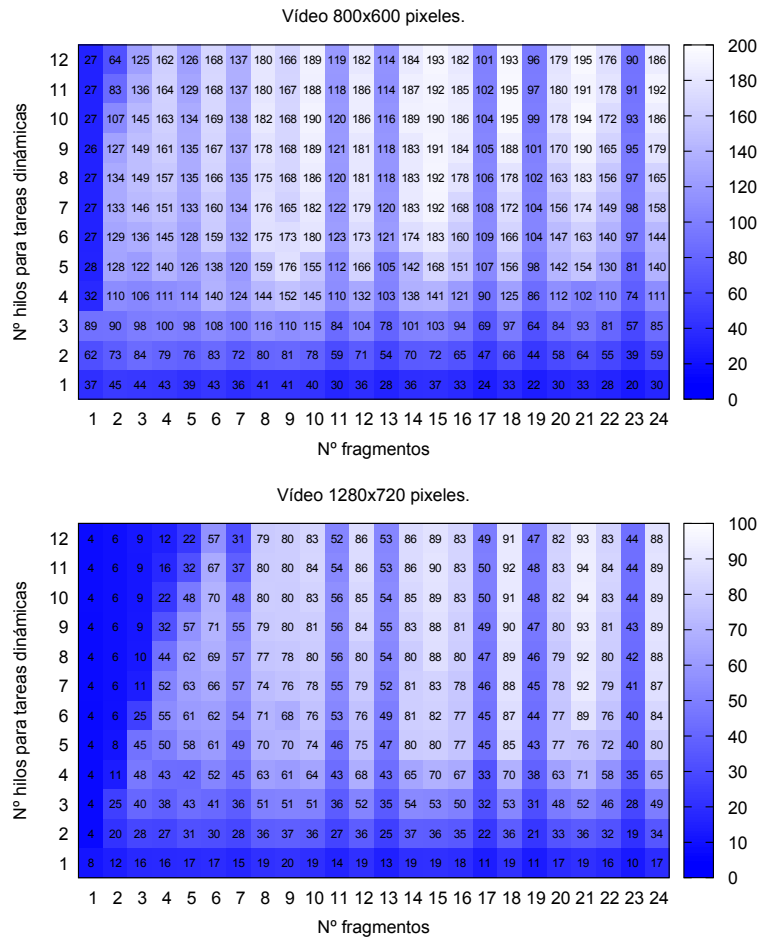


Figura 4. FPS para vídeos de 800x600 y 1280x720 píxeles

La memoria caché se comparte entre los cuadros que están siendo procesados en paralelo. Cuanto más cuadros se procesen en paralelo, mayor será la competencia por la memoria caché. La cantidad de cuadros que se procesan en paralelo está en relación con el número de fragmentos y la cantidad de hilos de ejecución para tareas dinámicas. Considerando que la cantidad de hilos de ejecución es mayor que la cantidad de fragmentos, si se disminuye la cantidad de fragmentos, manteniendo la cantidad de hilos de ejecución para tareas dinámicas, se incrementa la cantidad de cuadros a procesar en paralelo. Por lo tanto, en este caso, aumenta el número de fallos de caché.

Para comprobar que la hipótesis es correcta se diseñó el siguiente experimento. Se ejecutó el programa procesando cuadros de un vídeo de 1280x720 píxeles de resolución, y se midieron los fallos de caché en configuraciones de 1, 2, 6 y 12

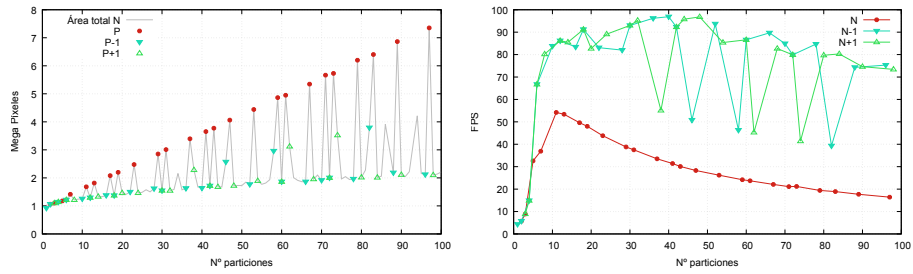


Figura 5. Comportamiento con números de fragmentos primos y sus inmediatos. A la izquierda el área total de la imagen, y la derecha los FPS.

fragmentos y cantidad de hilos de ejecución de tareas dinámicas de 1 a 11. Para obtener los fallos de caché de nivel 3 (es decir, cuando es necesario acceder a datos en la memoria RAM) se utilizó la herramienta *perf*. Ésta es una aplicación que permite acceder a los contadores de hardware del procesador que proveen datos muy precisos con un overhead despreciable.

En la figura 6 se muestran los fallos de caché por cuadro obtenidos en el experimento. Como se puede apreciar, a menor cantidad de fragmentos, mayor es la cantidad de fallos de caché ocurridos por cuadro.

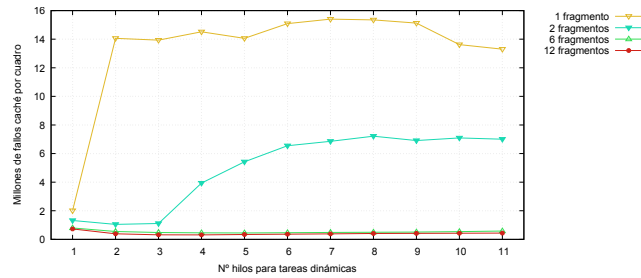


Figura 6. Fallos de caché por cuadro para distintas cantidades de fragmentos

Finalmente, en la figura 7 se muestran los tiempos máximos de retardo de los cuadros, para videos de 800x600 y 1280x720 píxeles de resolución (considerando que los FPS son limitados al máximo encontrado en la primera etapa de los experimentos). La figura 8 sintetiza la relación entre los FPS y el retardo de cuadros, mostrando el mínimo retardo de cuadros para los diferentes FPS que puede alcanzar el sistema en cada video.

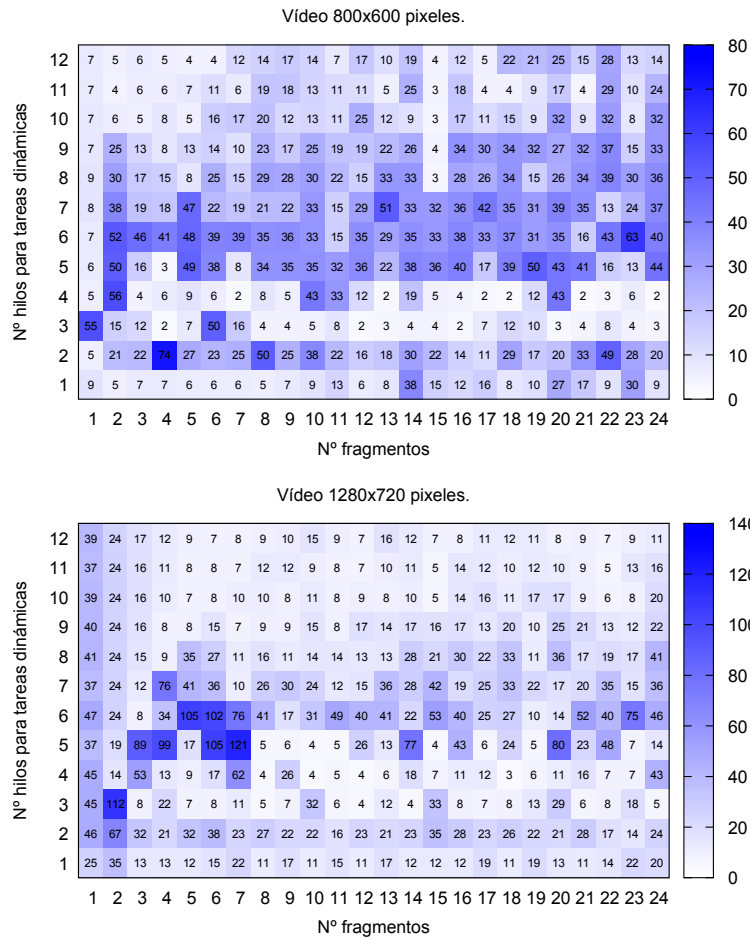


Figura 7. Retardo de cuadros para videos de 800x600 y 1280x720 píxeles

4. Conclusiones y trabajos futuros

En este trabajo se ha presentado un nuevo sistema de visión global por computadora para el fútbol de robots de la SSL, que puede ser utilizado como herramienta educativa en una asignatura de visión por Computadora y permite explorar distintas estrategias de paralelización. El sistema procesa cuadros de vídeo y reporta la posición y orientación de los robots y la posición de la pelota en cada uno de ellos. Se describieron dos estrategias de paralelización que son aplicadas en conjunto. Una de ellas explota el paralelismo dentro de cada cuadro, dividiendo los cuadros en fragmentos que son procesados de forma independiente. La otra estrategia se basa en el procesamiento simultáneo de diferentes cuadros del vídeo (independientes unos de otros). Se realizó una implementa-

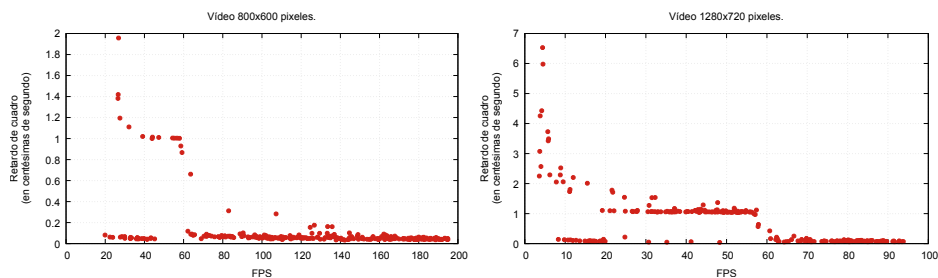


Figura 8. Retardo mínimo alcanzado para los FPS soportados por el sistema

ción en OpenMP para C++, basada en plugins para facilitar su modificación en un ambiente educativo. Se detallaron las tareas que ejecuta el sistema para implementar las estrategias de paralelización. Se estudió el comportamiento del sistema en términos de FPS y de retardo en el procesamiento de los cuadros para diferentes configuraciones del sistema en distintas resoluciones de vídeo. En un servidor con un procesador Intel Xeon E5-2630 (6 cores y multithreading simultáneo), la solución paralela respecto a la solución que utiliza un único núcleo (de los 6 disponibles) logró mejoras máximas de 5,27x en los FPS procesados mientras el retardo por cuadro se redujo a la mitad.

Como trabajo futuro se planea hacer una interfaz de usuario gráfica para configurar el sistema. Además, se cree conveniente implementar soluciones paralelas optimizadas de los plugins para utilizar el sistema en competencias de fútbol de robots.

Referencias

1. Small Size League Technical Committee. Laws of the RoboCup small size league 2015. http://robocupssl.cpe.ku.ac.th/_media/rules:ssl-rules-2015.pdf, March 2015. [Online 2016-03-10].
2. RoboCup. A brief history of robocup. <https://www.robocup.org/about-robocup/a-brief-history-of-robocup/>, 2016. [Online 2016-06-29].
3. Guillermo Torres, Javier Ballardini, and Rodolfo del Castillo. Un sistema de visión global para fútbol de robots. 2014. Trabajo de tesis para optar al grado académico de Licenciado en Ciencias de la computación.
4. Stefan Zickler, Tim Laue, Oliver Birbach, Mahisorn Wongphati, and Manuela Veloso. Ssl-vision: The shared vision system for the robocup small size league. In *RoboCup 2009: Robot Soccer World Cup XIII*, pages 425–436. Springer, 2010.

Optimización y computación paralela aplicados a mejorar la predicción de un simulador de cauce de ríos

Adriana Gaudiani¹, Emilio Luque², Pablo García³, Marcelo Naiouf⁴,
Armando De Giusti⁴

¹ Instituto de Ciencias, Univ. Nac. de General Sarmiento, Argentina

² Dpto. de Arquitectura de Computadores y Sistemas Operativos, España

³ Laboratorio de Hidráulica, Instituto Nacional del Agua, Argentina

⁴ III-LIDI, Fac. de Informática, Univ. Nac. de La Plata, Argentina

Resumen Este artículo presenta un método de optimización vía simulación que permite mejorar la predicción de un simulador computacional de cauces de ríos aprovechando los valiosos aportes del cómputo paralelo y distribuido. El problema a tratar para lograr este objetivo es la incertidumbre en los parámetros de entrada al modelo, pero buscar parámetros de simulación óptimos es un gran desafío, ya que el tamaño del espacio de búsqueda hace que sea un problema combinatorio con un alto costo de cómputo. La solución propuesta utiliza técnicas de optimización en el análisis de simulaciones, reduciendo el espacio de búsqueda e identificando regiones que permitan encontrar la configuración óptima en tiempos razonables. Se obtiene una muy buena aceleración del procesamiento que es potenciado al implementar el método con técnicas del cómputo de alto rendimiento (HPC).

Keywords: simulación computacional; optimización vía simulación; procesamiento en paralelo; calibración automática

1. Introducción

Los sistemas físicos son sistemas dinámicos que modifican sus variables constantemente. Un ejemplo es la simulación y predicción de inundaciones, provocadas por el desborde del agua que transporta el cauce de ríos, siendo un tema crítico para la población mundial [1]. Sólo las sucesivas calibraciones y validaciones del modelo permiten manejar errores en la entrada, como ser la incertidumbre en los parámetros, logrando que el modelo se ajuste lo mejor posible al comportamiento real del sistema. Normalmente son tareas muy costosas en recursos humanos, tiempo y dinero.

El objetivo principal de este trabajo es optimizar la predicción de un simulador hidrológico, mediante simulación. El método ofrece una calibración y sintonización del modelo con uso exclusivo de métodos computacionales que fueron posibles al utilizar los recursos que provee el HPC. Los resultados de este trabajo aprovechan y realimentan las investigaciones en curso desarrolladas por

el grupo *High Performance Computing for Efficient Applications & Simulation de la Universidad Autónoma de Barcelona*¹ [2], en colaboración con ingenieros del Instituto Nacional del Agua de Argentina (INA), quienes desarrollaron y nos facilitaron el simulador EZEIZA para avanzar con esta investigación.

Encontrar la solución óptima de esta manera implica encontrar un vector de parámetros de decisión optimizados que al ser ingresados al simulador, junto con los datos completos de un escenario de simulación, provea la menor diferencia entre los datos simulados y los datos observados. La gran dimensión de estos vectores impide implementar un método de búsqueda exhaustiva. La explosión combinatoria resultante convierten esta búsqueda en un método intratable computacionalmente.

El método que se presenta se sitúa entre los métodos de búsquedas exhaustivas y las heurísticas propias de la optimización, utiliza optimización vía simulación (OvS) y consta de dos etapas. La primera etapa es un proceso iterativo que aprovecha la estructura de vecindad del problema y mediante un método Monte Carlo (MMC) identifica una colección de regiones que acercan la búsqueda a la configuración óptima. Un método de agrupamiento K-Means determina la región prometedora donde buscar la solución. La segunda etapa implementa una búsqueda exhaustiva que encuentra la solución, pero ahora sobre una región mucho más pequeña. A pesar de la reducción del espacio de búsqueda, ambas etapas implementan métodos costosos computacionalmente, que necesitan los recursos del cómputo paralelo y distribuido.

La mejora en la predicción del simulador, con los distintos casos de experimentación que se implementaron, fue del 13 – 19 % sobre los errores de salida en un tramo del río. La aceleración en el tiempo de ejecución se compone de una ganancia propia del método de reducción del espacio de búsqueda y una ganancia en la implementación con HPC. El método computacional es altamente paralelizable. La aceleración lograda con el método de OvS es de 10X/20X en la experimentación presentada.

El resto del artículo está organizado de la siguiente manera: en la sección 2 se detalla el problema a solucionar. En sección 3 se brindan los detalles del método de optimización vía simulación aplicado. En sección 4 se describe la experimentación y los resultados de aplicar procesamiento en paralelo, finalmente en sección 5 se discuten las conclusiones y trabajos futuros.

2. Identificación del Problema

La simulación del cauce de un río representa eventos de inundación provocado por el desborde del cauce natural de ríos. Estos eventos se describen con modelos matemáticos que se basa en las leyes físicas que gobiernan su comportamiento. La certeza en la salida de estos modelo, independientemente de su dimensión, depende en gran medida en la disponibilidad de datos de entrada actualizados. Por otro lado, los errores en datos y parámetros constituyen la principal fuente

¹ <http://grupsderecerca.uab.cat/hpc4eas/>

de incertidumbre en los resultados de la simulación [3]. En este trabajo se busca una solución a este problema y provee un método computacional de calibración del modelo, que disminuye la incertidumbre de sus parámetros encontrando un conjunto optimizado de sus valores.

2.1. La Incertidumbre de los Datos

Los modelos hidrológicos necesitan al menos los siguientes datos para computar el nivel del agua y la velocidad del flujo en puntos seleccionados del cauce:

- Las secciones que dividen el dominio del río y los intervalos de discretización.
- Los valores de parámetros que definen el sistema.
- Las condiciones de contorno.
- Las condiciones de borde.

Los parámetros pueden evaluarse a partir de datos de campo, por ejemplo, observaciones hidro-meteorológicas, mapas de la topografía, tipos de suelo, etc. Sin embargo, rara vez están disponibles datos de campo completos para validar y determinar con certeza los parámetros del modelo.

2.2. EZEIZA: El Simulador del cauce del Río Paraná

Se seleccionó el simulador de cauces Ezeiza, el cual fue de sumo interés por ser parte del sistema computacional utilizado para el pronóstico que actualmente lleva adelante el *Servicio de Información y Alerta Hidrológico de la Cuenca del Plata*, correspondiente al INA, brindando alertas a la población que reside en zonas de influencia del Río Paraná. La red de cálculo, representada en la Fig. 1, se divide en 76 secciones transversales, que discretizan el Río Paraná, desde el embalse de Yacretá (Corrientes) hasta la ciudad de Villa Constitución (Santa Fe). Una amplia información puede encontrarse en [4].

Los parámetros seleccionados como los más sensibles del modelo son los siguientes: *la resistencia hidráulica*, que se representa con el valor del coeficiente de rugosidad de Manning, tanto para el cauce del río como para la planicie adyacente [5], y *la altura de los albardones*.

3. Optimización vía Simulación para mejorar la predicción

La búsqueda de un mejor rendimiento del modelo de simulación consiste en repetidamente analizar su salida y mediante sucesivos cambios en los valores de sus parámetros, continuar hasta obtener la mejor salida simulada. La Optimización vía Simulación (OvS) es un campo emergente que integra técnicas de optimización y de análisis de la simulación, el cual se ha convertido en una poderosa técnica para analizar y optimizar estos modelos. En principio, hay una separación entre el modelo que representa el sistema y el procedimiento que es usado para resolver el problema de optimización definido con este modelo. El modelo puede tener modificaciones sin afectar el proceso de optimización que sigue siendo el mismo [6] [7].

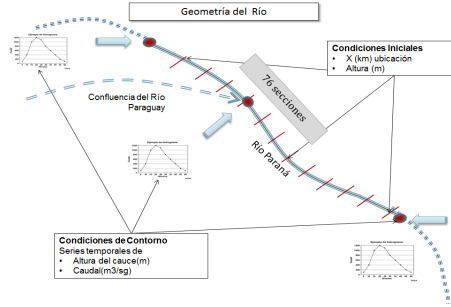


Figura 1. Red de cálculo del modelo hidrodinámico.

3.1. El problema de optimización

El problema de optimización es de múltiples objetivos, siendo necesario optimizar todos los objetivos de forma simultánea, aunque la mayoría de las veces, se encuentran en conflicto entre ellos y se debe encontrar un balance en la solución. Formalmente, la optimización está asociada con la especificación de una función matemática f llamada función objetivo y sus restricciones, de esta forma:

$$\begin{aligned} \max / \min f(x) \\ \text{sujeto a } x \in S \end{aligned} \quad (1)$$

donde x es la variable, f es la función ($f : S \rightarrow \mathfrak{R}$), S es el conjunto de restricciones y se cumple que $\exists x_0 \in S$ tal que $f(x_0) \preceq f(x) \forall x \in S$, para minimización, y $f(x_0) \succeq f(x) \forall x \in S$ para maximización.

En el contexto de este trabajo, el problema principal es un *problema de búsqueda*, el cual es considerado como sinónimo de *problema de optimización* y puede expresarse así:

$$\text{encontrar } x^* \text{ que optimiza } f(x), \forall x \in S$$

donde $f : \mathfrak{R}^p \rightarrow \mathfrak{R}$ es la expresión de la función objetivo o índice de calidad. x^* es el vector de solución óptima. El *espacio de búsqueda* está representado por el dominio $S \subseteq \mathfrak{R}^p$. La especificación de este espacio proviene de un conjunto de restricciones impuestas sobre S , las que restringiendo su tamaño, delimitan la *región de factibilidad*.

El objetivo de la optimización es identificar los valores óptimos de los parámetros que minimizan la diferencia entre la salida del simulador y los datos reales que se miden en el río. Se aplica optimización vía simulación debido a la complejidad del problema de optimización cuya función objetivo no puede resolverse analíticamente.

Ezeiza es considerado como una caja negra en el proceso de simulación, el cual requiere ser alimentado con cada escenario de simulación y el modelo devuelve los datos simulados (de salida) para el tiempo correspondiente a la simulación.

Con cada ejecución se obtiene un *índice de calidad* de la simulación, el cual mide las diferencias entre los valores simulados y los observados.

En cada una de las 76 secciones que dividen el dominio del Río Paraná se determinan los valores de los parámetros del modelo utilizados en la calibración: *Manning de cauce*, Mc , y *Manning de planicie*, Mp , y *altura de los albardones*, Al . Sus valores son reales y están definidos sobre intervalos continuos pudiendo tomar infinitos valores dentro de dichos intervalos. Los rangos asociados a cada parámetro deben ser discretizados. Tratar estos intervalos continuos como discretos nos permite ver el problema de optimización como un problema discreto combinatorio.

En la tabla 1 se muestra el excesivo crecimiento de la cantidad de escenarios en función de las secciones que se consideran, considerando solamente Mc y Mp . La explosión combinatoria y el enorme espacio de búsqueda caracteriza el problema. La cantidad de escenarios en función del número de secciones considerada (NSC), se obtiene como $(Mp * Mc)^{NSC}$.

Tabla 1. Explosión combinatoria de escenarios en función de las secciones consideradas

Cardinalidad		Número de Secciones Consideradas					
Mc	Mp	2	4	6	8	10	76
2	4	64	4096	262144	1,7 E+07	1,1 E+09	4,3 E+68
3	5	225	50625	1,1 E-07	2,6 E+09	5,8 E +11	2,4 E+89
4	5	400	160000	6,4 E+11	2,6 E+10	1,0 E+13	7,6 E+98
5	6	625	390625	2,4 E+08	1,5 E+11	9,5 E+13	1,8 E+106

3.2. Métricas de Evaluación de la Optimización via Simulación de Ezeiza

El índice utilizado como indicador del rendimiento general de la simulación, IC , depende de los datos de salida simulados, Q_{sim} , y los datos observados, Q_{obs} , en cada una de las 15 estaciones de monitoreo ubicadas sobre el dominio del río. El período de simulación, n , es medido en días. La siguiente ecuación define el índice de calidad de cada escenario de simulación, en función de los índices medidos en cada estación Ie_j :

$$IC = \frac{\sum_{j=1}^{15} Ie_j}{15} \quad (2)$$

$$\text{donde } Ie_j = \sqrt{1/n \cdot \sum_{i=1}^n (Q_{sim}^i - Q_{obs}^i)_j^2} \quad (3)$$

Un aspecto interesante de esta mejora es brindar un escenario de simulación que mejore los resultados que el INA logra actualmente con su escenario de

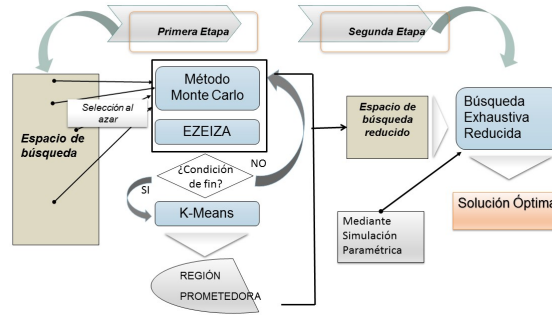


Figura 2. Método de Optimización vía Simulación de Dos Etapas.

pronóstico. Es necesario entonces definir un índice que mida esta mejora, IM , siendo su expresión:

$$IM = (\# \text{ Estaciones} : (IC_j^{Sim} < IC_j^{INA})_{j=1}^{15}) \quad (4)$$

IM mide la cantidad de estaciones cuyo índice de calidad, IC_j , sea menor que el logrado con el escenario INA, IC_{INA} , para las 15 estaciones de monitoreo.

3.3. Esquema de reducción propuesto

El método completo de OvS para sintonizar Ezeiza se grafica en la Fig. 2.

En una primera etapa, una combinación del método Monte Carlo + K-means, devuelve un espacio de búsqueda acotado que contiene los datos de los escenarios de simulación con mejores valores promedio del índice de calidad, índice que mide la calidad de simulación con cada escenario. La técnica implementada es un MMC basado en el algoritmo Metrópolis [8] combinado con un método K-means. El MMC selecciona, en cada iteración, 100 escenarios al azar del espacio total de posibles escenarios de simulación que pueden ejecutarse con Ezeiza. Selecciona los que tienen la mejor media y desviación standard que los ya acumulados. Luego el método K-Means identifica conjuntos de regiones prometedoras delimitadas por hiperplanos establecidos por los escenarios encontrados. De esta manera, se obtiene un nuevo espacio de búsqueda reducido. Sobre esta zona se implementa la segunda etapa del método que consiste en un método de búsqueda exhaustiva encontrando así la solución óptima en la región de factibilidad final. Se ofrece más información en [9].

4. Experimentación

A fin de validar la metodología propuesta, se presentan los resultados de una de las experimentaciones realizadas, basadas en simulaciones de $n = 365$ días,

correspondientes al año 1999, período caracterizado por bajantes y subidas en los niveles del cauce del río Paraná.

Los rangos del dominio de los parámetros son:

- Valores de Mc en $[0,015 \dots 0,045]$, tomado en pasos de 0,015. $\#Mc = 3$
- Valores de Mp en $[0,2 \dots 0,4]$, tomado en pasos de 0,1. $\#Mp = 4$
- Valores de Al en $[0 \dots 15]$, tomado en pasos de 5. $\#Al = 4$

Se utilizaron 3 secciones ubicadas en el bajo Paraná: 72, 74 y 76. El espacio de búsqueda, bajo estas condiciones, tiene $(4 \star 3 \star 4)^3$, o sea 110.592 configuraciones diferentes.

La expresión del índice de calidad IC es la siguiente:

$$\text{Minimizar } IC : f(Q_{sim}, Q_{obs}) \quad (5)$$

$$\begin{aligned} \text{donde } 0,2 &\leq Mp \leq 0,4 \\ 0,015 &\leq Mc \leq 0,045 \\ 0 &\leq Al \leq 15 \end{aligned} \quad (6)$$

La expresión del índice IM es la siguiente:

$$\begin{aligned} \text{Maximizar } IM &: f(Q_{sim}, Q_{obs}, Q_{INA}) \\ \text{tal que } \frac{1}{k} \cdot \sum (IC_{\hat{e}}^{Sim} - IC_{\hat{e}}^{INA}) &< tol \end{aligned} \quad (7)$$

donde k representa la cantidad de estaciones \hat{e} para las cuales se cumple:

$$IC_{\hat{e}}^{Sim} - IC_{\hat{e}}^{INA} < tol$$

donde tol es una tolerancia que se estableció en 0.1, según se acordó con los especialistas del INA. Se mantienen las restricciones (6) sobre las variables de decisión.

En la primera etapa, el MMC procesa 52 conjuntos de 100 escenarios para devolver el mejor conjunto de 100 puntos, o sea el que contiene el mínimo valor de IC . Se necesitaron 5200 ejecuciones del simulador para obtenerlo. Luego el método K-Means identifica 3 grupos, que se muestran en la Fig. 3. El grupo identificado con las mejores configuraciones contiene 8 puntos con valores mínimos. Estos puntos son 8 vectores de variables de decisión, que determinan los rangos de valores acotados que éstas pueden tomar y definen así el nuevo espacio de búsqueda reducido, sobre el cual se realiza la búsqueda exhaustiva. El nuevo espacio tiene 192 combinaciones.

La Fig. 4 compara los IC obtenidos para cada estación de monitoreo, con el escenario óptimo obtenido con el método de OvS y el escenario INA. En este gráfico puede apreciarse las mejoras logradas en tres ciudades del tramo inferior del Río Paraná.

La solución resultante es la configuración óptima mostrada en la tabla 2. Esta solución corresponde al escenario que tiene el *mínimo* IC del conjunto y a la vez *maximiza* IM . El índice IC óptimo del método es 0.801 y el valor de IM es: 3 estaciones. La configuración obtenida se muestra en la tabla 3.

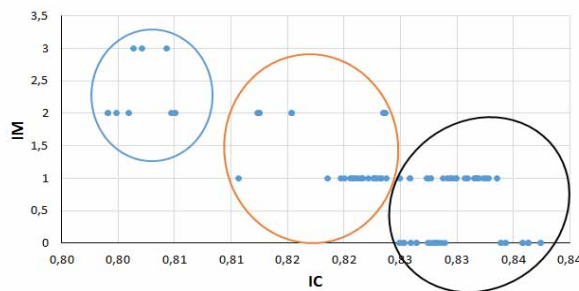


Figura 3. Grupos de configuraciones identificados por K-Means, considerando IC & IM .

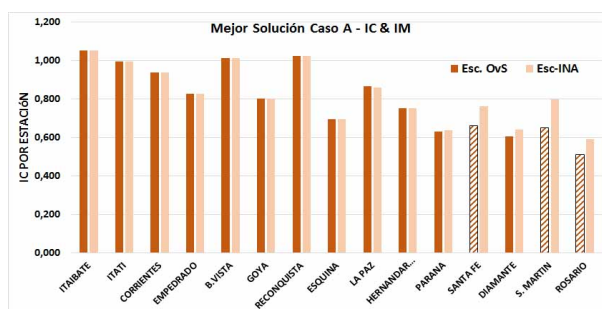


Figura 4. Comparación entre los valores de IC para el Esc.OvS en cada estación y la configuración INA tomando como objetivos IC & IM .

4.1. Implementación en Paralelo del Método OvS

Todas las simulaciones del método de OvS que se describen en este capítulo fueron implementados y ejecutados en equipamiento del Instituto de Investigación en Informática LIDI (III-LIDI) de la UNLP. Se utilizó un cluster de multicores HP Blade que consiste de 16 nodos, donde cada uno cuenta con 2 procesadores quad-core Intel Xeon e5405 de 2.0GHz. A su vez, cada núcleo cuenta con una caché L1 privada de 64KB (dividida en 32KB para instrucciones y otros 32KB para datos) y una caché L2 de 6MB compartida de a pares de núcleos. Con res-

Tabla 2. Solución óptima del método con **Tabla 3.** Mejora lograda en las estaciones objetivos: $IC + IM$. del cauce.

Solución Óptima - $IC + IM$			
Sección	Mc	Mp	Al
1	0,02	0,2	5
2	0,02	0,2	5
3	0,04	0,2	0

Estación	Escenario Mejora		
	OvS	INA	
Santa Fe	0,662	0,764	13 %
San Martín	0,651	0,799	19 %
Rosario	0,512	0,59	13 %

Tabla 4. Tiempo de ejecución y ganancia lograda en cada etapa del método de OvS.

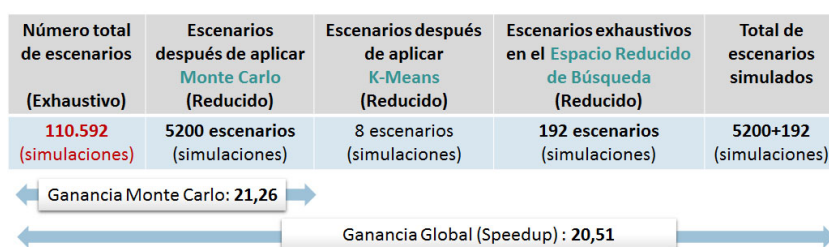
CASO A	Secuencial		16 cores		Ganancia 1000 cores	
	Simulaciones	minutos días	minutos días	speedup	minutos	
110592	243302	169	15896	11	15,306	240
5200	11440	8	756	0,5	15,132	11
8						
192	422	0,3	27	0	15,63	0,4

pecto a la memoria RAM disponible, 10 nodos cuentan con 10GB mientras que el resto dispone de 2GB. El sistema operativo instalado es GNU/Linux Debian 6.0 de 64 bits (versión del kernel 2.6.32).

4.2. Ganancia lograda con el uso HPC

El método propuesto permitió lograr una significativa ganancia en la eficiencia de la implementación, que se puede analizar en dos niveles de su desarrollo. En un primer lugar, el método de OvS permitió reducir del espacio de búsqueda y, en consecuencia, obtener una aceleración en cantidad de cómputo medida en simulaciones. En segundo lugar, la implementación del método aplicando técnicas de procesamiento en paralelo brindó su propia aceleración, que se suma a la primera.

La ganancia del MMC y la ganancia global están representadas en la Fig. 5.

**Figura 5.** Ganancia lograda con el método implementado

La tabla 4 compara los tiempos de ejecución del método, en el caso de la implementación de todo el espacio de búsqueda y en las dos etapas del método de OvS.

5. Conclusiones y líneas futuras

El método propuesto en este trabajo permite realizar una calibración automática de un simulador de cauces de ríos, brindando una técnica que mediante el uso exclusivo de cómputo y procesamiento paralelo y distribuido, mejora la

predicción del simulador. Las mejoras en la predicción del nivel del cauce en la ciudades del bajo Paraná entre 13-19 %. Estos valores se traducen en diferencias de altura entre 40 a 60 cm que pueden ser la diferencia entre la pérdida de bienes y de ganado, o el aviso temprano para evitarlo. La implementación en paralelo del método es fácilmente escalable, o sea cuando más recursos de cómputo se tengan más ganancia se obtiene, como se ve en la tabla 4 al utilizar 1000 cores. Pero el propósito de este trabajo es aportar una mejora a la predicción del simulador con la menor cantidad de recursos computacionales posible, objetivo que se alcanza con los dos niveles de speedup logrados. Una línea de trabajo futuro está centrada en la mejora del método de OvS para ajustar la predicción en los momentos que ocurre una crecida rápida del cauce del río, como también brindar períodos de validez del conjunto de parámetros óptimos encontrados por el método. El objetivo sería recalibrar automáticamente y predecir por períodos más extensos sin perder la calidad lograda en la simulación.

Agradecimientos

Esta investigación ha sido financiada por el MINECO España, bajo contrato TIN2014-53172-P y fue apoyado por el programa de investigación del Instituto de Investigación en Informática III-LIDI, Facultad de Informática, UNLP.

Referencias

1. S. Jonkman and J. Vrijling, "Loss of life due to floods," *Journal of Flood Risk Management*, vol. 1, no. 1, pp. 43–56, 2008.
2. E. Cabrera, E. Luque, M. Taboada, F. Epelde, and M. Iglesias, "Optimization of emergency departments by agent-based modeling and simulation," in *Information Reuse and Integration (IRI), 2012 IEEE 13th International Conference on*, 2012, pp. 423–430.
3. J. Warmink, J. Janssen, M. Booij, and M. Krol, "Identification and classification of uncertainties in the application of environmental models," *Environmental Modelling & Software*, vol. 25, no. 12, pp. 1518 – 1527, 2010.
4. A. Menéndez, "Three decades of development and application of numerical simulation tools at ina hydraulics lab." *Mecánica Computacional*, vol. XXI, pp. 2247–2266, 2002.
5. G. J. Arcement and V. R. Schneider, *Guide for selecting Manning's roughness coefficients for natural channels and flood plains*, ser. USGS Water-Supply Paper 2339. US Government Printing Office Washington, DC, USA, 1989.
6. L.-F. Wang and L.-Y. Shi, "Simulation optimization: A review on theory and applications," *Acta Automatica Sinica*, vol. 39, no. 11, pp. 1957 – 1968, 2013.
7. J. F. Santucci and L. Capocchi, "Optimization via simulation of catchment basin management using a discrete-event approach," *Simulation*, vol. 91, no. 1, pp. 43–58, Jan. 2015, iSSN:0037-5497.
8. D. P. Kroese, T. Taimre, and Z. I. Botev, *Handbook of Monte Carlo Methods*. John Wiley & Sons, 2013, vol. 706.
9. A. Gaudiani, E. Luque, P. García, M. Re, M. Naiouf, and A. De Giusti, "Computing, a powerful tool for improving the parameters simulation quality in flood prediction," *Procedia Computer Science*, vol. 29, no. 0, pp. 299 – 309, 2014.

Cloud Robotics: Sistema Multi-Robot conectado al Cloud público AWS

Rodriguez Ismael P.¹, Paniego Juan Manuel¹, Rodriguez Eguren Sebastian¹, Estrebou Cesar¹, De Giusti Armando E.^{1,2}

¹Instituto de Investigación en Informática LIDI (III-LIDI),
Facultad de Informática, Universidad Nacional de La Plata, Argentina
²CONICET, Ministerio de Ciencia, Tecnología e Innovación Productiva, Argentina.
{ismael, jmpaniego, seguren, cesarest, degiusti}@lidi.info.unlp.edu.ar

Abstract.

Este trabajo presenta el despliegue de un escenario de sistema multi-robot, integrado por robots conectados vía WiFi al Cloud público de Amazon Web Services, simulando un sistema de riego artificial de campo. Se detallan los prototipos desarrollados, como así también el protocolo de comunicación utilizado con el servicio de AWS IoT, y los algoritmos implementados que permiten procesar las imágenes adquiridas e identificar determinados patrones y objetos.

Keywords: *Cloud Robotics, Cloud Computing, Internet of Things, AWS, MQTT.*

1 Introducción

Los avances en el paradigma de Cloud Computing han provocado un factor disruptivo de las TI en la industria tecnológica. Según el Instituto Nacional de Estándares y Tecnologías del Departamento de Comercio de los EEUU (NIST), como en varias publicaciones de diversos autores, se ha definido a Cloud Computing como: “un paradigma informático de cómputo distribuido, que proporciona grandes conjuntos de recursos virtuales (como ser hardware, plataformas de desarrollo, almacenamiento y/o aplicaciones), fácilmente accesibles y utilizables por medio de una interfaz de administración web. Estos recursos son proporcionados como servicios (del inglés, “as a service”) y pueden ser dinámicamente reconfigurados para adaptarse a una carga de trabajo variable (escalabilidad), logrando una mejor utilización y evitando el sobre o sub dimensionamiento (elasticidad). El acceso a los recursos se realiza bajo la demanda de los usuarios, en base a un modelo de autoservicio” [1].

El modelo de Cloud Computing presenta las características y beneficios siguientes: recursos disponibles bajo demanda, escalabilidad y elasticidad, aprovisionamiento automático de recursos y autoservicio.

Dicho paradigma, brinda al menos tres modelos de despliegue: Cloud Público, Cloud Privado y Cloud Híbrido [2] [3].

Por otro lado, considerando la gran influencia de la robótica en la sociedad y la diversidad de servicios ofrecidos por robots, nos encontramos que los mismos presentan grandes limitaciones en consumo de energía, poder de cómputo, capacidad de almacenamiento, toma de decisiones, tareas cognitivas, etc.

En el año 2010, comenzaron a surgir proyectos de investigación (ej: RoboEarth [4]), que integraban las tecnologías de Cloud con los sistemas de robots. Es así, que James Kuffner propone el concepto de Cloud Robotics, basado en combinar las tecnologías de robots con el paradigma de Cloud Computing [5].

La idea de Cloud Robotics, permite por medio de aplicaciones tratar los datos de los componentes de hardware del robot (sensores, actuadores, cámaras, microcontroladores, memoria, etc.), sin importar las limitaciones de cómputo de las placas de desarrollo con microcontroladores y la capacidad de almacenamiento de las mismas [6]. En otras palabras, este concepto permite a los robots obtener resultados de tareas de cómputo intensivo, tales como: procesamiento de imágenes, reconocimiento de voz, determinación de rutas, confección de mapas, acciones cognitivas, etc., sin tratamiento local, sino en el Cloud.

Este paradigma brinda la capacidad de establecer escenarios para sistemas de multi-robot, donde cada robot se integra de un hardware mínimo; una placa microcontroladora con conectividad WiFi permite al robot comunicarse con el Cloud y otros robots. Los datos de sensores y/o sistemas de adquisición de imágenes se procesan en el Cloud y los actuadores de cada robot llevarán a cabo las operaciones necesarias [7].

El propósito de investigación del presente trabajo es el despliegue de un escenario de multi-robot, inicialmente con dos robots conectados al Cloud Público de Amazon Web Services con el fin de simular un sistema de riego artificial de campo. Para alcanzar tales objetivos, se determinó el método de comunicación con el Cloud; se desarrollaron algoritmos que permitan procesar imágenes adquiridas por medio de una cámara de video, identificando determinados patrones y objetos, como así también, se dotó a uno de los robots, de un sistema autónomo con capacidad de orientarse, trasladarse y detenerse.

La estructura de este trabajo es la siguiente: la Sección 2 introduce algunos conceptos elementales. A continuación, la Sección 3 presenta los prototipos confeccionados y sus componentes. La Sección 4 describe el trabajo experimental realizado y los resultados obtenidos. Por último, la Sección 5 expone las líneas de trabajo futuro en relación a esta investigación.

2 Algunos conceptos elementales

En esta Sección, mencionaremos algunos conceptos empleados en la investigación realizada.

- **Amazon Web Services**

Es una plataforma de Cloud Público, que provee “Infraestructura como servicio”; a través de la tecnología de virtualización, provee un gran conjunto de recursos de cómputo, como ser, almacenamiento y capacidad de procesamiento, que pueden ser

solicitados a demanda, como así también, se adecuan dinámicamente en tamaño conforme la necesidad del consumidor. Los servicios más conocidos de AWS son “Elastic Compute Cloud” (EC2), “Simple Storage Service” (S3) y en nuestro caso utilizaremos los servicios de “AWS Internet of Things” (AWS IoT) [8].

- **Internet of Things (IoT)**

Es un nuevo paradigma cuya definición deriva de considerar “objetos” o “cosas” conectadas a Internet por medio de redes Wireless, sean estas de tecnología WiFi o 4G LTE, utilizando protocolos de comunicación estándar. Dichas “cosas” pueden considerarse etiquetas de identificación por radio frecuencia (RFID), sensores, actuadores, teléfonos móviles, etc. [9].

- **AWS IoT**

Es un servicio que proporciona AWS con el fin de conectar, administrar y operar grandes conjuntos de dispositivos o cosas. Ofrece mecanismos de conectividad y seguridad para la transmisión de datos. También, permite que los datos, una vez enviados a la plataforma, puedan ser procesados por las aplicaciones de análisis masivo de datos (Elastic MapReduce), análisis predictivo para aprendizaje automático (Amazon Machine Learning), almacenamiento en bases de datos, etc. [10].

AWS IoT permite conectar fácilmente dispositivos al Cloud y a otros dispositivos. El servicio es compatible con HTTP, WebSockets y MQTT. Este último se ha convertido en el protocolo por excelencia. AWS IoT utiliza MQTT v.3.1.1 con QoS 0 y 1.

Cada dispositivo debe ser registrado como una “cosa” en AWS IoT, para lo cual se emitirá un certificado y un par de llaves privada-pública para el mismo. El certificado y la llave privada, junto con el certificado de la CA de Amazon deberán ser instalados en el dispositivo con el fin que este pueda conectarse al servicio de AWS IoT.

- **MQTT**

Message Queuing Telemetry Transport [11] es un protocolo de comunicación ligero, especialmente diseñado para tolerar conexiones intermitentes y reducir los requisitos de ancho de banda de la red. Desde finales del año 2014, fue presentado como un estándar abierto OASIS [12]. Soporta comunicación segura con TLS. Maneja tres niveles de calidad de servicio: QoS 0: A lo sumo una vez la entrega del mensaje. QoS 1: Al menos una vez la entrega del mensaje. QoS 2: Exactamente una vez la entrega del mensaje.

Este protocolo, implementa la comunicación de mensajes por medio de la publicación/suscripción sobre un tópico específico (canal de comunicación), como se puede observar en la figura siguiente:



Figura 1: MQTT: publicación/suscripción.

3 Prototipos y componentes

Con el fin de simular un escenario de riego artificial de campo, sectorizado en 4 zonas, se han implementado dos robots que interactúan con el Cloud vía conexión WiFi; de aquí en adelante se denomina al primer robot, como “Rover 4WD” y al segundo, “Gestor de Sensores de HS”. También, se ha desplegado una instancia de AWS EC2, para ejecutar los algoritmos de procesamiento de imágenes y toma de decisiones, la cual denominaremos aplicación “Master”.

3.1 Robot Rover 4WD

El robot Rover 4WD, emula un transporte cisterna, que recibe, a través del Cloud, la zona de riego a la que debe dirigirse. Para la confección de tal robot, se han utilizado los componentes siguientes:

- Chasis Rover de 4 ruedas
Consiste en dos placas de acrílico, con cuatro motores DC de 3v, cuatro ruedas y un codificador de velocidad.



Figura 2: Chasis Rover de 4 ruedas.

- Placa de desarrollo
Placa Raspberry Pi 3 con conexión WiFi, almacenamiento en memoria MicroSD de 8Gb y alimentación de energía de 5v.



Figura 3: Placa Raspberry Pi 3.

- Cámara de video
“Pi Camera Board”, de 5MP y calidad 1080p. Se integra a la placa Raspberry Pi 3.



Figura 4: Cámara de Video para Raspberry Pi.

- Puente H
Módulo L298 para controlar la dirección de los motores y por medio de PWM la velocidad de los mismos.



Figura 5: Puente H Módulo L298.

- Sensor Giroscopio y acelerómetro
El sistema MicroElectroMecánico MPU-6050, dispone de un sensor giroscopio de 3 ejes y un sensor acelerómetro de 3 ejes, integrados juntos por un procesador digital de movimiento (DMP); este módulo permite controlar el ángulo de giro del robot.



Figura 6: Sensor Giroscopio MPU-6050.

- Decodificador óptico
Módulo Optoacoplador de tipo “herradura” para codificar la rotación de la rueda.



Figura 7: Módulo Optoacoplador tipo “herradura”.

- Prototipo Rover 4WD ensamblado



Figura 8: Robot Rover 4WD.

3.2 Robot Gestor de Sensores HS

El robot Gestor de Sensores de Humedad de Suelo, adquiere datos de nivel de humedad de suelo, provenientes de los sensores. Para la confección de tal robot, se han utilizado los componentes siguientes:

- Placa de desarrollo

Placa Raspberry Pi 3 con conexión WiFi, almacenamiento en memoria MicroSD de 8Gb y alimentación de energía de 5v.



Figura 9: Placa Raspberry Pi 3.

- Sensores de Humedad de Suelo

Consisten en 4 sensores de humedad de suelo, módulos YL38 y YL69.



Figura 10: Sensor de Humedad de Suelo YL38 + YL69.

- Conversor ADC

Microchip MCP 3008, que convierte datos analógicos a digital. Posee 8 canales analógicos, de los cuales 4 se utilizan la conectar los sensores de humedad de suelo.



Figura 11: Conversor ADC MCP 3008.

- **Marcadores gráficos**

Consisten en 4 marcadores gráficos que identifican cada una de las zonas de riego, con un sensor de humedad de suelo en cada una de ellas.



Figura 12: Marcadores Gráficos.

3.3 Instancia AWS EC2 – Aplicación “Master”

Consiste en el despliegue de una instancia de AWS EC2 “t2.large”, con 2 VCPU y 8GB de RAM; Sistema Operativo Linux distribución Ubuntu Server 14.04 LTS [13] [14]. Dicha instancia ejecuta los algoritmos de procesamiento de imágenes y la toma de decisiones sobre los robots Rover 4WD y Gestor de Sensores HS.

3.4 Escenario sistema de riego artificial de campo

Se puede observar, el sistema de riego de campo simulado, en la figura siguiente:



Figura 14: Escenario sistemas de riego artificial de campo.

4 Trabajo experimental

El trabajo de investigación consistió en desarrollar un escenario que simule un sistema de riego artificial de campo, emulando un robot cisterna de transporte y un gestor de sensores de humedad de suelo. Se puede observar la conexión de los componentes de los robots Rover 4WD y Gestor de Sensores HS, en las figuras 15 y 16, respectivamente.

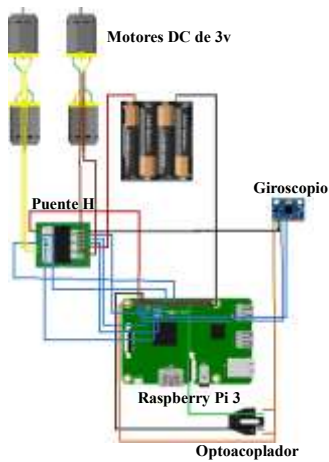


Figura 15: Rover 4WD.

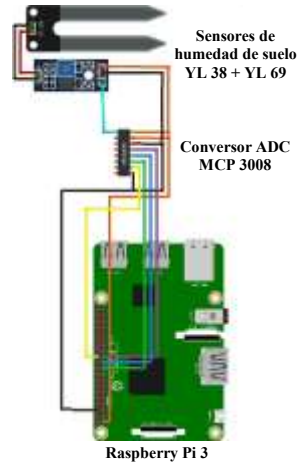


Figura 16: Gestor de Sensores HS.

Ambos robots, se conectan a internet vía tecnología WiFi, al Cloud Público de Amazon Web Services. Para tal fin se dispone de una cuenta en el proveedor de Cloud, y se utilizan los servicios AWS EC2 y AWS IoT.

Sobre la plataforma de AWS se despliega una instancia “t2.large”, con S.O. Linux Ubuntu Server 14.04 LTS, con Python v.3.4.2 y la librería “Paho Python Client” que dispone de una clase cliente que soporta MQTT v.3.1.1 sobre Python v.3.x. En esta instancia se instala el certificado de dispositivo de AWS IoT, el certificado de la CA de Amazon y la clave privada del dispositivo (la instancia es considerada un dispositivo o “cosa” conectada al servicio AWS IoT). Además, cuenta con la aplicación “Master”, escrita en lenguaje Python y tiene como objetivo procesar imágenes y tomar decisiones sobre los dispositivos robot Rover 4WD y Gestor de Sensores HS. En la figura 17 se puede observar el sistema multi-robot conectados a AWS IoT.

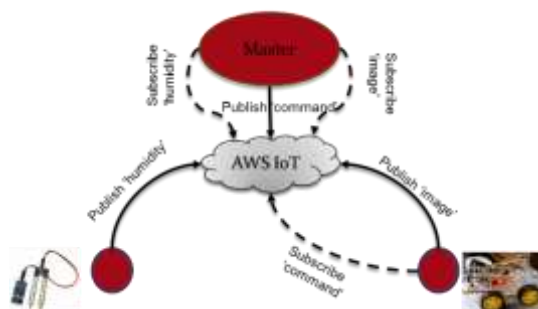


Figura 17: Cloud Robotics: Sistema multi-robot conectado a AWS IoT.

Los dispositivos se conectan a Internet y utilizan el protocolo de comunicación MQTT para interconectar los robots con el Cloud. Una vez efectuada negociación de

autenticación de seguridad, se establecen los canales de comunicación a utilizar (tópicos).

A continuación, sobre el robot Rover 4WD, se realiza una tarea de calibración al sensor giroscopio, como también, se ingresan los umbrales de nivel de humedad de suelo aceptables, en el robot Gestor de Sensores HS.

En esta instancia, los robots se encuentran preparados para comenzar a operar.

El robot de Sensores HS comienza a adquirir los niveles de humedad de suelo de cada una las zonas de riego, provistas de un sensor y un marcador gráfico que las distingue. Cada valor adquirido se compara con el valor del umbral, de la zona correspondiente, ingresado por el usuario al inicio. Esto permitirá, que solo aquellos que se encuentren dentro del umbral, sean enviados al Cloud.

Posteriormente, la aplicación “Master”, utilizando el protocolo MQTT (Figura 18), se suscribe al tópico “humidity” y recibe los valores de humedad, los cuales son procesados, asignándoles una prioridad, la cual servirá para decidir qué zona necesita de riego con mayor antelación.

El robot Rover 4WD, se suscribe al tópico “command”, por el cual recibe del Cloud el comando para iniciar su trabajo. En este instante, comienza la comunicación entre ambos. Principalmente, se realizan capturas de imágenes para ubicar la zona a ‘regar’. Esta captura, es realizada por el Rover y enviada al Cloud, publicando la misma en el tópico “image”, para iniciar la detección de la zona a través de los marcadores.

En el Cloud, la aplicación “Master” se suscribe al tópico “image” y por medio de JSON, regenera la imagen recibida del robot. Esta es procesada por el algoritmo de procesamiento de imágenes, utilizando la biblioteca OpenCV (Open Source Computer Vision) [15], por medio de la cual intenta identificar un cuadrilátero con contorno de color negro, para luego examinar su interior y obtener el ID del marcador de la imagen. En caso de no identificar el marcador, la aplicación “Master” publicará en el tópico “command” los comandos para girar a derecha o izquierda una cierta cantidad de grados; posteriormente, se publica sobre el mismo tópico, el comando para capturar una nueva imagen.

Este ciclo se repite hasta detectar en la imagen un marcador gráfico. Una vez detectado, se obtiene la distancia de separación del robot al marcador, y se publica en el tópico “command” el comando para avanzar una cierta distancia hasta detenerse a 20 cm del marcador. Ya posicionado frente al marcador, publica en el tópico “finish” para que el robot emita un sonido de finalización y realice un movimiento en reversa hasta su posición inicial, quedando disponible para realizar la siguiente búsqueda que especifique el Cloud.

Tópico	Mensaje	Descripción
humidity	Arreglo con el valor analógico de humedad y la 'zona' a la que hace referencia. Ej: [m':0, 'h':600]	Se envía desde el robot de sensor de HS. Se envía al detectar uno de los valores de humedad dentro del umbral definido.
command	Avanzar una distancia hacia adelante ('a x') o girar sobre el eje una determinada cantidad de grados ('[d] x'). Así como también el pedido de captura de imagen.	Los mensajes de comandos, se envían desde el Cloud al Rover 4WD. Según el contenido del 'payload' o 'mensaje', el rover actúa de manera distinta.
image	Imagen capturada, en base64	Se utiliza para transmitir las imágenes capturadas. Las mismas se pasan a base64 para poder ser transmitidas desde el rover al cloud.
finish	-	Su uso es solo para advertir desde el rover la finalización de las tareas al alcanzar el objetivo.

Figura 18: Tópicos de mensajes de MQTT.

5 Conclusiones y Líneas de Trabajo Futuro

En esta investigación se ha implementado un sistema multi-robot conectado al Cloud. Se han desarrollado algoritmos para el procesamiento de imágenes, utilizando las librerías de OpenCV, y toma de decisiones, para indicar las acciones que debe tomar cada robot. Podemos concluir que Cloud Robotics es una tecnología que favorecerá el avance de los sistemas multi-robot reduciendo las limitaciones actuales.

Las líneas de trabajo futuro relacionadas con este trabajo incluyen la integración de una cámara aérea, que permita procesar una imagen y determinar la ubicación del robot Rover 4WD y del Marcador de la zona de riego, para dirigir el robot hasta dicha zona. Además, es deseable incorporar cierto nivel de “inteligencia” para eludir obstáculos.

Por otro lado, adicionar uno o más robots Rover 4WD, adaptando la solución para atender la prioridad de la zona de riego por cercanía, estado de cisterna, etc. También es deseable, incorporar trabajos colaborativos de mayor complejidad a ser efectuados por un conjunto de robots.

Referencias

1. Mell P., Grance T.: “The NIST Definition of Cloud Computing”. In: Publicación Especial 800-145. Septiembre, 2011.
2. Pettoruti J., Rodriguez I., Chichizola F., De Giusti A.: “Análisis de la degradación de las comunicaciones en algoritmos de cómputo científico en un Cloud privado”. In: XII Workshop de Procesamiento Distribuido y Paralelo (WPDP) – XVIII Congreso Argentino de Ciencias de la Computación (CACIC2012). Bahía Blanca, Argentina, 2012.
3. Rodriguez, I., Pettoruti, J.E., Chichizola, F., De Giusti, A.: Despliegue de un Cloud Privado para entornos de cómputo científico. In: Proceedings del XI Workshop de Procesamiento Distribuido y Paralelo (WPDP) - XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011). La Plata, Argentina. (2011).
4. RoboEarth. In: <http://www.roboearth.org>. Junio, 2016.
5. Kuffner J.: “Cloud-enabled robots”. In: IEEE-RAS International Conference on Humanoid Robot. Nashville, USA, 2010.
6. Wang L., Liu M., Meng M., Siegart R.: “Towards Real-Time Multi-Sensor Information Retrieval in Cloud Robotic System”. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI). Hamburgo, Alemania, 2012.
7. Turnbull L., et al.: “Cloud Robotics: Formation Control of a Multi Robot System Utilizing Cloud Infrastructure”. In: Proceedings of IEEE – Southeastcon. Jacksonville, USA, 2013.
8. Bermudez I., Traverso S., Mellia M., Munafò M.: “Exploring the cloud from passive measurements: The Amazon AWS case”. In: Proceedings of IEEE – INFOCOM, 2013.
9. Atzori L., Iera A., Morabito G.: “The Internet of Things: A survey”. In: ELSEVIER Journal Computer Networks. Volume 54, Issue 15. 2010.
10. Informe técnico: AWS Whitepapers. “Core Tenets of IoT”. Abril 2016.
11. Protocolo MQTT: <http://mqtt.org>. Julio 2016.
12. OASIS MQTT: <https://www.oasis-open.org>. Julio 2016.
13. AWS Instance Types: <https://aws.amazon.com/es/ec2/instance-types>. Julio 2016.
14. AWS Marketplace: <https://aws.amazon.com/marketplace/pp/B00JV9JBDS>. Julio 2016.
15. OpenCV: <http://opencv.org>. Julio 2016

Análisis de la migración del cómputo intensivo a un cloud privado con OpenStack

María A. Murazzo^{1*}, Federico Castiglione^{2*}, Diego Medel^{3*}, Fernando G. Tinetti^{4#},
Nelson R. Rodríguez^{5*}

¹marite@unsj-cuim.edu.ar, ²feder102@gmail.com, ³mdiego88@gmail.com,
⁴fernando@lidi.info.unlp.edu.ar, ⁵nelson@iinfo.unsj.edu.ar

*Departamento e Instituto de Informática – UNSJ – Cereceto y Meglioli – C.P. 5400 –
Rivadavia, San Juan, Argentina

#III-LIDI, Facultad de Informática, UNLP - Comisión de Inv. Científicas Prov. de Bs. As.
50 y 120, 2do. Piso, 1900, La Plata, Argentina

Resumen. Cloud Computing en un modelo de computación que permite que los recursos computacionales, tales como infraestructura, aplicaciones, software o procesamiento puedan ser ofrecidos y consumido bajo demanda como un servicio más en Internet. Esta capacidad se logra mediante la abstracción de los recursos físicos, generando un pool de recursos virtualizados, que pueden ser aprovisionados dinámicamente. Es por ello se están migrando datos y aplicaciones al cloud. Sin embargo, uno de los aspectos que se debe tener en cuenta a la hora de realizar la migración es el costo en términos de degradación de performance. En este sentido se debe analizar cómo impacta en el cloud el proceso de la virtualización de los recursos, por ello el objetivo de este trabajo es analizar cuánto se degrada la performance de algoritmos cuando se corren en el cloud.

Keywords: Cloud Computing, IaaS, OpenStack, Virtualización, OpenMP

1 Introducción

Los sistemas de cómputo convencional han evolucionado a sistemas de procesamiento de altas prestaciones (HPC) para realizar operaciones de cómputo intensivo y mejorar la velocidad de procesamiento, frente a la creciente demanda computacional y complejidad computacional involucrados.

En este aspecto el desafío se centra en cómo se aprovecha al máximo el potencial de la arquitectura física para la ejecución de algoritmos de cómputo intensivo. Con el objetivo de mejorar los tiempos de procesamiento en los algoritmos, se puede optar por la implementación en plataformas con mayor potencia de cálculo, pero los costos de estos equipos son elevados, lo que dificulta su acceso a una gran cantidad de comunidades científicas.

Para resolver los problemas de costo, la computación distribuida es un modelo destinado a resolver problemas de cómputo masivo utilizando un gran número de computadoras organizadas sobre una infraestructura de comunicaciones distribuida. De esta manera es posible compartir recursos heterogéneos, basados en distintas plataformas, arquitecturas y lenguajes de programación, situados en distintos lugares y pertenecientes a diferentes dominios de administración sobre una red que utiliza estándares abiertos [1].

Una alternativa para lograr esto es el cloud, el cual permite contar con una cantidad de recursos computacionales virtualmente infinitos, administrados por terceros y accedidos bajo demanda pagando por el uso, de esta manera es posible lograr calidad de servicio (QoS) garantizado por parte del proveedor de la infraestructura cloud.

Cloud Computing se ha convertido en una tecnología centrada en ofrecer cómputo bajo demanda como cualquier otro servicio. Uno de los aspectos clave que afecta la performance de las aplicaciones al montarlas sobre arquitectura cloud, es la capacidad de virtualización de cualquier recurso (bases de datos, red, procesador, etc.) y ofrecerlo como un servicio (AaaS, Anything as a Service) [2].

Esto puede tener un impacto negativo en la ejecución de aplicaciones de cálculo intensivo, debido a que los ambientes cloud deben virtualizar la plataforma en la cual se ejecuta y esto conlleva un overhead de contextualización de las máquinas virtuales.

En función de esto, el objetivo de este trabajo es evaluar cómo se degrada la performance de una aplicación de cómputo intensivo al migrarla a una infraestructura cloud y determinar en qué punto esa migración no es beneficiosa.

2 Fundamentos del Cloud Computing

Cloud Computing, es un modelo para permitir el acceso en red omnipresente, conveniente y bajo demanda a un conjunto compartido de recursos informáticos configurables tales como, redes, servidores, almacenamiento, aplicaciones y servicios que pueden ser rápidamente aprovisionados y liberados con un mínimo de gestión esfuerzo e interacción con el proveedor o administrador de servicio [3].

Según [4] cloud es un modelo de prestación de servicios informáticos cuya principal orientación es la escalabilidad. Esto es, que desde el punto de vista de los usuarios, los servicios son elásticos, o sea, pueden crecer o recuperar su tamaño original de manera rápida y sencilla. Esta orientación permite que los usuarios que acceden a los servicios, perciban que todo funciona de manera simple y rápida, dando como resultado una experiencia más gratificante.

El paradigma cloud es atractivo para las aplicaciones de cómputo intensivo debido a que ofrece disponibilidad inmediata de recursos, un stack de software que puede seleccionar el dueño de la aplicación, elasticidad de los recursos, abstracción del hardware mediante la virtualización, lo cual provee portabilidad, entre las más importantes. Estos beneficios hacen que usar cloud como plataforma para los algoritmos de cómputo intensivo sea una alternativa atractiva desde el punto de vista de la performance y la eficiencia [5].

3 Migración al Cloud Computing

El principal objetivo que se persigue al instalar una infraestructura cloud, es permitir a los usuarios la migración de sus datos y aplicaciones sin que se genere un impacto en la performance. De esta manera se logra reducir los costos de mantenimiento y abastecimiento, al tiempo que se aumenta la escalabilidad y la disponibilidad.

Una de las características más relevantes del Cloud Computing es su elasticidad, es decir, su capacidad de ajustarse a la demanda de los usuarios. Ante un pico elevado de requisitos de cómputo, un usuario puede solicitar mayor capacidad de cálculo, que provocará el despliegue automático de nuevas instancias sobre la infraestructura física

existente. Por el contrario, si lo que se detecta es una baja de la demanda de cómputo, el usuario puede decidir prescindir de parte de su infraestructura, pudiéndola ajustar a su uso de forma dinámica. Esta es una gran ventaja, puesto que uno de los problemas que existían hasta la aparición del Cloud Computing era la creación de una infraestructura ad hoc preparada para soportar demandas no previsibles de cómputo, ya que esto suponía una elevada inversión y además no se podía asegurar que fuese capaz de responder correctamente a estos incrementos repentinos en la demanda de los recursos de cómputo [6].

Desde el punto de vista de la arquitectura de Cloud Computing la capa más involucrada en la migración hacia el cloud es el IaaS, pues es la que dará el soporte de hardware (físico o virtualizado) para concretar la migración. El IaaS brinda la capacidad de cómputo, procesamiento y almacenamiento para que el usuario pueda desplegar de forma transparente sistemas operativos y aplicaciones. Para lograr esta transparencia, la infraestructura se monta sobre una capa de virtualización que permite contar con un ambiente optimizado independientemente del hardware físico mediante la generación de instancias abstractas de recursos llamadas máquinas virtuales, las cuales se encuentran administradas por los hipervisores.

Los productos disponibles en el mercado para instalar infraestructuras cloud cuentan con diferentes hipervisores, los cuales se encargan de orquestar los recursos virtuales de cada máquina virtual con el objeto de sacar el máximo beneficio a los recursos físicos. Para lograr esto, es necesario *maximizar la escalabilidad* de la infraestructura en la medida que exista un aumento en la demanda de recursos virtualizados. La escalabilidad de una infraestructura cloud está fuertemente ligada con la tecnología de virtualización o hipervisores que se está usando. Por esta razón es tan importante la elección adecuada del hipervisor. Hay que destacar que si bien la virtualización aísla al usuario / desarrollador de los detalles de orquestación de recursos, un aspecto que se debe analizar es el overhead producido por los hipervisores. Estos, consumen recursos para la administración de los recursos virtualizados, lo que lleva a que mientras más se virtualice, mas overhead haya y por consiguiente más sobrecarga y menos rendimiento [7].

4 Infraestructura Cloud Propuesta

Con el objetivo de medir el desempeño de la capa de virtualización, se ha instalado un cloud privado open source. Para ello, el mercado ofrece muchos productos de los cuales se ha seleccionado OpenStack, debido a que es el proyecto que mayor aceptación y mayores contribuciones ha tenido en los últimos años [8]. Además, es el producto con mayor número de hipervisores soportados, lo que lo hace adecuado para poder realizar una evaluación más amplia de su desempeño.

OpenStack [9] es un proyecto global para crear una plataforma de cómputo open source en el cloud, que cumpla con las necesidades de los proveedores de servicios en clouds públicos, privados e híbridos, independientemente de su tamaño y cuyas premisas son que sea fácil de implementar y escalable.

La arquitectura de OpenStack es totalmente modular y en la actualidad engloba dos proyectos principales: *OpenStack Compute (NOVA aportado por la NASA)* y *OpenStack Object Storage (SWIFT, aportado por Rackspace)*. Con base en estos dos proyectos, OpenStack, posee tres módulos principales: Compute, Networking y Storage, a partir de los cuales se han desarrollado un conjunto de componentes que

permiten la administración y gestión del cloud, de los cuales los principales son: Nova, Glance, Swift, Horizon, Keystone, Neutron. Además, dependiendo de la versión de la que se trate se cuentan con otros componentes [10], los cuales se comunican mediante APIs provistas por el componente.

En este trabajo, se ha trabajado con OpenStack Liberty [11], la cual es la décima segunda versión liberada en octubre de 2015. A partir de esta versión, el cambio más importante radica en la nueva política para el tratamiento de los proyectos en incubación, adoptando el modo *Big Tent*. Esto significa que los proyectos incubados, podrán integrar OpenStack siempre que respeten el flujo de trabajo y las convenciones de la comunidad. Así, OpenStack no se limitará al IaaS, sino que podrá integrar funcionalidades de PaaS, o evolucionar hacia otras versiones de más alto nivel en las capas de software.

Con el objeto de evaluar la degradación del desempeño causada por la virtualización, se pone énfasis en el componente *OpenStack Compute*, el cual proporciona una herramienta para orquestar un cloud, incluyendo la ejecución de instancias, administración de redes y control de acceso a usuarios y proyectos. El nombre del proyecto es *Nova* y proporciona el software que controla la infraestructura cloud. *OpenStack Compute* no incluye ningún software de virtualización, en lugar de eso, define drivers que interactúan con los hipervisores que se encargan de la gestión del ciclo de vida de las instancias mediante la gestión de las máquinas virtuales. De los servicios de este componentes, el de mayor interés es *nova-compute service* que es un demonio para administrar las instancias de las máquinas virtuales mediante la API del hipervisor. En *OpenStack Liberty*, hay soporte para QEMU, KVM y XEN, por defecto se instala KVM, pero es posible cambiar el hipervisor mediante la modificación del archivo *nova.conf*.

En este trabajo se ha decidido trabajar con KVM, debido a que al ser un hipervisor que no necesita libvirt para que se ejecute, permite acelerar la virtualización. Además, es un hipervisor nativo, lo cual es ventajoso pues la contextualización de las máquinas virtuales demanda menos tiempo.

La topología con la cual se ha instalado OpenStack Liberty es pseudo distribuida, en la cual se instala Nova en un nodo de forma apropiativa, dejando el resto de los componentes para ser instalados en otro nodo con su configuración básica.

5 Definición del Problema

El objetivo que se persigue es evaluar cómo impacta la virtualización de una infraestructura cloud cuando se realiza cómputo intensivo. Para ello, se analiza la degradación del desempeño que sufre los algoritmos programados bajo el modelo de memoria compartida usando OpenMP.

Se tomó como caso de estudio un algoritmo de álgebra lineal, más precisamente la multiplicación de matrices cuadradas. La multiplicación de matrices es una operación fundamental en múltiples campos de aplicación científica, esto hace que sea el caso más estudiados en la computación de altas prestaciones y se utiliza como benchmark de bajo nivel.

Para realizar la evaluación se definieron cuatro escenarios de trabajo, uno de ellos es la ejecución nativa sobre una PC y los otros tres son instancias virtualizadas con tres sabores diferentes. Las especificaciones de los escenarios se muestran en la tabla 1. En los cuatro escenarios se ha instalado Ubuntu 14.04 LTS. En los escenarios cloud

las instancias se levantan con cada sabor independientes entre si, esto es, solo una instancia con un sabor para realizar la ejecución del algoritmo.

		Cores	RAM	Disco
Nativo		4	16 GB	1Tb
Cloud	Sabor 1	4	4 Gb	10 Gb
	Sabor 2	6	5 Gb	11 Gb
	Sabor 3	8	6 Gb	12 Gb

Tabla 1: Especificaciones de los escenarios

6 Análisis de Resultados

En cada escenario se corre el algoritmo de manera secuencial y paralela (usando OpenMP) y se trabaja con tres grados de las matrices: 800, 1000 y 2000. Los tiempos de ejecución del algoritmo para los cuatro escenarios y los tres grados de las matrices se muestran en la tabla 2.

	800		1000		2000	
	Secuencial	OpenMP	Secuencial	OpenMP	Secuencial	OpenMP
Nativo	2,199702	0,827707	4,248192	1,597	42,243984	19,123565
Sabor 1	2,884403	1,132979	5,027938	2,405722	44,380807	28,564535
Sabor 2	2,290272	1,100913	4,478949	2,144922	43,834091	28,682011
Sabor 3	2,884706	1,188093	5,035481	2,002979	82,411872	27,060603

Tabla 2: Tiempos de ejecución (en segundos)

Como era de esperar, en todos los casos la ejecución del algoritmo sobre el escenario nativo arroja mejores resultados. También era de esperar que en todos los casos, OpenMP brinde tiempos de ejecución más rápidos que la opción secuencial. En la tabla 3, se puede ver en la columna *Mejora*, cuanto más rápida es la ejecución del algoritmo en OpenMP con relación al algoritmo secuencial, para todos los escenarios y todos los grados de las matrices; mientras que en la columna *Porcentaje*, se expresa el valor de esta mejora.

	800		1000		2000	
	Mejora	Porcentaje	Mejora	Porcentaje	Mejora	Porcentaje
Nativo	1,371995	62,37%	2,651192	62,41%	23,120419	54,73%
Sabor 1	1,751424	60,72%	2,622216	52,15%	15,816272	35,64%
Sabor 2	1,189359	51,93%	2,334027	52,11%	15,15208	34,57%
Sabor 3	1,696613	58,81%	3,032502	60,22%	55,351269	67,16%
δ		0,04588643		0,053764192		0,157675023

Tabla 3: Mejora en la ejecución de OpenMP (en segundos)

Un dato interesante de analizar es el valor δ , que representa la desviación estándar, como se puede ver este valor para matrices de grado 800 y 1000 indica que la variabilidad de los tiempos de ejecución se mantiene más o menos cercanos. Sin embargo para matrices de grado 2000, si bien el valor de δ es bajo, se ha incrementado en 0,1078 con respecto a la media de los δ de las matrices de grado 800 y 1000. Este aumento en la desviación estándar se debe a que para el Sabor 3, la

multiplicación de matrices de grado 2000, de forma secuencial demora el doble que en los otros escenarios, sin embargo la ejecución con OpenMP se mantiene estable con los otros escenarios.

Con el objeto de analizar en detalle los tiempos de ejecución en función del cómputo realizado, se analizaran para cada grado de las matrices, como se comportan los diferentes escenarios. Para todos los casos, se ha calculado δ y δ^2 para los escenarios cloud, sin incluir el escenario nativo, debido a que incorpora información que no es relevante para analizar el cómputo en ambientes virtualizados.

En la figura 1, se puede ver gráficamente el comportamiento de los tiempos de ejecución del algoritmo para matrices de grado 800.

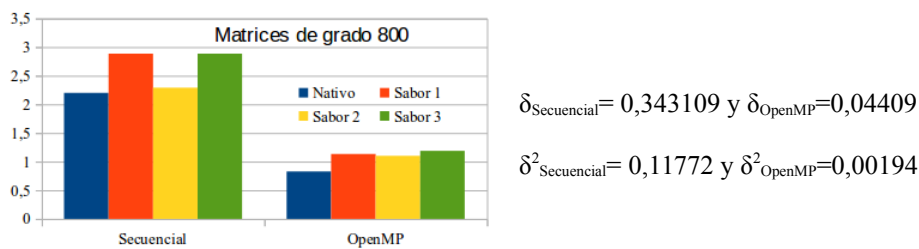


Figura 1: Tiempos de Ejecución para Matrices de grado 800

Como se puede ver, a pesar de que la ejecución en los escenarios cloud generan una degradación de la performance, el comportamiento de los tiempos es más estable cuando se ejecuta el algoritmo con OpenMP.

En la figura 2, se puede ver gráficamente el comportamiento de los tiempos de ejecución del algoritmo para matrices de grado 1000. Para este caso, a medida que el escenario le da más recursos, los tiempos de ejecución mejoran, es por ello que el valor de δ aumenta en relación al caso de las matrices de grado 800.

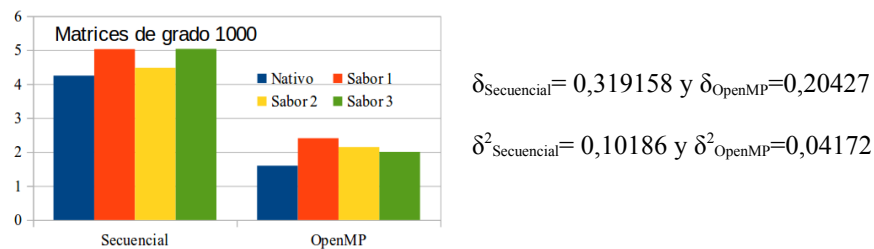


Figura 2: Tiempos de Ejecución para Matrices de grado 1000

En la figura 3, se puede ver gráficamente el comportamiento de los tiempos de ejecución del algoritmo para matrices de grado 2000. En este caso, en el último escenario se degrada drásticamente la performance para la ejecución secuencial, lo cual se refleja en los valores de δ y δ^2 . Sin embargo, esto no sucede con la ejecución en OpenMP donde la variabilidad de los tiempos de ejecución se mantiene dentro de parámetros aceptables.

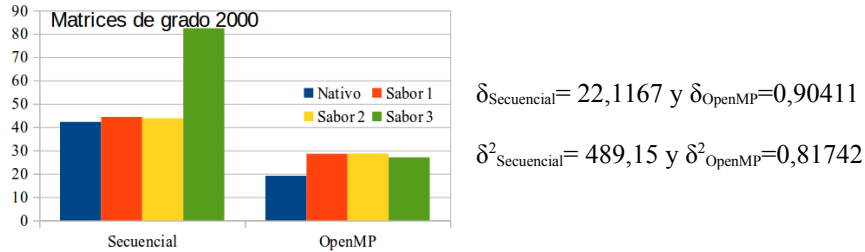


Figura 3: Tiempos de Ejecución para Matrices de grado 2000

Estos análisis permiten determinar que la mejora en la performance obtenida en la ejecución con OpenMP es estable sobre los tres escenarios cloud, o sea, el aumento en los VCPU (CPU virtuales o cores virtuales) no tienen un impacto significativo en los tiempos de ejecución.

A pesar de los resultados analizados, como se puede ver en la figura 4, la ejecución del algoritmo en OpenMP para matrices de grado 2000, tienen una escala en los tiempos de ejecución muy significativa en todos los escenarios basados en cloud.

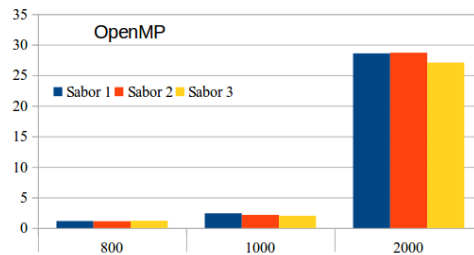


Figura 4: Tiempos de Ejecución de OpenMP

En la tabla 4, se ha calculado cuales serían los tiempos estimados de ejecución del algoritmo en OpenMP en cada escenario (se ha descartado el escenario *Nativo*) y para los tres grados de matrices. Esta estimación se ha realizado por mera aproximación y descartando los tiempos en los cuales los hipervisores pueden incurrir para contextualizar las instancias.

	800		1000		2000	
	Tiempo Real	Tiempo Esperado	Tiempo Real	Tiempo Esperado	Tiempo Real	Tiempo Esperado
Sabor 1	1,132979	1,4162238	2,405722	4,8114440	28,564535	28,564535
Sabor 2	1,100913	1,3761413	2,144922	4,2898440	28,682011	28,682011
Sabor 3	1,188093	1,4851163	2,002979	4,0059580	27,060603	27,060603

Tabla 4: Tiempos de ejecución estimados y reales de OpenMP por sabor (en segundos)

En la figura 5 se puede ver gráficamente las diferencias entre los valores de ejecución esperados y los que realmente le tomo al algoritmo ejecutarse. Como se puede ver, los tiempos de ejecución son aproximadamente catorce veces más lentos para matrices de grado 2000.

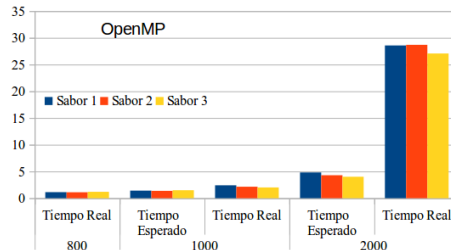


Figura 5: Tiempos de ejecución estimados y reales de OpenMP (en segundos)

A pesar de ello, en los escenarios basados en cloud para un grado determinado de matriz, migrar el cómputo a los otros escenarios genero mejores tiempos de ejecución que los esperados, tal como se muestra en la tabla 5.

		800	1000	2000
Sabor 1	Tiempo Real	1,132979	2,405722	28,564535
	Tiempo esperado	1,100913	2,144922	28,682011
Sabor 2	Tiempo Real	1,6994685	3,608583	42,8468025
	Tiempo esperado	1,188093	2,002979	27,060603
Sabor 3	Tiempo Real	1,467884	2,859896	38,24268
	Tiempo esperado			

Tabla 5: Tiempos de ejecución estimados y reales al migrar el escenario (en segundos)

En función de los resultados analizados, es posible inferir que bajo estos escenarios y con estos parámetros de ejecución, los escenarios más adecuados son el 1 y el 2, debido a que en el escenario 3, los tiempos de ejecución del algoritmo se degradan drásticamente.

7 Conclusiones

El cloud computing es un nuevo paradigma que está cambiando la ubicación de las infraestructuras informáticas hacia Internet. Con ello se consigue que los usuarios no tengan que gestionar ni sus propios servidores ni tampoco su software, ahorrando de este modo energía, espacio físico y personal técnico. Además, los sistemas cloud ofrecen un gran rendimiento en términos de escalabilidad, mantenibilidad y procesamiento masivo de datos en entornos dinámicos y de necesidades cambiantes.

Desde el punto de vista académico, el contar con una infraestructura cloud privada resulta un tema interesante de investigación y evaluación de cuan buena es esta solución para comenzar a migrar los datos, aplicaciones y cómputo a un entorno virtualizado. Sin embargo la migración, tiene ventajas y desventajas; entre las ventajas sobresale la posibilidad de contar con una plataforma de cómputo basada en instancias virtualizadas de recursos. La desventaja más relevante de migrar al cloud es la pérdida de performance producida por los procesos de comunicación y la virtualización de los recursos.

En este sentido, el grupo de investigación ha instalado y configurado una infraestructura cloud privada con OpenStack, dejándola en instancia de prueba para la

evaluación del impacto de la migración del cómputo a esta plataforma. Las pruebas realizadas sobre la infraestructura cloud instalada arrojaron resultados positivos, pues ha permitido evaluar cuál es el impacto que los procesos de virtualización sobre la ejecución de algoritmos en instancias virtualizadas de recursos. La conclusión más importante a la que se ha podido llegar es probar que cuando existe un cómputo intensivo el uso de infraestructura cloud brinda una mejor performance aun contar la ejecución nativa, que a priori debería ser la de mejor performance.

References

- [1] Kahanwal, D., & Singh, D. T. (2013). The distributed computing paradigms: P2P, grid, cluster, cloud, and jungle. *arXiv preprint arXiv:1311.3070*.
- [2] Rittinghouse, J. W., & Ransome, J. F. (2016). *Cloud computing: implementation, management, and security*. CRC press.
- [3] Antonopoulos, N., & Gillam, L. (Eds.). (2010). *Cloud computing: Principles, systems and applications*. Springer Science & Business Media.
- [4] Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing*. NIST Special Publication 800 – 145. 2011.
- [5] Chee, Franklin. *Cloud computing: technologies and strategies of the ubiquitous data center*. CRC Press. 2010.
- [6] Murazzo, M. A. (2015). *Análisis de una infraestructura cloud open source* Tesis de Especialización, Facultad de Informática – UNLP.
- [7] Murazzo, M. A., & Rodríguez, N. R. (2016). Evaluación del Impacto de Migración al Cloud. *Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação, 1(4)*.
- [8] Vogel, A., Griebler, D., Maron, C. A., Schepke, C., & Fernandes, L. G. (2016, February). Private IaaS Clouds: A Comparative Analysis of OpenNebula, CloudStack and OpenStack. In *2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)* (pp. 672-679). IEEE.
- [9] OpenStack. *OpenStack: The Open Source Cloud Operating System*. URL: <http://www.openstack.org/software/>
- [10] Murazzo, M. A., Tinetti, F. G., Rodríguez, N. R., & Guevara, M. J. (2015, June). Infraestructura de Cloud Computing. In *XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015)*.
- [11] OpenStack. *OpenStack Liberty*. URL: <https://www.openstack.org/software/liberty>

A Parallel Proposal for SEIR model using Cellular Automata

Facundo Casares, Cristian Tissera, Fabiana Piccoli

LIDIC. Universidad Nacional de San Luis,
Ejército de los Andes 950 - 5700 - San Luis - Argentina
{ptissera, mpiccoli}@unsl.edu.ar

Abstract. Cellular Automata have been used with success in simulations of simple and complex systems belonging to different scientific areas, such as chemistry, biochemistry, economy, physics, etc.. In this work, we propose to use it in order to specify and implement a simulation model that allows to investigate behavioural dynamics for seasonal flu. This work presents a general solution where parallel programming techniques of shared memory are applied. Finally some experimental results about performance and flu behaviour are showed.

1 Introduction

Throughout times, the diffusion and spread of disease were a major concern of the human being. There were cases where a disease caused the disappearance of an entire population and important demographic changes, some of them were the plague (Europe, XIV century), yellow fever (Buenos Aires, XIX century) and cholera (Asia, XIX century). Today the situation continues, there are diseases monitored in certain regions such as malaria, and other are new as Influenza A or persistent as AIDS. For all this, it is important and priority the study and control of these diseases and their mode of transmission or contagion. One way to address the problem is analysing how the disease is distributed in a specific population. The study and analyses of complex real systems like this can be done through some simulation models.

When it comes to simulate discrete dynamical systems, cellular automata (CA) has been successfully used in simulation of diffusion process and, it is a valid alternative when we work with discrete dynamic systems which have complex behaviours from a simple set of rules. These rules allow to specify the new state of a component based on its state and its neighbourhood. In this way, it is possible to model complex dynamic systems from the specification of the local dynamics of each component. Besides, the state of each of them can be calculated simultaneously, i.e. in parallel. The behaviour of CA can display graphically the system evolution, allowing an easy comprehension of the studied dynamics. Besides, CA have demonstrated to be very useful simulation tools at the time of constructing artificial scenes, mainly in domains not suitable for other approaches. In other word, CA can be used to simplify complicated relationships

by means local interactions. For example, transmission of rumor, diseases or computer viruses can be reduced to local interactions among individuals/computers, as the case. In this work, we focus in spread of diseases, particularly the influenza or flu[1–3].

Different techniques have been developed to study the spread of diseases. This technique divide population into different types considering the characteristics of the disease: susceptible, exposed (with or without symptoms), infected, infectious, recovered, vaccinated, isolated, diagnosed, etc.. According to the disease and its infectious agent, an individual can be in some of above states. From these states and the dynamics governing their compartments, different mathematical models arise, some of them are: SI (Susceptible-Infected), SIS (Susceptible-Infected-Susceptible), SIR (Susceptible-Infected-Recovered), SEIR (Susceptible- Exposed-Infected-Recovered), and other variants (SIRS, SEIRS, SEIQR, etc.). Particularly in this work we focus on the SEIR model. It is the most suitable for epidemics whose infectious agent is a virus. Generally when infectious agents are viruses, the individuals recovered or cured can achieve a state of resistance for same virus [4, 5].

Taking into account the previous aspects and the parallel nature of CA, we present a parallel solution to simulate the flu propagation using CA and SIRS models. This solution will allow to study, in short time, the spread of influenza or any virus mutation in different environments, considering type of population, its distribution and other characteristics; and to take decisions in consequence, such as vaccination campaigns, isolation, quarantine, etc.

The paper is organized as follows: the next sections describe all the previous concepts. Sections 3, 4 and 5 sketch the characteristics of seasonal flu, our parallel proposal, and its empirical performance. Finally, the conclusions and future works are exposed.

2 Previous Concepts

In this section, we explain the main concepts to develop this work.

2.1 Cellular Automata

In their research about the machines with auto-replication capabilities, John Von Neumann and Stanislaw Ulam were the first in formulate the Cellular Automata(CA). But was in 1970 when these systems received special attention, Jhon H. Conway proposed the *game of life*, the most known CA. Since that date, the CAs have grown in popularity within of the scientific community and actually they are considered to solve problems with different nature[6].

A CA is a mathematical system with discrete values in space, time and state. It has different characteristics, some of them are *auto-replication*, universal computation capabilities and *auto-organisation* effects. This last property plays a very important role at the time of explaining certain kind of behaviours observed in physical and biological phenomena [7, 8], in consequence, the CA have

been used, for example, to simulate different phenomena as chemical reactions, diffusion processes, hydrodynamic, mechanic, filtration, chaos theory and others.

A CA is a discrete dynamic systems with capacity to develop complex behaviours from a simple set of rules. It represents a grid of locally connected finite automata, each of them produces an output from several inputs, next state is a result to apply a transition function. The upcoming state of a CA cell depends of its own current state and the states of its neighbouring cells [9].

Intuitively, we consider a CA as a system composed by a array of cells A . Each cell c_i in A represents a finite automaton with a set of states Q , an input alphabet σ and a transition function $\delta : Q \times \sigma \rightarrow Q$. The input alphabet σ is given by all the possible combination of the cell states of the adjacent (neighbouring) cells. If we denote N_{c_i} to the set of cells that we consider as neighbour of cell c_i , and $|N_{c_i}| = n$ is the number of adjacent cells, then the input alphabet is $\sigma \equiv Q_n$. Usually, a cell c_i and its adjacent are considered and represented as a unique set $N = \{c_i\} \cup N_{c_i}$. N is referenced as the *neighbourhood*.

By all above exposed, a CA is defined as a 4-tuple $M = \langle A, Q, \delta, N \rangle$ where:

- A is a D -dimensional array, and each component (cell of the array) has associated a *finite automaton*.
- Q is a finite set of states (of the automaton) of a cell.
- N is the specification of which cells are included in a neighbourhood, $N \equiv \{c_i\} \cup N_{c_i}$ such that N_{c_i} are the adjacent cells to c_i .
- Let $\sigma \equiv Q^n$ where $n = |N_{c_i}|$ is the number of adjacent cells to c_i . The *transition function* of states, $\delta : Q \times \sigma \rightarrow Q$, is a mapping such that if $q_i \in Q$ is the state of the cell c_i in the time t and $q_{i+1}, q_{i+2}, \dots, q_{i+n} \in \sigma$ are the states of the adjacent cells to c_i , and

$$\delta(q_i, q_{i+1}, q_{i+2}, \dots, q_{i+n}) = q'_i$$

where $q_i, q_{i+1}, q_{i+2}, \dots, q_{i+n}$ are the states of the central cell and its neighbours at the time t and q'_i is the state of the central cell at the time $t + 1$.

In some cases, it is possible to specify probabilistic transition rules, where an arbitrary probability p can be associated to a transition rule. The semantic of this kind of rules establishes that, always that a cell matches the configuration of the specified neighbourhood in a probabilistic rule, the cell will have at time $t + 1$ the new state specified in such rule with probability p .

2.2 SIR Model and Derivated

In the modelling of diseases, several considerations must be taken into account, among them are important to consider: the infectious agents (they are responsible for transmitting the disease and condition the states through which passes an individual affected) and transmission modes (they can be person-to-person, by the environment, by some vectors such as insects or agents, or among animals of the same or different species).

Because all the numerous factors involved in a disease, it is impossible to study them of same way. A start point is to classify states in that an diseased individual can be. A set of possible states is:

- *S*: Healthy individuals and *Susceptible* to be infected.
- *E*: *Exposed* individual to disease, infected but not infect others (i.e., the disease is latent).
- *I*: *Infected* individuals who infect others.
- *R*: *Resistant* individuals to diseases (normally, it happens after that a person recovers from illness or vaccinates).

In a same time, an individual can be in a single stage of the disease, therefore for a population of N persons, if we consider the above set of state, the following equation must be satisfied:

$$S + E + I + R = N$$

Kermack and McKendrick in 1927a formulated a simple model, SIR model, which consists of three stages: *Susceptible*, *Infected* and *Recovered*. The SIR model is easily written using ordinary differential equations (ODEs), this implies a deterministic continuous model. It assumes encounters among infected (I) and susceptible (S) individuals at a rate proportional to their respective numbers in the population.

The analytical techniques are good to address problems in a basic way. But, in the case of disease epidemic study, the system is complex, and, in consequence, more realistic solutions with high level of detail are necessary.

2.3 Multi-threaded Parallel Programming

Multi-threading and parallel programming are concept different, in this work we apply them together to obtain good performance in a CA solution.

The multi-thread programming imply a single process and this process generates many threads. All of them share the same space memory, and they can be able to execute independently and at the same time: in parallel. The traditional multi-threading was used to do time-slicing or take advantage of the CPU idle time, i.e. while one of the threads waits, another thread could execute.

By its side, parallel programming allows explicitly breaks the task down into smallest units, where each unit can be executed in parallel on a single CPU core. When it is possible divide the task and its sub-tasks share the same memory space and run in parallel, the problem can be solved applying Multi-threaded Parallel Programming.

To solve a problem by applying Multi-threaded Parallel Programming, we can use OpenMP(Open Multi-Processing)[10]. It is an API that supports multi-platform shared memory multiprocessing programming, it achieves parallelism via multi-threading and shared-memory.

In this work, we present a simulation system for flu transmission using CA and SEIR as models and multi-threaded parallel programming techniques.

3 Seasonal Flu

Influenza is a viral infectious disease that affects, primarily, the respiratory tract of humans. Usually it accompanied by other symptoms such as sore throat, weakness, dry cough, fever, and muscle aches, of stomach and head . In some cases, it may be complicated and derive in pneumonia becoming fatal. This can occur in certain age groups, such as young children and elderlies. There are three types of seasonal influenza: A, B and C. The influenza virus A and B are the most common, they are classified into subtypes according to the combination of two proteins in virus surface (H and N)[11, 12].

Virus transmission is done person-to-person, mainly through particles ejected when a sick person coughs, sneezes or talks. Also, it can be transmitted by means blood or contact with surfaces or objects contaminated. Besides, flu virus is resistant in a dry and cold environment, this property allows its rapid spread mainly in autumn and winter, seasons when it becomes seasonal epidemic. The virus can keep its infections level by about one week at body temperature, however, there are patients that require 15 days of recovery. Most people recover without medical treatment. Antibiotics are only useful if there is a bacterial infection.

A infected person with the flu virus goes through an incubation period (approximately from two to four days). The contagious period begins one day before that person has symptoms (this is a serious problem, a person could be spreading the influenza without knowing who is sick). After a week, the transmission power is reduced, even it disappears. The figure 1 summarizes how the disease evolves in a person, from he/she is susceptible until his/her recovering or, in the worst case, death.

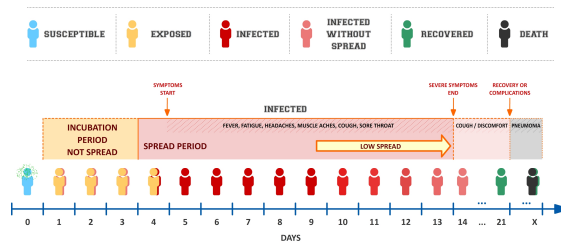


Fig. 1. Influenza progression in people

The most effective way to prevent the flu and its serious consequences is vaccination. In healthy adults, it can provide reasonable protection, while in elderlies can reduce its severity, the incidence of complications and deaths. As *C* influenza cases are much less frequent, generally the vaccines try influenza *A* and *B*. They are usually trivalent, they contain purified and inactivated proteins of the three strains most common in the following epidemic: two subtypes of influenza *A* and one *B*. The vaccination effectiveness depends on the match between the vaccine virus and surrounding virus. Moreover, a vaccine made one

year may not be effective to the next by two reasons: the virus change and mutate rapidly, and the strains have variable dominance[12].

4 Parallel Simulation of Seasonal Flu Epidemic

The epidemiology studies the factors of potentially harmful infectious agents that affect a particular population, and tries to explain and predict how the disease evolves in the time. As explained earlier, the simulation models that use CA concepts are ideal to represent this type of real systems. Through setting the main features of the problem, CA recreates a virtual world that comes alive and gives an approximation of what would happen in the real world.

In this section, we describe a model based on epidemiological model with SEIR approach and CA. By means this model, it is possible to analyse the effects of seasonal influenza in a population with a certain territorial distribution.

4.1 SEIR-CA Model

The *SEIR – CA* model has a cellular space defined by a finite two-dimensional lattice. Each cell of the automata is a place busy by only one person. In this work, the cellular space (2-D array) represents a social space in which the individuals can interact, this means, two adjacent cells occupied represent two people in touch. To perform the simulation, it is necessary to establish the following considerations:

- **Neighborhood:** For this systems, we consider the known Moore neighborhood: eight cells surrounding the central cell define the neighborhood . With this, every individual interacts with at most 8 people by once.
- **Cell State:** a cell is in one state of $Q = \{B, F, S, E, I, W, R, D\}$ where:
 - *B*: Automata limits (when borders are absorbent).
 - *F*: Free Cell (There is not any person and it can be selected to occupy).
 - *S*: The person is susceptible to contract flu.
 - *E*: When the person is in incubation period but not spread.
 - *I*: The person is infected and, he/she spreads the flu
 - *W*: The person is infected but does not spread (Infected without spread).
 - *R*: When the person is recovered.
 - *D*: Dead person. Generally, the deaths can be by any complication.

The transition through each of the states is shown in Figure 1.

- **Initial Configuration:** Before simulation begins, it is necessary to set relevant information such as size of population surface (CA Size), how the population is distributed in the surface, which is the infection percentage of population and their ages.
- **Virtual Clock:** Time is discrete, at the beginning of simulation, an interval of time is set. During this interval, a person can move to any of the neighboring cells and relate with other people. Generally, this movement follows some probabilistic pattern.

– **Model Evolution Rules:** There are three kinds of rules which are: the rules related to CA, with spreading of diseases, and those associated with persons movement inside CA. Each one of rules are:

1. **CA Rules:**

- If a cell is in state B (CA with absorbent borders), its state does not change at no time of the simulation.

2. **Diffusion and Spread Rules:** A cell occupied by a person in time t , also will be occupied in time $t+1$. It can change its state according to:

- If at time t , the central cell is susceptible (S) and, some of its adjacent cells are infected (I), the central cell will be incubating the influenza (E) at time $t+1$, but will not spread. The probability of state change is proportional to the number of adjacent cells I ($S \rightarrow E$).
- If the cell is incubating flu (E) and, the time t is the end of the asymptomatic period, at time $t+1$ the cell will be in infected state I ($E \rightarrow I$).
- After 8 days, a cell infected (I) in time t will pass to state infected without spread (W) at time $t+1$ ($I \rightarrow W$).
- In time t , a cell infected (I or W) could become recovered (R) or dead (D) state (time $t+1$). The selection between two stages depends of a probability function ($I \rightarrow \{R, D\} | W \rightarrow \{R, D\}$).

3. **Rules of Person Motion:** A person can move to a neighbour cell if it is free (state F). All free neighbour cells have the same probability to be occupied.

An important aspect to note is that the matrix does not necessarily represent a spatial universe. The model represents the people interrelation, a movement in the surface is an abstraction, it can mean that a person moves to speak with other (for example an officemate) or, he/she goes to a business and relates with a vendor. When a person comes in a neighbourhood of another, this means an interaction between two people. As we assume an uniform distribution of probabilities (a person can move equally to any free adjacent cells), this model simplifies the problem.

The movement is not physical, it models the interaction between a person with its neighbourhood.

Once defined all CA characteristics, in next section we explain the main issues of our proposal: a parallel SEIR-CA using shared memory.

4.2 Parallel Solution

To solve the *SEIR-Flu* in parallel, we consider the shared memory paradigm, in consequence it is necessary to consider synchronization mechanisms to access to memory or, other programming techniques. For the implementation, some characteristics are:

- OpenMP is selected as application programming interface (API).
- In each timestep, two CA are necessary, one represents the state in time t (input CA), and the other is the output (CA in time $t+1$).

- Each thread takes a particular cell and works over it. When it finishes, takes the next cell and starts again.

The solution is structured in three stages, each stage does:

- *First Stage*: In this stage, we calculate for each cell: the next state and a list of movement intentions. For that we apply the CA rules. When this stage finishes, the new CA for time $t + 1$ is obtained and each free cell has a intention list to be occupied by one of its neighbour.
- *Second Stage*: Taking into account the intentions list of each free cell, one of candidates is selected in random form. To improve the solution and save memory, we use a bitmap technique for intention lists. Each list is represented by 9-bits to the Moore neighbourhood plus the central cell (fifth bit). If some position is set to 1, this means that central cell wants to move there.
- *Third Stage*: After the second stage, the data structures of CA are inconsistent state. This stage carries to a safe state, all cells and their positions are adjusted. In time $t + 1$, the input and output CA exchange their roles.

Every stage are made in parallel, independent and sequential sections without using synchronization mechanisms.

Other characteristic is which is the breakpoint of simulation. It can end by two reasons, whichever first occurs, they are:

- Maximum simulation time: A time limit for simulation is established, if this is reached, the simulation ends.
- Some stable or starvation state: The simulation is based on spread from person to person, once the last sick person is cured or dies, none new flu infection in the current population can be developed. For example, when population is small, the rate of spread is not enough for the disease persists over 4 months, usually a stable state is reached and the simulation ends before 120 days.

The next section some experimental results of this parallel *SEIR-Flu* are displayed.

5 Experimental Results

In this section, we show and analyse the experimental results for the parallel solution of *SEIR-Flu*.

The environment of simulations, sequential and parallel, were in a multi-core computer whose characteristics are: 2 processors AMD Opteron 6272, 2.1GHz, 16 cores per processor, RAM memory of 64GB Memory (16x4GB), 1333MHz, OpenMP 4.0 and Debian 8.

In order to obtain the results, different scenarios of simulation are considered. Each of them is a combination of next parameters:

- The maximum time of simulation: 120 days. The stationary flu has more propagation in winter, in others seasons its infection decreases drastically.

- Virtual clock: A timestep is equivalent to 1 hour.
- Initial infection factor: 1% of population has flu.
- Population: 50% of cells of grid.
- Size of Square Grid: 50, 100, 200, 400, 800, 1600 and 3200 by side. For lack of space, we only report the more representative results .

We consider a stationary type of population pyramid [13], each individual has an age between 1 to 90 years. The degree susceptibility to infection and the mortality rate is determined according of individual age[14, 12]. In all populations, there are three groups of person: *Children* (Up to 6 years old), *Young-Adult* (7 to 60 years) and *Elderlies* (Greater than 61 years), each of them has a susceptibility equal to 35%, 20% and 35% respectively.

For mortality, we recognize the following groups and each mortality rate: *Under 3 years* (8%), *4 to 10 years* (5%), *11 to 18 years* (2%), *19 to 50 years* (0.5%), *51 to 60 years* (2%), and *more than 60 years* (8%).

Each reported value is the average of 10 executions of *Parallel SEIR-Flu*. For parallel solution and each scenarios, we use 2, 4, 8, 16, 32 and 64 threads. To sequential solution, we consider the same solution but for only one processor.

In first time, we evaluate if our parallel proposal works well and its performance is better than sequential solution. The figure 2(a) shows the speedup reached for every simulation scenario and different numbers of threads. In majority of scenarios, we achieve speedup. Although this is not close to optimal, we can reduce significantly the computational time of simulation. Similar behaviours and earnings are observed in Figure 2(b) for the same size of population (50%) but the 10% of them are infected initially.

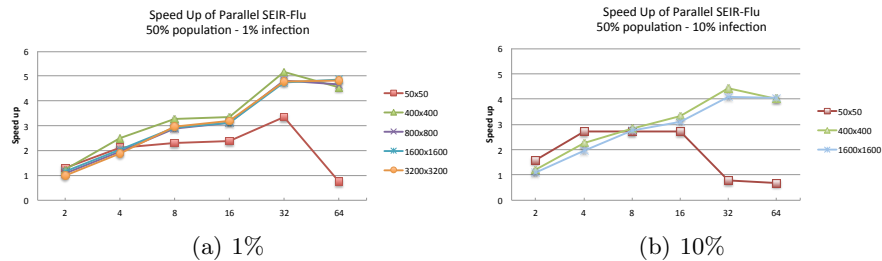


Fig. 2. Speedup of Parallel SEIR-Flu for 50% population with 1% and 10% initial infection

Besides of performance results, we can observe some behaviour of the flu infection process. For example, a higher level of initial infection implies less time to reach a state free of flu in the population of each grid, the Figure 3 shows this situation.

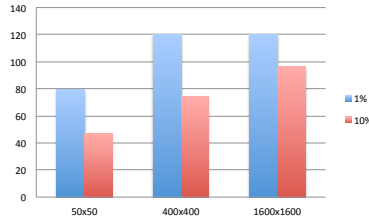
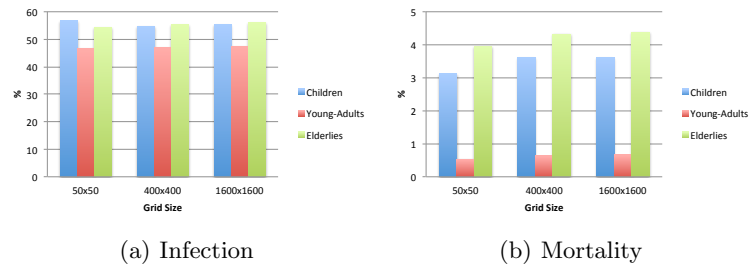


Fig. 3. Days of SEIR-Flu for 1% and 10% initial infection

In Figure 4, we sketch the percentage of infection(4(a)) and mortality (4(b)) for each populations group: Children, Young-Adults and Elderlies. The grid occupation is 50% and the 10% are infected.



(a) Infection

(b) Mortality

Fig. 4. Percentage of Mortality and Infected Individuals for 50% population and 10% initial infection

Figure 5 summarizes the speedups achieved for different levels of grid occupation when 1% of initial population is infected (Figures 5(a), 5(b) and 5(c)). Besides, Figure 5(d) displays how many days are necessary to reach a state free of flu.

6 Conclusions and Future Work

We presented a parallel model capable to simulate the propagation of seasonal flu. The proposed model uses the CA and SEIR concepts to analyse the effects of seasonal influenza in a population with a certain territorial distribution modelling the people interaction.

One of objectives of this work is to develop and select techniques of high performance computing in order to perform and execute large-scale complex simulations of diffusion processes. While our proposal is not yet complete, the results of our multi threading implementation are encouraging and give us an

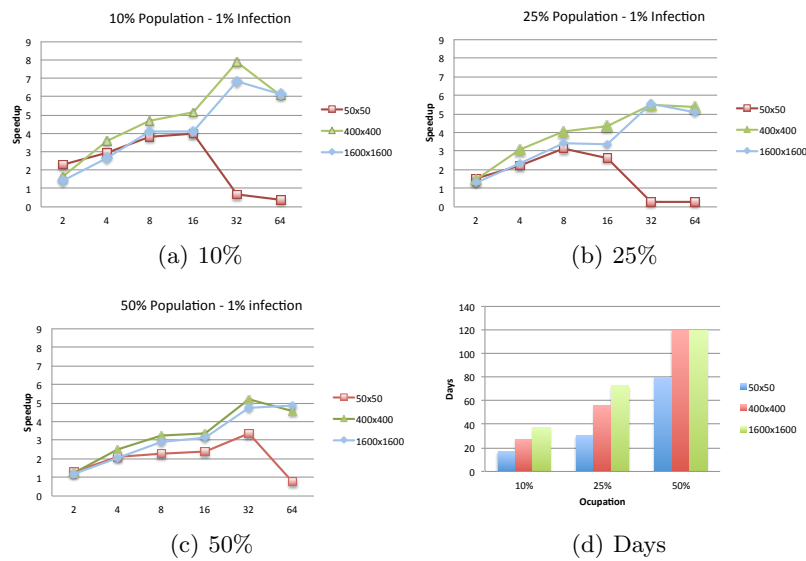


Fig. 5. Speedup and Days of SEIR-Flu for different levels of occupation with 1% initial infection for and

initial framework for future works in the characterization of diffusion processes like signal propagation, avalanches, rumours spread, etc.

As future works, it is important to decide the optimal size of set of threads needed to solve the model taking into account the performance parameter to try characterize this kind of applications and this experience will be transferred to others developments of more complex parallel applications.

References

1. F. Chierichetti and A. Panconesi. Rumor spreading in social networks. *Theoretical Computer Science*, 412(24):2602–2610, 2011.
2. M. Nekovee, Y. Moreno, G. Bianconi, and M. Marsili. Theory of rumour spreading in complex social networks. *Physica A: Statistical Mechanics and its Applications*, 374(1):457–470, 2007.
3. S. White, A. del Rey, and G. Sánchez. Modeling epidemics using cellular automata. *Applied Mathematics and Computation*, 186(1):193–202, 2007.
4. A. Huppert and G. Katriel. Mathematical modelling and prediction in infectious disease epidemiology. *Clinical Microbiology and Infection*, 19(11):999–1005, 2013.
5. T. Johnson and B. McQuarrie. Mathematical modeling of diseases: Susceptible-infected-recovered (sir) model. In *University of Minnesota, Morris, Math 4901 Senior Seminar*, 2009.
6. D. Reyes. Descripción y aplicaciones de los autómatas celulares, 2011. Technical Report, Autónoma de Puebla.

7. S. Kauffman. Emergent properties in random complex automata. *Physica D: Nonlinear Phenomena*, 10(1):145–156, 1984.
8. S. Wolfram. Universality and complexity in cellular automata. *Physica D: Nonlinear Phenomena*, 10(1):1–35, 1984.
9. J. Klüver and C. Klüver. *On communication. An interdisciplinary and mathematical approach*, volume 40.
10. B. Chapman, G. Jost, and R. van der Pas. *Using OpenMP: Portable Shared Memory Parallel Programming*. Number v. 10 in Scientific Computation Series. MIT Press, 2008.
11. C. Beauchemin, J. Samuel, and J. Tuszynski. A simple cellular automaton model for influenza a viral infections. *Journal of theoretical biology*, 232(2):223–234, 2005.
12. World Health Organization. Influenza (seasonal), 2014. Fact sheet N211.
13. P. Hernández. *Demografía y antropología demográfica*. CONACULTA-INAH, 2004.
14. Argentina Ministerio de Salud. *Manual para el fortalecimiento de la vigilancia de la enfermedad tipo Influenza utilizando la estrategia de Unidades Centinela*. 2011.



XIV Workshop Computación Gráfica, Imágenes y Visualización (WCGIV)

Comité Científico:

Abásolo, María José (Argentina)
Baldassarri, Sandra (España)
Collazos Ordóñez, César Alberto (Colombia)
D'Amato, Juan Pablo (Argentina)
Ganuzza, María Luján (Argentina)
Gazcón, Nicolás (Argentina)
Gröller, Eduard (Austria)
Larese, Mónica (Argentina)
Larrea, Martín (Argentina)
Lecumberry, Federico (Uruguay)
Manresa-Yee, Cristina (España)
Meschino, Gustavo (Argentina)
Pina, Alfredo (España)
Simari, Patricio (USA)
Urribarri, Dana (Argentina)



XIV Workshop Computación Gráfica, Imágenes y Visualización (WCGIV)

ID	Trabajo	Autores	
9156	Agrupamiento de trayectorias vía clustering espectral incremental	Luciano Lorenti (UNLP), Javier Giacomantone (UNLP), Armando E. De Giusti (UNLP)	222
9157	Real-Time Estimation of Illumination Direction for Augmented Reality with Low-Cost Sensors	Matias Nicolas Selzer (UNS), Martín Larrea (UNS)	232
9216	Estudo e implementação de técnicas para identificação de anuros em imagem	Angélica Larissa Marques Queiroz(IFSP), César Alberto da Silva (UFMG), Andrea Padovan Jubileu (IFSP), Linnyer Beatriys Ruiz (UEM)	241
9284	Herramientas para la detección y seguimiento de personas a partir de cámaras de seguridad	Leonardo Dominguez (UNCPBA), Alejandro J. Perez (UNCPBA), Aldo J. Rubiales (UNCPBA), Juan Pablo D'Amato (UNCPBA), Rosana Barbuzza (UNCPBA)	251
9324	Evaluating the spoken dialogue system of a conversational character: A Simulation Study	Yoselie Alvarado(UNSL), Claudia Ruth Gatica (UNSL), Verónica Gil Costa (UNSL), Roberto Guerrero (UNSL)	261
9329	Un nuevo descriptor para la identificación de personas mediante caracteres simples	Verónica Aubin (UNLaM), Marco Mora (UCM), Matilde Santos (UCM)	271
9332	Experiences accelerating features selection in Viola-Jones algorithm	Germán Ezequiel Lescano (UNSE), Pablo Santana Mansilla (UNSE), Rosanna N. Costaguta (UNSE)	281
9334	Non-Verbal Communication for a Virtual Reality Interface	N. Jofré (UNSL), G. Rodríguez (UNSL), Y. Alvarado (UNSL), J. Fernandez (UNSL), R. Guerrero (UNSL)	291

Agrupamiento de trayectorias vía clustering espectral incremental

Luciano Lorenti, Javier Giacomantone, Armando De Giusti

Instituto de Investigación en Informática (III-LIDI),
Facultad de Informática - Universidad Nacional de La Plata - Argentina.
La Plata, Buenos Aires, Argentina.
{llorenti,jog,degiusti}@lidi.info.unlp.edu.ar

Resumen. El agrupamiento de trayectorias permite entender los patrones de movimiento de los objetos capturados en vídeo. Debido a su naturaleza secuencial, los datos de trayectorias son adquiridos cuadro a cuadro de forma incremental. Este procedimiento implica actualizar las estructuras detectadas por el algoritmo de agrupamiento ante la llegada de nuevos datos. En este artículo se presenta un método de agrupamiento espectral incremental, que permite extraer las trayectorias representativas de los patrones de movimiento. El método propuesto explota un mecanismo orientado a reducir la demanda computacional en el cálculo de autovectores en cada iteración. Se presentan resultados experimentales sobre trayectorias de vídeos reales y trayectorias sintéticas, que permiten evaluar el método propuesto.

Palabras clave: Aprendizaje no supervisado, Clustering espectral incremental, Agrupamiento de trayectorias

1. Introducción

Las trayectorias de los objetos capturados en video son elementos importantes para el análisis de patrones de movimiento [1]. Las técnicas de agrupamiento, *clustering*, son un componente clave para el análisis de trayectorias. Su utilización resulta conveniente cuando en lugar de modelar y analizar el movimiento de un objeto individual, se procesan múltiples trayectorias para detectar las estructuras subyacentes de las actividades en video [2]. Las técnicas de clustering de trayectorias proveen beneficios para muchas tareas de visión automática como segmentación de movimientos [3], detección de objetos [4], reconocimiento de acciones [5] y modelado de escenas [6].

Una propiedad usual de las aplicaciones de seguimiento de objetos es que nuevas trayectorias son incorporados a la escena en cada instante de tiempo [7]. Frente a estos datos continuos, la información relacionada con las trayectorias debe ser procesada en forma incremental. Un punto importante a notar es que los agrupamientos que se encuentren distantes a los nuevos datos no se verán influidos. Por lo tanto, un enfoque que permite manejar datos que varían en

el tiempo consiste en ajustar los agrupamientos que se encuentren cercanos en algún sentido a los datos agregados o removidos para un instante de tiempo dado.

Por lo general las trayectorias se encuentran en un subespacio de baja dimensionalidad [8]. Es por esto que, usualmente, las trayectorias primero se representan en un espacio de baja dimensión antes de realizar el agrupamiento. Las técnicas de agrupamiento espectral obtienen una representación de los datos en un espacio de dimensión inferior obteniendo la descomposición en autovectores de la matriz laplaciana asociada al grafo de semejanza de los datos. Un enfoque posible para el problema del agrupamiento espectral incremental consiste en calcular los autovectores de la matriz laplaciana del grafo de proximidad en cada instante de tiempo, sin embargo, esto es computacionalmente costoso dado que resolver el sistema de autovalores es de orden cúbico con respecto a la cantidad de elementos en el grafo. Un modo de mejorar la eficiencia de los métodos de agrupamiento espectral es calcular una aproximación de la descomposición de autovectores en cada iteración utilizando el método de Nystrom [9] en el que se calculan las semejanzas entre un subconjunto de patrones y se aproximan las semejanzas restantes. Se ha abordado, recientemente, el problema de actualizar eficientemente el sistema de autovectores ante una modificación del grafo de semejanza [10] [11] [12].

En este trabajo se presenta un método de agrupamiento espectral incremental de trayectorias en el que en cada instante de tiempo se actualiza el grafo de semejanza entre los segmentos de trayectorias. Se aproximan los autovectores de la matriz laplaciana del grafo de semejanza utilizando lo calculado en el instante de tiempo anterior, se realiza un agrupamiento de los segmentos de trayectorias y, por último, se obtiene una trayectoria representativa para cada grupo. La evaluación del método propuesto se realizó utilizando trayectorias reales obtenidas en vídeos de vigilancia y trayectorias simuladas.

El artículo está organizado del siguiente modo, en la sección 2 se describe el problema del agrupamiento de trayectorias. En la sección 3 se presenta el mecanismo de agrupamiento espectral incremental. En la sección 4 se expone el método propuesto. En la sección 5 se muestran los resultados experimentales obtenidos. Finalmente, en la sección 6 se presentan las conclusiones.

2. Agrupamiento de trayectorias

En el contexto de este trabajo una trayectoria consiste en una secuencia de puntos. Cada punto p_i tiene asociada una información espacial $pos_{p_i} \in \mathbb{R}^n$ y una información temporal $t_{p_i} \in \mathbb{R}$. Un segmento de trayectoria es un segmento $s = p_i p_j (i < j)$ donde p_i y p_j son puntos elegidos de la misma trayectoria. Cada segmento de trayectoria tiene asociado, a su vez, información temporal: $t_s = t_{p_j}$. Se define $B(s)$ como el rectángulo delimitador mínimo de s . Un cluster es un conjunto de segmentos de trayectorias. Los segmentos de trayectorias que pertenecen al mismo cluster son similares de acuerdo a una función de semejanza. Segmentos distintos de una misma trayectoria puede pertenecer a agrupamientos

diferentes, dado que el agrupamiento se realiza sobre los segmentos y no sobre las trayectorias. Dado un conjunto de trayectorias $I = \{TR_1, \dots, TR_n\}$ obtenidas en un instante de tiempo t , la salida del método propuesto para ese instante consiste en un agrupamiento de los segmentos de las trayectorias $TrClu = \{C_1, \dots, C_k\}$ y una trayectoria representativa $\forall C_i \in TrClu$. Una trayectoria representativa de un cluster es una secuencia de puntos que indica el comportamiento mayoritario de los segmentos de línea pertenecientes al mismo.

2.1. Semejanza entre segmentos

La función de distancia entre segmentos de trayectorias propuesta por [13] está compuesta por tres componentes: la distancia perpendicular, la distancia paralela y la distancia angular. La figura 1 muestra las tres componentes de la distancia. Sean $L_i = s_i e_i$ y $L_j = s_j e_j$ segmentos de línea d -dimensionales. Donde $s_i, e_i, s_j, y e_j$ son las posiciones en el espacio de los puntos iniciales y finales de cada segmento. Asignemos, sin perder generalidad, al segmento más largo el nombre L_i y al segmento más corto, el nombre L_j .

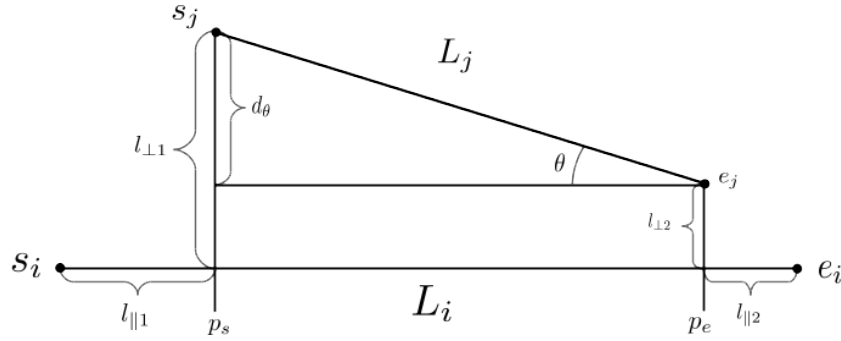


Figura 1: Componentes de la función de distancia para los segmentos de línea

La distancia perpendicular entre L_i y L_j se define como:

$$d_{\perp}(L_i, L_j) = \frac{l_{\perp 1}^2 + l_{\perp 2}^2}{l_{\perp 1} + l_{\perp 2}}$$

donde $l_{\perp 1}$ es la distancia euclídea entre el punto s_j y el punto p_s . El punto p_s es la proyección de s_j sobre L_i . La distancia $l_{\perp 2}$ se define de la misma manera para e_j y p_e . La distancia paralela entre L_i y L_j se define como:

$$d_{\parallel}(L_i, L_j) = \min(l_{\parallel 1}, l_{\parallel 2})$$

Supongamos que la proyección de los puntos s_j y e_j en L_i son p_s y p_e respectivamente. Se define $l_{\parallel 1}$ como el mínimo de la distancia euclídea de p_s a s_i y e_i . Del mismo modo, $l_{\parallel 2}$ es la distancia euclídea mínima de p_e a s_i y e_i . La distancia angular entre L_i y L_j se define como:

$$d_{\theta}(L_i, L_j) = \begin{cases} \|L_j\| \sin(\theta) & \text{si } 0^{\circ} \leq \theta \leq 90^{\circ} \\ \|L_j\| & \text{si } 90^{\circ} \leq \theta \leq 180^{\circ} \end{cases}$$

donde $\|L_j\|$ es la longitud de L_j y α el ángulo de intersección mas pequeño entre L_i y L_j . Finalmente, se define la distancia entre dos segmentos de trayectoria de la siguiente manera:

$$dist(L_i, L_j) = d_{\perp}(L_i, L_j) + d_{\theta}(L_i, L_j) + d_{\parallel}(L_i, L_j) \quad (1)$$

2.2. Trayectoria representativa

La trayectoria representativa de un cluster describe el movimiento general de los segmentos de trayectorias que pertenecen al mismo. En el método propuesto en [13], ilustrado en la figura 2, los puntos que componen la trayectoria representativa son extraídos analizando los puntos finales e iniciales de cada segmento de trayectoria. La idea central del algoritmo consiste en hacer pasar una línea vertical a lo largo de los segmentos de trayectorias en la dirección del eje mayor del cluster. Se contabiliza el número de segmentos que son atravesados por la línea de barrido. Este número solo cambia cuando la línea de barrido pasa por un punto de inicio o por un punto final de un segmento. Si este número es mayor o igual a un parámetro definido por el usuario, denominado $MinLns$, se calcula la coordenada promedio de los segmentos de línea con respecto al eje mayor y se inserta el punto promedio en la trayectoria representativa, caso contrario se saltea el punto que se está procesando. Además, con el objetivo de suavizar la trayectoria generada, si el punto que se está procesando se encuentra demasiado cerca de un punto ya insertado en la trayectoria representativa, ese punto no es tenido en cuenta.

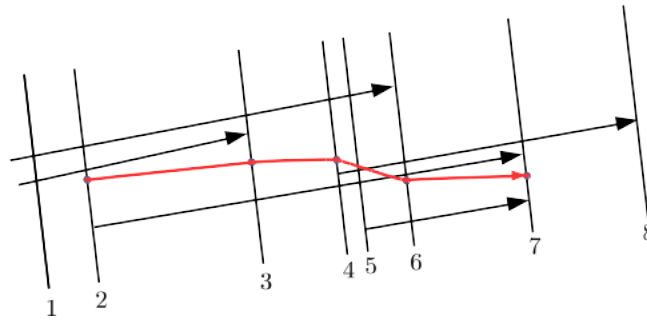


Figura 2: Trayectoria representativa de un agrupamiento. $MinLns = 3$

3. Clustering espectral incremental

Dado un conjunto de patrones $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^m$ y una función de semejanza $d(x_i, x_j) = \exp\left(\frac{-d^2(x_i, x_j)}{\alpha^2}\right)$, es posible construir un grafo de semejanza $G = (V, E)$ tal que $V = X$ y los pesos de las aristas estén dados por $d(x_i, x_j)$. Los algoritmos de agrupamiento espectral obtienen una representación de los datos en un espacio de dimensión inferior en la que resulta sencillo aplicar un algoritmo de clustering tradicional [14] [15]. Este espacio se obtiene calculando la matriz de afinidad W del grafo G y resolviendo el siguiente problema de optimización:

$$\begin{aligned} \max_{U \in \mathbb{R}^{n \times m}} \quad & Tr(U^T L U) \\ \text{s.t.} \quad & U^T U = I \end{aligned} \quad (2)$$

donde $L = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ es la matriz laplaciana de W y D es una matriz diagonal con la suma de las filas de W ubicadas en su diagonal principal. Una vez obtenido U sus filas son consideradas como las nuevas coordenadas de los patrones. En este nuevo espacio los pares de puntos semejantes se encuentran cercanos. Esto se debe a que los autovectores de la matriz laplaciana minimizan la siguiente función objetivo [16]:

$$\sum_{i,j} W_{ij} \|u^{(i)} - u^{(j)}\|^2$$

donde $u^{(i)} = [u_1(i), \dots, u_m(i)]$ es la representación m -dimensional del i -ésimo patrón.

3.1. Escalado local del peso de las aristas

El parámetro de escalado α de la función de semejanza es una medida que determina cuando dos puntos son considerados semejantes. La selección del parámetro se realiza, por lo general, de forma manual. La presencia de un escalado fijo para todo el conjunto de datos ocasiona problemas cuando se encuentran agrupamientos de distintas densidades. Los métodos de clustering espectral pueden no encontrar un buen agrupamiento cuando los datos contienen múltiples escalas, incluso utilizando el α óptimo. En lugar de seleccionar un sólo parámetro de escala, en [17] se propuso un método para calcular la escala local de α_i para cada punto s_i . La distancia de s_i a s_j vista por s_i es $\frac{d(s_i, s_j)}{\alpha_i}$ mientras que la distancia s_i a s_j vista por s_j es $\frac{d(s_i, s_j)}{\alpha_j}$. Por lo tanto la semejanza entre un par de puntos puede ser escrita como:

$$W_{ij} = \exp\left(-\frac{d(s_i, s_j)^2}{\alpha_i \alpha_j}\right)$$

Un parámetro de escalado específico para cada punto permite configurar la escala de acuerdo a la densidad encontrada en la vecindad del punto i y del punto j . La selección de la escala local α_i puede ser establecida estudiando los estadísticos locales de la vecindad del punto s_i .

3.2. Número de Clusters

Una herramienta para determinar el número de agrupamientos utilizando las técnicas de clustering espectral consiste en encontrar una caída drástica en la magnitud de los autovalores. Sea $\lambda_1, \lambda_2, \dots, \lambda_m$ los autovalores ordenados de menor a mayor de la matriz laplaciana. El objetivo es elegir un número k tal que los valores de $\lambda_1, \lambda_2, \dots, \lambda_k$ sean pequeños y λ_{k+1} sea relativamente grande [18]. Se define, por lo tanto, $k_{\text{opt}} = \underset{i}{\operatorname{argmax}}(|\lambda_i - \lambda_{i+1}|)$.

3.3. Aproximación incremental de los autovectores

Debido a que nuevos segmentos de trayectorias son incorporados al algoritmo de agrupamiento en cada instante de tiempo, el objetivo es procesar dichos datos y producir nuevos agrupamientos utilizando un enfoque incremental.

En [11] se propuso un método incremental eficiente que a partir de los autovectores de la matriz laplaciana de un grafo G_t permite obtener los autovectores de la matriz laplaciana del grafo G_{t+1} que se obtuvo agregando o eliminando vértices o modificando los pesos de las aristas de G .

Para esto plantean a las modificaciones del grafo como la suma de matrices simétricas $L_{t+1} = L_t + U$, donde L_t es la matriz laplaciana del grafo G_t y U es una matriz que codifica el cambio realizado a G_t para transformarlo en G_{t+1} . La idea central consiste en calcular la descomposición de L_t y luego aproximar la descomposición de rango k de $L_{t_k} + U$, donde L_{t_k} es la aproximación de L_t usando los k autovectores mas grandes. El método aproxima los autovectores de $L_k + U$ a calculando $\tilde{Q}H_k$, donde \tilde{Q} es una matriz con columnas ortonormales que generan $L_k + U$ y H_k representan los k autovectores mas grandes de la matriz $\Delta = \tilde{Q}(L_k + U)\tilde{Q}^T$. En [11] se muestra que las dos matrices pueden ser obtenidas a partir de L_k y U en $O((k^2 + p^2)(p + k) + np(p + k))$, donde n es la cantidad total de patrones, p es la cantidad de datos modificados y k es la cantidad de autovectores obtenidos.

4. Descripción del método propuesto

Dado el tamaño de la ventana de tiempo a analizar tw , un conjunto de segmentos de trayectorias $C = \{l_1, l_2, \dots, l_n\}$, un segmento l_i , y un parámetro de vecindad β , se define el conjunto de segmentos vecinos de l_i como

$$N(l_i, \beta) = \{ l_j \mid \beta B(l_i) \cap \beta B(l_j) \neq \emptyset \forall_{i \neq j} \}.$$

donde $\beta B(l_j)$ es el rectángulo delimitador mínimo de l_j escalado por β .

En el método propuesto se utiliza como función de semejanza entre dos segmentos de trayectoria:

$$w(s_i, s_j) = \exp\left(-\frac{\operatorname{dist}(s_i, s_j)^2}{\alpha_i \alpha_j}\right) * \exp\left(-\frac{|t_{s_i} - t_{s_j}|}{0,5tw}\right) \quad (3)$$

donde $\alpha_i = \frac{\sum_{\forall l_j \in N(l_i, \beta)} \text{dist}(l_i, l_j)}{|N(l_i, \beta)|}$ y dist es la función de distancia entre dos segmentos definida en la ecuación 1.

Sean k el máximo número posible de clusters, β un parámetro de vecindad, T el número de iteraciones que transcurren sin volver a calcular el sistema de autovectores. Sea $G_{t-1} = (V_{t-1}, E_{t-1})$ el grafo de segmentos de trayectorias obtenido en la iteración $t - 1$, J_{t-1} los autovectores aproximados obtenidos en la iteración $t - 1$ y traj_i los segmentos de trayectorias extraídos en la iteración i . En cada iteración el método realiza los siguientes pasos:

1. $G_i = G_{i-1}$ donde $G_i = (V_i, E_i)$.
2. Se eliminan del grafo G_i los segmentos $\{s_j | s_j \in V_i, i - t_{s_j} > tw\}$.
3. $V_i = V_i \cup \text{traj}_i$. Se conecta cada $t \in \text{traj}_i$ con sus $N(t, \beta)$. El peso de la arista está dado por la ecuación 3.
4. Se calcula la escala local $\forall t \in \text{traj}_i$.
5. Se obtiene la matriz laplaciana L_i a partir de G_i de acuerdo a [11].
6. Si $i \bmod T = 0$.
 - a) Se recalculan las escalas locales de cada punto.
 - b) Se vuelven a calcular los k autovectores J_i a partir de L_i .
7. Si $i \bmod T \neq 0$.
 - a) Se aproximan los k autovectores J_i de acuerdo a la sección 3.3 a partir de J_{i-1} .
8. Se calcula el número de agrupamientos k_{opt} según lo expuesto en la sección 3.2.
9. Se aplica k -medias con centroides iniciales \mathbf{c}_{i-1} sobre J_i para obtener k_{opt} centroides. Se almacena \mathbf{c}_i .
10. Se extraen las trayectorias representativas de cada cluster.

5. Resultados Experimentales



(a) Trayectorias en una iteración (b) Trayectorias representativas obtenidas con $\beta = 5$ (c) Trayectorias representativas obtenidas con $\beta = 7$

Figura 3: Trayectorias extraídas en el conjunto de datos de utilizando una ventana de tiempo de 100 cuadros y $k = 15$

El rendimiento del algoritmo de clustering propuesto fue evaluado sobre un conjunto de trayectorias extraídos de videos de vigilancia reales. El conjunto de datos provisto por [19] consiste en un conjunto de trayectorias obtenidas mediante el seguimiento de personas en el edificio principal de la Escuela de Informática de la Universidad de Edimburgo. Los datos contienen varios meses de observaciones que resultan en más de mil trayectorias observadas por cada día. La figura 3 muestra la influencia del parámetro de escala en la extracción de trayectorias para un instante de tiempo dado. Se puede visualizar como el método permite extraer trayectorias con diferente nivel de detalle.

Uno de los conjuntos de datos provisto por [20] consiste en trayectorias simuladas en una intersección de calles. El algoritmo de agrupamiento fue evaluado utilizando el criterio propuesto en [13] que denominamos SSQ . El criterio consiste en minimizar el promedio de la suma de las distancias al cuadrado de los segmentos pertenecientes a cada cluster. La figura 4 muestra una comparación entre la suma de los errores al cuadrado de los agrupamientos obtenidos mediante el método propuesto y el algoritmo de agrupamiento de trayectorias TRACCLUS [13] para cuatro instantes de tiempo utilizando una ventana de tiempo de 500 iteraciones. En cada instante se muestra la cantidad de segmentos de trayectorias presentes. Comparando con TRACCLUS, el promedio de la suma de las distancias al cuadrado es ligeramente mayor, pero el tiempo de procesamiento del método propuesto es significativamente menor. Para procesar los 5000 puntos en promedio de cada cuadro el método propuesto toma en promedio 0.6 segundos, mientras que TRACCLUS toma alrededor de 8 segundos. Esto se debe a que solo se actualizan los segmentos de trayectorias incorporados o eliminados a la escena y los segmentos cercanos a ellos.

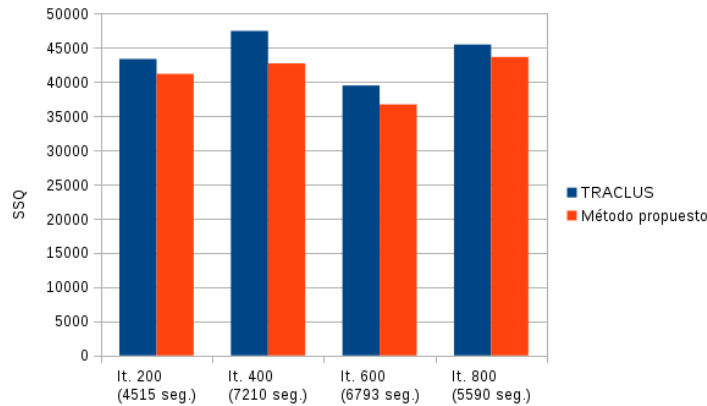


Figura 4: Comparación sobre la efectividad el algoritmo de agrupamiento

6. Conclusiones

En este artículo se presenta un método de agrupamiento incremental de trayectorias obtenidas mediante el seguimiento de objetos en vídeo. Los resultados obtenidos sobre vídeos de vigilancia reales presentan resultados preliminares satisfactorios. El algoritmo incorpora adecuadamente la información provista por los nuevos segmentos de trayectorias mediante la actualización incremental del sistema de autovalores. El método permite obtener las trayectorias representativas de cada cluster en una ventana tiempo. Una etapa futura de este trabajo prevé la incorporación de técnicas de discretización de autovectores con el fin de evitar el paso de aplicar un algoritmo de agrupamiento sobre el espacio generado por los autovectores. Otro aspecto importante sería evaluar la conveniencia de utilizar una medida de semejanza alternativa entre trayectorias.

Referencias

1. B. T. Morris and M. M. Trivedi, "A survey of vision-based trajectory learning and analysis for surveillance," *IEEE transactions on circuits and systems for video technology*, vol. 18, no. 8, pp. 1114–1127, 2008.
2. N. Anjum and A. Cavallaro, "Multifeature object trajectory clustering for video analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1555–1564, 2008.
3. G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.
4. T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *European conference on computer vision*, pp. 282–295, Springer, 2010.
5. M. Vrigkas, V. Karavasilis, C. Nikou, and I. A. Kakadiaris, "Matching mixtures of trajectories for human action recognition," *Computer Vision and Image Understanding*, vol. 19, pp. 27–40, 2014.
6. X. Wang, K. T. Ma, G.-W. Ng, and W. E. L. Grimson, "Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models," *International journal of computer vision*, vol. 95, no. 3, pp. 287–312, 2011.
7. Z. Li, J.-G. Lee, X. Li, and J. Han, "Incremental clustering for trajectories," in *International Conference on Database Systems for Advanced Applications*, pp. 32–46, Springer, 2010.
8. Z. Zhang, K. Huang, T. Tan, P. Yang, and J. Li, "Red-sfa: Relation discovery based slow feature analysis for trajectory clustering," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
9. C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the nyström method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 214–225, February 2004.
10. H. Ning, W. Xu, Y. Chi, Y. Gong, and T. S. Huang, "Incremental spectral clustering by efficiently updating the eigen-system," *Pattern Recognition*, vol. 43, no. 1, pp. 113–127, 2010.
11. C. Dhanjal, R. Gaudel, and S. Cléménçon, "Efficient eigen-updating for spectral graph clustering," *Neurocomputing*, vol. 131, pp. 440–452, 2014.

12. T. Kong, Y. Tian, and H. Shen, "A fast incremental spectral clustering for large data sets," in *2011 12th International Conference on Parallel and Distributed Computing, Applications and Technologies*, pp. 1–5, IEEE, 2011.
13. J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory clustering: a partition-and-group framework," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pp. 593–604, ACM, 2007.
14. J. Shi and J. Malik, "Normalized cuts and image segmentation," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pp. 731–737, Jun 1997.
15. A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, pp. 849–856, MIT Press, 2001.
16. M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
17. L. Zelnik-manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems*, pp. 1601–1608, 2005.
18. U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.
19. B. Majecka, "Statistical models of pedestrian behaviour in the forum,"
20. B. Morris and M. Trivedi, "Learning trajectory patterns by clustering: Experimental studies and comparative evaluation," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 312–319, IEEE, 2009.

Real-Time Estimation of Illumination Direction for Augmented Reality with Low-Cost Sensors

Kevin E. Soulier¹, Matias N. Selzer^{1,2}, and Martín L. Larrea^{1,2}

¹Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur

Av. Alem 1253, B8000CPB Bahía Blanca, ARGENTINA

^{1,2}Laboratorio de Investigación y Desarrollo en Visualización y Computación Gráfica
VyGLab

{kevin.soulier, matias.selzer, mll}@cs.uns.edu.ar

Abstract. In recent years, Augmented Reality has become a very popular topic, both as a research and commercial field. This trend has originated with the use of mobile devices as computational core and display. The appearance of virtual objects and their interaction with the real world is a key element in the success of an Augmented Reality software. A common issue in this type of software is the visual inconsistency between the virtual and real objects due to wrong illumination. Although illumination is a common research topic in Computer Graphics, few studies have been made about real time estimation of illumination direction. In this work we present a low-cost approach to detect the direction of the environment illumination, allowing the illumination of virtual objects according to the real light of the ambient, improving the integration of the scene. Our solution is open-source, based on Arduino hardware and the presented system was developed on Android.

Keywords: Augmented Reality, Illumination, Real-Time, Arduino, Android, Unity3d, Human Computer Interaction.

1 Introduction

Augmented Reality (AR) is the combination of real and virtual elements to augment the real world, improving people's senses and skills. Realistic appearance of virtual objects and their proper interaction with the real world is of high interest in entertainment, design, medicine, education and many other applications areas. In the last years, mobile devices like smartphones and tablets became part of our everyday life. Due to their computational power and integrated camera, they can act like a window into an augmented real world ([1]).

As virtual objects share the same scene with real objects in real lighting conditions, a common issue in AR is the appearance inconsistency between the virtual and real objects due to wrong illumination. The illumination of the virtual objects has to change in the same way as the illumination of the real scene. In order to achieve this consistency, the estimation of the illumination direction is required.

To properly align the simulated virtual object with the real world scene and produce a consistent simulation, an AR system must track the position and orientation of the

camera respecting to the real world. This is typically performed by using AR markers with known shapes or textures ([2]). Recent development using feature extraction and recognition provides a new approach for AR tracking without needing any marker, however, this approach is still under development ([3]).

Tracking and alignment is required to provide overlay information. To improve the user experience, the system must provide a realistic blending between the virtual objects and the real world. The ability to properly simulate the illumination condition of the real world is still limited, mainly due to the difficulty of estimating the illumination conditions of the real environment.

Another important aspect of an AR system is portability ([4]). An AR system with high portability would give the user a greater degree of freedom during the interaction with the augmented environment. Based on the positioning of the display, there are two categories of AR systems with high degrees of portability: head-worn AR and hand-held AR ([5]). From these two categories, the hand-held AR systems are currently considered to be the best for introducing AR to the mass market due to its low production cost and ease of use ([5]). The proliferation of smartphones that could be used as hand-held AR systems also helps the ease of adoption by potential users. Similarly, head-worn AR is a viable option nowadays since there are many virtual reality headsets on the market.

In this work we present a low-cost approach to detect, in real time, the light source origin using portable devices, like Android and Arduino. Our work allows the AR developer to illuminate virtual objects according to the real illumination of the ambient, improving the integration of the scene.

The rest of the paper is organized as follows. Section 2 provides background information about previous works. Then, in Section 3 and 4, we describe the proposed system for real-time calculation of illumination, followed by an experimental application to show the features of the system. We conclude in Section 5 with a brief discussion on limitations and advantages of our approach and future work.

2 Previous Works

Several methods were proposed and developed to estimate the illumination conditions and create coherence between virtual and real objects in an AR system ([6,7,8,9]). Some progress has been made towards detecting the illumination direction by using image processing algorithms ([10,11,12,13,14,15]). Liu *et al.* ([10]) proposed an approach for tracking outdoor illumination variations by analysing feature points extracted from precomputed video frames, based on GPS coordinates and local time. Clements *et al.* ([11]) built a model of the sun-earth system to determine all shadows possible at a given approximate latitude, and compare shadows within a query to those possible under the model to determine illumination and camera orientation. The points of interest of each frame are described manually by user interaction. Jachnik *et al.* ([12]) presented an algorithm for real-time surface light-field capture from a single hand-held camera, which is able to capture dense illumination information from general specular surfaces. They divided the light-field into diffuse and specular components, showing that the specular component can be used for estimation of an environment map. That allowed a convincing placement of an augmented object on a specular surface, with realistic synthesised

shadow, reflection and occlusion of specularities as the viewpoint changes. However, their method is computationally expensive since the intensity of the shadow is calculated as a brute-force approach. Xing *et al.* ([13]) proposed a method to insert virtual objects into a sample photograph of an outdoor scene, simulating the environment illumination and the shadow casting between virtual objects and real scenes. Arief *et al.* ([15]) proposed a method for real-time illumination direction estimation for mobile AR systems, using analysis of shadows produced by a reference object. The method could estimate the direction of a single light source in a controlled environment with a good accuracy. However, the estimation takes within 15 seconds and the method is not good at estimating the distance to the light source. Even though these approaches showed good results, their image processing algorithms are not capable of running in a mid-range smartphone in real-time due to their high computational cost. Furthermore, some of them only work with one frame at time and the user interaction is required to configure or calibrate the system.

Other approaches use special hardware to generate a 3d reconstruction of the scene ([16,17]). Thus, by knowing the location of the objects and the light sources, they can estimate the illumination of the virtual objects. Rohmer *et al.* ([16]) presented a differential illumination method obtaining a consistent illumination of the inserted virtual objects on mobile devices. They use multiple HDR video cameras in a predetermined scenario. Gruber *et al.* ([17]) presented an approach for real-time light estimation and photorealistic rendering in AR. They use a Microsoft Kinect to reconstruct the scene geometry. These approaches still do not work in real-time scenarios and expensive hardware is required.

3 Proposed System

Direction and colour of the illumination source are two important and necessary properties to simulate a proper illumination effect. Moreover, experiments by Slater, Usoh, and Crysanthou in [18] showed that the existence of shadows could improve the spatial perception of the user. We therefore focused on detecting directional illumination sources in order to simulate realistic shadows for a mobile AR system. To achieve this, real-time estimation of the illumination condition of the environment surrounding the mobile device is required.

In this paper we present an illumination direction estimation method to simulate illumination on mobile AR systems in real-time. A low-cost sensor was built in order to detect the illumination of the real environment. Since no image processing method is used, our method is affordable for any smartphone capable of running an AR application. A PC application is required to transmit the information from the sensor to the smartphone.

3.1 Sensing Illumination

In order to estimate the light source direction, we need to estimate the angle of incidence from the light source to a certain point in space. Hence, we can obtain an approximation of the light source rotation and therefore, estimate its direction. A photo-resistor or

light-dependent resistor (LDR) is a very common and low-cost sensor which decreases the current resistance according to the light intensity detected on its surface. A semi-sphere shape sensor was built to sense illumination from different angles. This semi-sphere has an array of 8 LDRs along its surface in order to estimate the incident angle of the light source in the yaw rotation axis, each one separated by 45 degrees. Also, one last LDR is located on the top of the semi-sphere. The location of the LDRs can be seen in Figures 1 and 2. Together with the other LDRs, the angle of incidence of the light source in the pitch rotation axis can be estimated. The sensor is shown in Figure 3.

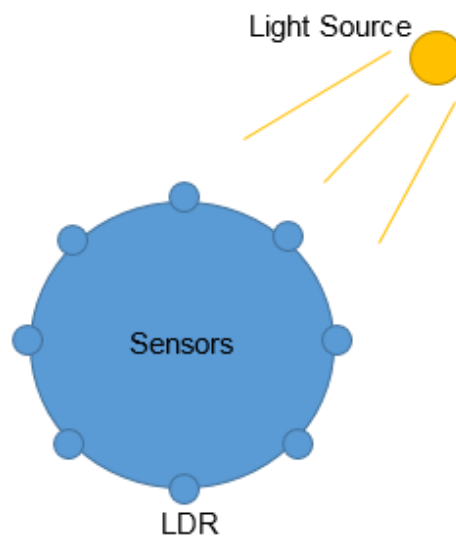


Fig. 1. Sensor top view. The array of eight LDRs is distributed around the sensor, each one separated by 45 degrees.

As Rohmer *et al.* proposed in [16], the computation of the system is distributed between the smartphone application and a PC application. Hence, the calculation of the direction of the light is done in the PC, allowing the smartphone application to focus only on the processing of the AR application. A local wireless communication is established between the smartphone application and the PC application. The PC application calculate the illumination source position in the space, and send that information to the smartphone application. An Arduino UNO Microcontroller¹ was used to gather the information from the sensors, sending it to the PC application.

The yaw rotation of the light source is calculated as follows. First, the eight LDRs around the sensor are numbered from 0 to 7. Then, the LDR detecting more light is considered. For instance, the LDR 3 in Figure 4. As each LDR is separated by 45 degrees, the angle of incidence of the light would be the number of the LDR detecting

¹ <https://www.arduino.cc/>

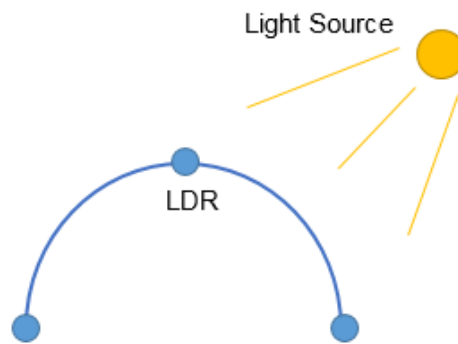


Fig. 2. Sensor sectional view. One last LDR is located on the top of the sensor.



Fig. 3. The semi-sphere shape sensor showing the position of the LDRs.

more light, multiplied by 45 degrees. In Figure 4, the angle of incidence of the light would be 135 degrees.

A second approach is proposed in order to get a better approximation. In this approach, not only the LDR detecting more light is considered, but also the others immediately next to it. Thus, the yaw angle is calculated as an interpolation of those three measures. Similarly, the pitch rotation of the light source is calculated as an interpolation between the LDR detecting more light and the LDR on the top.

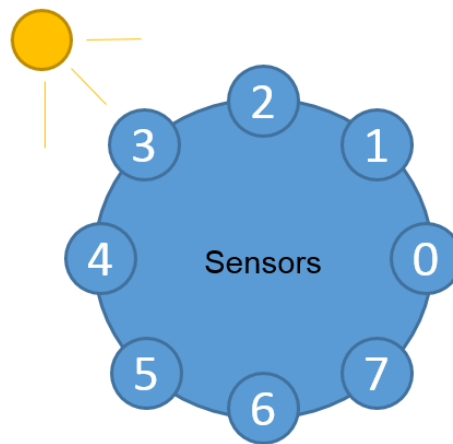


Fig. 4. Illumination example. The LDR 3 is sensing more illumination than the others.

4 Experimental Setup

In order to test our system, an AR application running on an Android device was implemented. Unity3d² with Vuforia³ were used to create the application.

The application consists of a virtual object projected over an AR marker and a virtual directional light source, which rotation depends on the data transmitted by the sensor. The main purpose of the experiment is the comparison between the illumination and shadows projected by a real object and a virtual object. The shadows projected by the real object is produced by the real light source, and the shadow projected by the virtual object is produced by the virtual light source, which orientation depends on the sensed data. Figure 5 shows the experiment where the illumination and shadow of the real and virtual objects can be appreciated. As the position of the light source changes, the illumination and shadows of the virtual object also change in real-time.

² <http://unity3d.com/es/>

³ <http://www.vuforia.com/>

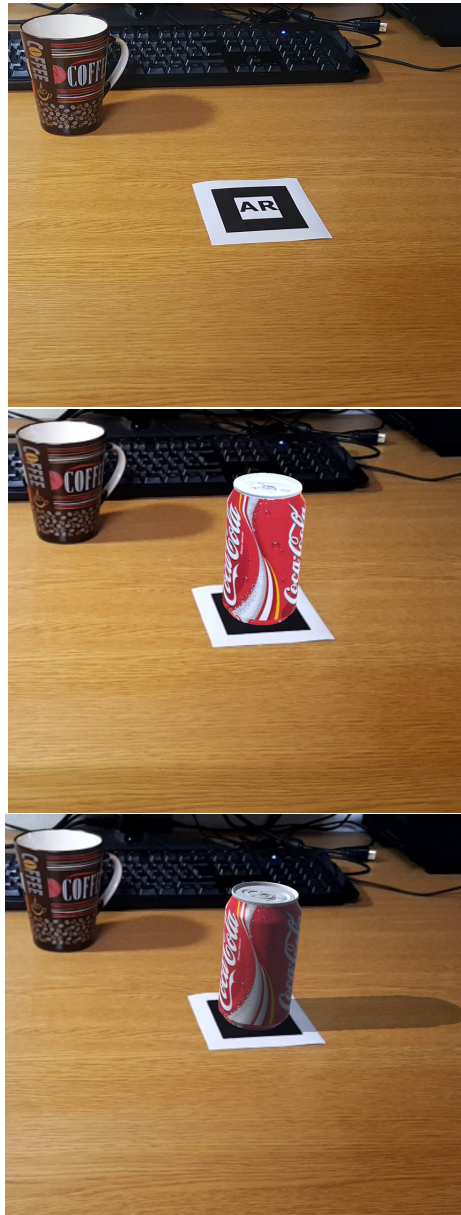


Fig. 5. Complete system working. No virtual object projected (top). Virtual object projected with ambient illumination (middle). Virtual object projected with directional illumination and shadow mapping (bottom).

4.1 Experimental Results

The main purpose of this work was the creation and test of a low-cost sensor to detect the illumination direction of the environment. The gathered information is then used to manipulate a virtual light source in an AR application running on an Android device. In general, our method generated a coherent AR environment, where the virtual illumination resembles the real illumination.

The illumination and shadows comparison between real and virtual illumination can be seen in Figure 5. The results are qualitatively similar to those of earlier studies. For instance, both Liu *et al.* [10] and Arief *et al.* [15] obtained similar results, however they performed image processing algorithms. On the other hand, Gruber *et al.* [17] also obtained good results, but using very expensive hardware.

The results seem to indicate that this approach is useful in any situation, specially where low monetary and computational cost are required, or the computational core of the system is based on smartphones.

5 Conclusions and Future Work

In this paper we have described a solution for real-time estimation of illumination direction for an AR application. Our approach is based on low-cost sensors and Arduino. We presented an experimental test using Android as a platform for our AR software and a PC to transmit the illumination information provided by Arduino.

The appearance of virtual objects and their integration with the real world is fundamental. A visual inconsistency between the virtual and real objects due to wrong illumination negatively affects the appreciation of the scene. As we have seen, few studies have been made about the estimation of illumination direction in real-time with affordable hardware.

Our results are encouraging and future research must include the development of a PC-free sensor, such as a stand-alone sensor or one physically attached to the mobile device. Such device should be able to interact with the AR software without needing any extra hardware. We must continue investigating new hardware in order to achieve a portable sensor, thus maintaining the low cost of the system. Finally, the detection of multiple light sources and the light color and distance were not considered for this work. In order to accomplish this, further work is planned to extend and improve the sensor features.

References

1. G. W. Fitzmaurice, S. Zhai, and M. H. Chignell, "Virtual reality for palmtop computers," *ACM Transactions on Information Systems (TOIS)*, vol. 11, no. 3, pp. 197–218, 1993.
2. X. Zhang, S. Fronz, and N. Navab, "Visual marker detection and decoding in ar systems: A comparative study," in *Proceedings of the 1st International Symposium on Mixed and Augmented Reality*. IEEE Computer Society, 2002, p. 97.
3. D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, and D. Schmalstieg, "Real-time detection and tracking for augmented reality on mobile phones," *IEEE transactions on visualization and computer graphics*, vol. 16, no. 3, pp. 355–368, 2010.

4. R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and virtual environments*, vol. 6, no. 4, pp. 355–385, 1997.
5. D. Van Krevelen and R. Poelman, "A survey of augmented reality technologies, applications and limitations," *International Journal of Virtual Reality*, vol. 9, no. 2, p. 1, 2010.
6. P. Supan and I. Stuppacher, "Interactive image based lighting in augmented reality," in *Central European Seminar on Computer Graphics*, vol. 17, 2006, p. 18.
7. S. DiVerdi and T. Höllerer, "Combining dynamic physical and virtual illumination in augmented reality," *Environment*, vol. 5, p. 12, 2004.
8. K. Jacobs, J.-D. Nahmias, C. Angus, A. Reche, C. Loscos, and A. Steed, "Automatic generation of consistent shadows for augmented reality," in *Proceedings of Graphics Interface 2005*. Canadian Human-Computer Communications Society, 2005, pp. 113–120.
9. J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan, "Estimating natural illumination from a single outdoor image," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 183–190.
10. Y. Liu and X. Granier, "Online tracking of outdoor lighting variations for augmented reality with moving cameras," *IEEE Transactions on visualization and computer graphics*, vol. 18, no. 4, pp. 573–580, 2012.
11. M. Clements and A. Zakhor, "Interactive shadow analysis for camera heading in outdoor images," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 3367–3371.
12. J. Jachnik, R. A. Newcombe, and A. J. Davison, "Real-time surface light-field capture for augmentation of planar specular surfaces," in *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 91–97.
13. G. Xing, X. Zhou, Q. Peng, Y. Liu, and X. Qin, "Lighting simulation of augmented outdoor scene based on a legacy photograph," in *Computer Graphics Forum*, vol. 32, no. 7. Wiley Online Library, 2013, pp. 101–110.
14. H. Kolivand, M. Kolivand, M. S. Sunar, and M. A. M. Arsad, "A fast silhouette detection algorithm for shadow volumes in augmented reality," *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 10, no. 4, pp. 668–672, 2016.
15. I. Arief, S. McCallum, and J. Y. Hardeberg, "Realtime estimation of illumination direction for augmented reality on mobile devices," in *Color and Imaging Conference*, vol. 2012, no. 1. Society for Imaging Science and Technology, 2012, pp. 111–116.
16. K. Rohmer, W. Büschel, R. Dachsel, and T. Grosch, "Interactive near-field illumination for photorealistic augmented reality on mobile devices," in *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*. IEEE, 2014, pp. 29–38.
17. L. Gruber, T. Langlotz, P. Sen, T. Höllerer, and D. Schmalstieg, "Efficient and robust radiance transfer for probeless photorealistic augmented reality," in *2014 IEEE Virtual Reality (VR)*. IEEE, 2014, pp. 15–20.
18. M. Slater, M. Usoh, and Y. Chrysanthou, "The influence of dynamic shadows on presence in immersive virtual environments," in *Virtual Environments '95*. Springer, 1995, pp. 8–21.

Estudo e implementação de técnicas para identificação de anuros em imagem

Angélica Larissa Marques Queiroz¹, César Alberto da Silva², Andrea Padovan Jubileu¹, Linnyer Beatrys Ruiz³,

¹ Instituto Federal de São Paulo (IFSP)
Departamento de Ciência da Computação
Rua José Ramos Júnior 27-50, 19470-000, Presidente Epitácio, SP, Brasil

² Universidade Federal de Minas Gerais (UFMG),
Programa de Pós-Graduação em Engenharia Elétrica
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

³ Universidade Estadual de Maringá (UEM)
Programa de Pós-Graduação em Ciência da Computação
Av. Colombo 5790, 87020-900, Maringá, PR, Brasil
laryssa_here@hotmail.com, casilva91@gmail.com, andreapjubileu@ifsp.edu.br,
linnyer@gmail.com

Abstract. This paper describes the process used in extracting characteristics of Anurans in pictures and it makes a comparison between J48, Random Tree and Random Forest classification algorithms. Two scenarios for evaluation were defined: in the first one we used two classes to represent objects in the image and in the second scenario five classes were used. The Random Forest algorithm obtained best result in the classification of data in relation to J48 algorithms and Random Tree.

Keywords: Frog, Pattern, Recognition, Segmentation, Image.

1 Introdução

Ao longo dos anos o ser humano tem causado danos à natureza, resultando na redução da biodiversidade e da qualidade de vida no mundo. Alguns animais têm sido usados como bioindicadores dessas mudanças [1], como por exemplo os anuros (sapos, rãs e pererecas). Estudar os costumes desses animais no seu ambiente natural é importante para a conservação da natureza. Por exemplo, as transformações no habitat são tão significativas que prejudicam a reprodução desses anfíbios, sendo notória a perda na população.

Atualmente, existem variadas soluções para monitoramento ambiental que utilizam tecnologias de comunicação sem fio. Dentre essas propostas, destaca-se a Rede de Sensores Sem Fios (RSSF), que é considerada pouco invasiva ao ambiente [2]. Essas redes são formadas por centenas ou milhares de nós sensores equipados com sensores para detecção de fenômenos tais como iluminação, umidade do ar, temperatura, etc [3]. As imagens utilizadas neste trabalho foram coletadas por uma Rede de Sensores Multimídia Sem Fios (RSMSF) formada por dispositivos que utilizam

microcontroladores Arduino equipado com câmera OV2640, shield ArduCAM e com o módulo XBee para a transferência das imagens até a estação base. As imagens capturadas possuem dimensão 320x240 pixels. Não faz parte deste trabalho descrever a implementação da RSMSF e sim o processo para identificação de anuros em imagens.

O principal objetivo desse trabalho é estudar técnicas de extração de características dos anuros para reconhecê-los nas imagens. A extração das características é uma etapa crucial no processo de reconhecimento. O reconhecimento de padrões é um processo pertinente à área de Aprendizado de Máquina [4].

Este artigo apresenta o processo para extração de características de anuros em imagens e compara o resultado dos algoritmos de classificação J48, *Random Tree* e *Random Forest* para extração de um padrão para identificar anuros em imagens.

O trabalho está dividido em 5 seções. Na seção 2, é descrita a metodologia de pesquisa utilizada. Na seção 3 são apresentados os trabalhos relacionados ao tema desta pesquisa. Os resultados obtidos são apresentados na seção 4. Por fim, na quinta seção contém as conclusões do trabalho.

2 Metodologia de Pesquisa

O método científico adotado neste trabalho é o indutivo porque a partir do estudo de diversas particularidades, deseja-se chegar a um resultado geral. A partir do estudo dos descritores locais invariantes, dos algoritmos de reconhecimento de objetos, das técnicas de reconhecimento de padrões, dos algoritmos e softwares para o reconhecimento de imagens, pretende-se reconhecer a existência ou não de anuros nas imagens.

Este trabalho pode ser classificado como uma pesquisa aplicada porque pretende implementar uma solução para um problema específico: o reconhecimento de anuros em imagens. Trata-se de uma pesquisa exploratória porque envolve o levantamento bibliográfico, realização de experimentos práticos e análise dos resultados para a compreensão do problema estudado. Além disso, este trabalho visa produzir conhecimentos, adquirir experiência, contribuir para a discussão da área ou fazer avançar o debate acerca dos assuntos abordados.

Considerando que é apresentado o resultado estatístico do percentual de imagens com anuros identificados corretamente com o uso dos algoritmos J48, *Random Tree* e *Random Forest*, esta pesquisa é classificada além de qualitativa também quantitativa.

Para atingir os objetivos, inicialmente foi realizada uma revisão bibliográfica sobre os principais assuntos desta pesquisa: Técnicas de Reconhecimento de Padrões, Visão Computacional, Segmentação de Imagens, Aprendizado de Máquina e Algoritmos de Classificação.

3 Trabalhos Relacionados

Pesquisas para detecção de animais têm sido realizadas a fim de monitorar e rastrear animais para compreender seu comportamento com o ambiente, prevenir

colisões em veículos em rodovias e prevenir a entrada de animais perigosos em área residencial [5]. Nesta seção são apresentados alguns trabalhos que fazem a identificação de outros animais em imagens, uma vez que não foram encontrados na literatura trabalhos que utilizem imagens coletadas por RSMSF para identificação de anuros.

[6] implementaram um algoritmo para segmentação e extração de características para representar as pegadas dos animais. Embora as espécies dos animais utilizadas no experimento não tenham sido divulgadas, os autores utilizaram um conjunto contendo 25 animais com um total de 150 pegadas foi obtido na internet. A identificação da espécie dos animais é baseada no cálculo das características morfológicas de cada pegada. O vetor de características da imagem é formado por: Sete Momentos invariantes de Hu, Proporção do aspecto, Retangularidade, Contraste, Uniformidade, Homogeneidade, Entropia, Proporção de componentes, Área da pegada, Diferença da área, Relação do tamanho do perímetro e *Narrow Factor*. Foram utilizados três fórmulas de distância para classificação das imagens: Distância Euclidiana, *City Block Distance* e *Canberra Distance*. No experimento realizado, além dessas três fórmulas, também foi usada uma combinação entre as três fórmulas. A distância euclidiana apresentou o pior resultado (86%) e as outras obtiveram 97% de taxa de precisão, como mostra a Tabela 1.

Table 1. Taxa de precisão obtida na classificação das espécies de animais [6].

Métodos	Taxa de precisão
Distância euclidiana	86%
City Block Distance	97%
Canberra Distance	97%
Combinado	97%

Um sistema para detecção de cachorros em imagens é apresentado por [5]. Uma câmera captura o vídeo do ambiente monitorado que em seguida é convertido em frames (imagens). Na detecção do objeto é utilizado um método para a remoção do fundo da imagem. Os autores destacam dois desafios nesse processo: mudanças de iluminação e os objetos não estacionários, como por exemplo, chuva, folhas e sombras. De acordo com [7], normalmente, três métodos são utilizados para a detecção do fundo da imagem: *optical flow*, *background subtraction* e *frame differencing*. No trabalho apresentado por [5] foi utilizado o método *frame differencing* que utiliza a diferença de pixels entre frames consecutivos. O experimento foi realizado no Matlab e foram utilizadas 5000 imagens obtidas em várias posições. A eficiência do sistema proposto calculada e validada manualmente por quatro parâmetros: *true positive* (TP), *true negative* (TN), *false positive* (FP) e *false negative* (FN). *True positives* e *true negatives* são classificações corretas. *False positive* é quando o modelo prevê que o resultado deve ser positivo, mas, ao invés disso, o valor obtido é negativo. De modo inverso, *false negative* ocorre quando o modelo prevê um resultado negativo, mas o valor obtido é positivo. A eficiência do algoritmo foi calculada pela equação (1) e obteve 86,25% de taxa de sucesso na detecção do animal.

$$Efficiency = \frac{TN+TP}{TN+TP+FN+FP} \quad (1)$$

[8] desenvolveram um sistema para reconhecimento de espécies de animais baseado no método *eigenface*. Normalmente, este método é utilizado para reconhecimento de face de humanos a partir de um conjunto de características que são relevantes para diferenciar uma face de outra. Os experimentos foram realizados com três tipos de animais: gambás, gatos e doninhas. No primeiro experimento as imagens continham diferentes imagens de fundo e o sistema desenvolvido obteve taxa de sucesso de 55% para gambás, de 33% gatos e de 45% para doninhas. No segundo experimento a imagem de fundo foi retirada e a taxa de sucesso aumentou, obtendo 65%, 52% e 64%, respectivamente para cada espécie de animal.

4 Identificação de anuros em imagens

Os bioindicadores da natureza estão ganhando cada vez mais importância devido ao aumento de trabalhos relacionados com a preservação do meio ambiente e ao grande investimento para monitorar as áreas de conservação [1]. O que pode afetar os estudos nessa área é a limitação ao formular soluções para o problema da visão computacional.

O campo da visão computacional estuda técnicas de extração de características dos objetos de uma imagem, considerando suas formas e outras informações relevantes. Sua diferença em relação ao processamento de imagens é que, enquanto esse refere-se à mudança de uma determinada figura em outras figuras, a partir de modificações inseridas na mesma, a visão computacional trata da obtenção e manipulação das informações de uma imagem e do uso desses dados para diversos fins [9].

Diferentes áreas do conhecimento estão relacionadas à visão computacional. Por exemplo a visão biológica, campo no qual a visão computacional se baseia. Enquanto a biologia estuda as atividades neurais envolvidas na formação e percepção das imagens pelos seres vivos, a visão computacional analisa todos os processos e algoritmos usados pelas máquinas para reconhecer o que os humanos reconhecem [9].

O processo de reconhecimento de um objeto é facilmente realizado pelos humanos porque faz parte das suas atividades cotidianas, como encontrar o automóvel no estacionamento, por exemplo. A partir do momento que o conceito carro é agregado ao universo de aprendizagem de uma pessoa, ela saberá reconhecer a vasta variedade de modelos de veículos, que diferenciam-se em cor, tamanho, entre outras características. Entretanto, é uma tarefa desafiadora em visão computacional [10].

De acordo com [11], algoritmos usados para reconhecer objetos nas imagens, não alcançaram os níveis de acertos de uma criança de dois anos. Segundo [12], para dez mil categorias de imagens, o melhor resultado de um algoritmo até hoje é menor que 5% quando comparado com o desempenho de um humano.

Geralmente, na imagem digital encontra-se um fundo, em companhia de um ou vários objetos. Portanto, os objetos podem ser definidos como regiões que diferenciam-se do fundo. Um caractere, uma silhueta, um animal, uma casa; são exemplos de objetos em uma imagem [13].

A complexidade está em reconhecer os objetos em diferentes cenas, posições e em todas as condições possíveis. A iluminação, por exemplo, afeta a aparência do objeto, dificultando muitas vezes a tarefa de identificá-lo [11].

O principal objetivo do reconhecimento de padrões (ou objetos) é separar os objetos em classes ou categoria, normalmente o sistema de reconhecimento de padrões pode ser dividido basicamente em cinco sub-sistemas: sensoriamento, segmentação, extração de características, classificação e pós-processamento [4].

No entanto, vale ressaltar que são etapas típicas, podendo ser acrescentadas outras no sistema. Neste trabalho, por exemplo, foi efetuado um pré-processamento nas imagens do cenário 1. Na edição, foram adicionados mais brilho e contraste. No cenário 2 foram usadas as imagens originais. A seguir, a Fig. 1 apresenta as etapas do sistema de reconhecimento de padrões realizadas neste trabalho.

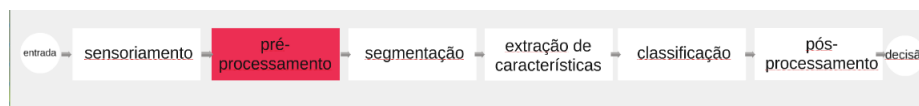


Fig. 1. Etapas realizadas neste trabalho para Identificar Anuros nas Imagens.

No estágio de sensoriamento, um sensor converte um fenômeno físico em um conjunto de dados [4]. Os dados coletados formam as imagens. A Fig. 2 mostra uma imagem de um ranário que foi capturada por uma RSMSF.



Fig. 2. Imagem de Anuros capturada por uma RSMSF.

Segundo [14], a segmentação, ou a etapa de detecção como também é chamada, é um processo importante para um reconhecimento bem sucedido, pois uma falha nesse momento pode comprometer todo o sistema de análise de imagens. A segmentação agrupa os pixels de maneira lógica para simplificar o problema da extração de informações a partir de imagens [15].

Neste trabalho, foi efetuada a segmentação por meio do Software Superpixel. Ele captura a redundância na imagem e reduz significativamente a complexidade das

tarefas subsequentes de processamento de imagens. Têm-se revelado cada vez mais útil para aplicações tais como estimativa de profundidade e segmentação de imagens. Os seus principais benefícios são: sua rapidez, a facilidade na usabilidade e a produção de produtos de alta qualidade de segmentação [16].

A abordagem do algoritmo *Simple Linear Iterative Clustering* (SLIC) é gerar superpixels agrupando pixels com base em sua semelhança de cor e proximidade do plano da imagem. Isto é feito em 5D $[labxy]$, onde $[lab]$ é o vetor de cor do pixel no espaço de cor CIELAB, e xy é a posição de cada pixel [16].

A Fig. 3 apresenta a segmentação através do software Superpixel. Foram usados valores padrões para os parâmetros do algoritmo de geração de superpixel SLIC.

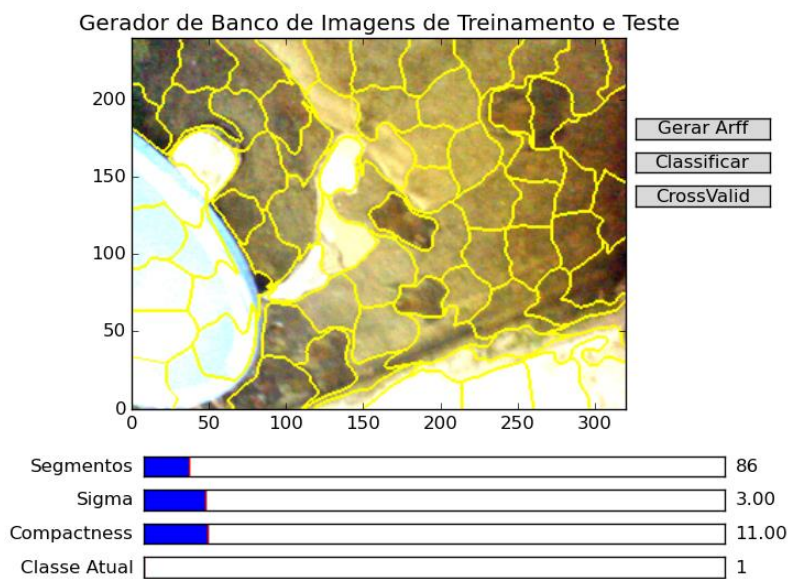


Fig. 3. Segmentação da Imagem através do Software Superpixel.

A possibilidade de compreender e interpretar o conteúdo das imagens é possível através da organização da grande quantidade de informações que a englobam. Normalmente, efetua-se uma análise geral da imagem para extrair uma informação (característica). A maior parte dos extratores de características, no entanto, não interpretam graficamente essas imagens, mas capturam informações numéricas e relevantes para distinguir e dividir os objetos (imagens de interesse), para indicar qual grupo de imagens pertencem à mesma classe [17].

Em cada cenário de teste, foram usadas trinta imagens para a seleção de características. Para classificar, em ambos os casos, foram utilizadas dez imagens.

Neste trabalho, os atributos extraídos dos superpixels foram separados por suas classes. No cenário 2, os atributos, foram separados em cinco classes: a primeira equivalente aos anuros, a segunda a bacia, a terceira ao chão, a quarta a parede e a quinta aos outros atributos que não se encaixam nos anteriores.

A imagem a seguir refere-se ao cenário de Teste 1. A Fig. 4 apresenta a extração de características da classe 1, os anuros, os objetos em questão.

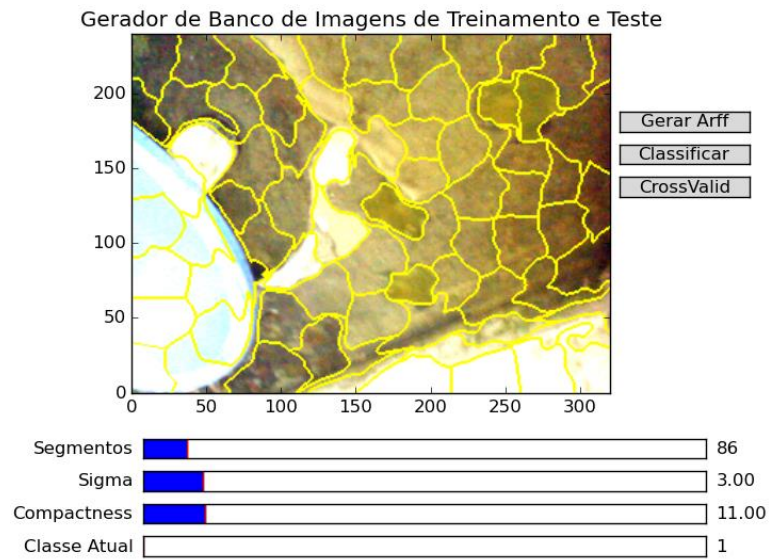


Fig. 4. Extração de Características da Classe 1- Cenário de Teste 1.

A Fig. 5 refere-se ao treinamento do cenário de Teste 2, a extração das características da classe 5.

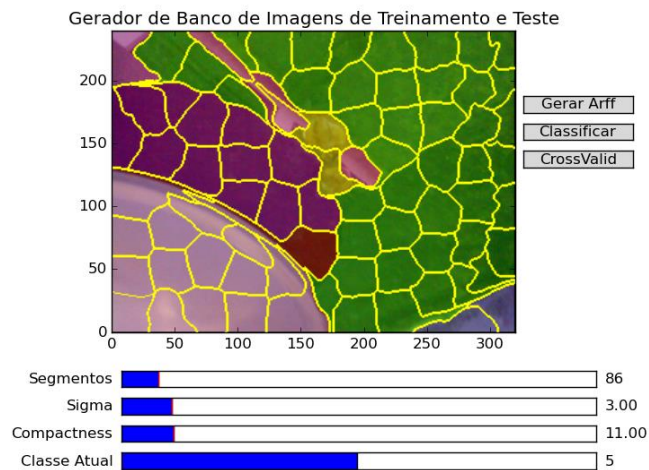


Fig. 5. Extração de Características da Classe 5- Cenário de Teste 2.

Por padrão, o algoritmo de aprendizagem usado pelo Superpixel é o J48. Na Fig. 6 encontra-se uma imagem do conjunto de teste sendo classificada. As características de

cada segmento são armazenadas em um arquivo ARFF que posteriormente é utilizado pelos algoritmos de classificação J48, *Random Tree* e *Random Forest*. Os algoritmos de classificação definem um padrão para classificação dos dados.

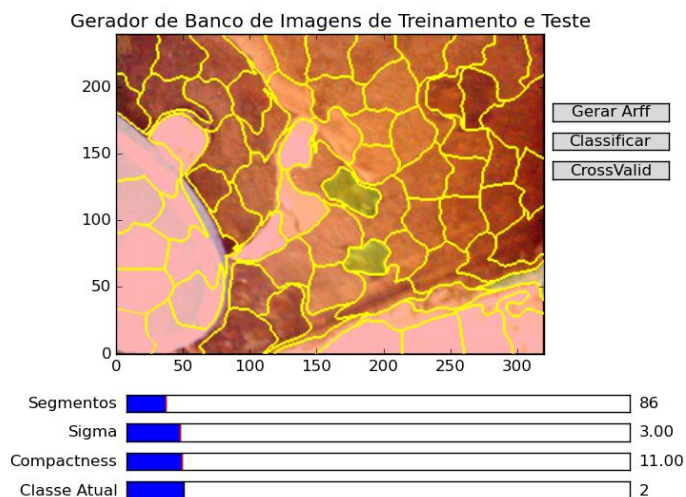


Fig. 6. Classificação pelo Algoritmo J48- Cenário de Teste 1.

No pós-processamento foi realizada a avaliação de desempenho dos diferentes classificadores para tomar a decisão final.

4 Resultados

Os algoritmos de aprendizagem foram construídos a partir dos atributos extraídos dos superpixels separados por suas classes.

No cenário 1, como pode ser observado na Tabela 2, o algoritmo de aprendizagem automática que obteve melhor resultado foi o *Random Forest*, seguido pelo J48 e por fim o *Random Tree*.

Table 2. Métricas de Desempenho para os algoritmos utilizados no Cenário 1.

Algoritmos	Instâncias classificadas corretamente	Instâncias classificadas incorretamente
Random Forest	93,9749%	6,0251%
J48	90,9826%	9,0174%
Random Tree	90,1739%	9,8261%

No cenário 2, podemos notar através da Tabela 3, que o algoritmo de aprendizagem automática que obteve o melhor resultado foi novamente o *Random Forest*, seguido pelo J48 e por fim o *Random Tree*.

Table 3. Métricas de Desempenho para os algoritmos utilizados no Cenário 2.

Algoritmos	Instâncias classificadas corretamente	Instâncias classificadas incorretamente
Random Forest	71,6381%	28,3619%
J48	61,8741%	38,1259%
Random Tree	57,2246%	42,7754%

5 Conclusões

Nota-se que o primeiro cenário comparado ao segundo, obteve maior êxito nas classificações corretas, concluindo-se que para a identificação do objeto, a divisão da imagem em poucas classes obteve melhor resultado. Ao observar os algoritmos *Random Forest* e J48, pode ser visto que estes, em comparação com o *Random Tree*, conseguiram classificar uma boa porcentagem dos anuros corretamente, porém, o restante da imagem está sendo classificado erroneamente, resultando assim com que áreas da imagem que não são anuros sejam classificadas como tais.

Agradecimento. Os autores agradecem o apoio do IFSP e do INCT NAMITEC pelo apoio no desenvolvimento desta pesquisa.

Referências

- Oliveira, M.A.d., Gomes, C.F.F., Pires, E.M., Marinho, C.A.G.S., Della Lucia, T.M.C.: Bioindicadores ambientais: insetos como um instrumento desta avaliação. *Revista Ceres* 61, 800-807 (2014), http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0034-737X2014000700005&nrm=iso
- Rivero, I.A.S.: Rede de Sensores sem Fio para Monitoramento de Equipamentos Eletrônicos. Master's thesis, Curso de Pós-graduação em Informática, Pontifícia Universidade Católica de Minas Gerais, Belo Horizonte, MG (2011)
- Dasgupta, R., Mukherjee, R., Gupta, A.: Congestion avoidance topology in wireless sensor network using karnaugh map. In: *Applications and Innovations in Mobile Computing (AIMoC)*, 2015. pp. 89-96 (Feb 2015)
- Araujo, G.M.: Algoritmo para Reconhecimento de Características Faciais Baseado em Filtros de Correlação. Master's thesis, Curso de Engenharia Elétrica, Universidade Federal do Rio de Janeiro, Rio de Janeiro (2010)
- Sharma, S.; Shah, D.; Bhavsar, R.; Jaiswal, B.; Bamniya, K. Automated detection of animals in context to indian scenario. In: *2014 5th International Conference on Intelligent Systems, Modelling and Simulation*, p. 334–338, Jan (2014)
- Alli, M.N.; Viriri, S. Animal identification based on footprint recognition. In: *2013 International Conference on Adaptive Science and Technology*, p. 1–4, Nov (2013)

7. Fei, M.; Li, J.; Liu, H. Visual tracking based on improved foreground detection and perceptual hashing. In: *Neurocomput.*, v. 152, n. C, p. 413–428, March (2015)
8. Lampert, C.H.; Nickisch, H.; Harmeling, S. Attribute-based classification for zero-shot visual object categorization. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 36, n. 3, p. 453–465, March (2014)
9. Rios, L.R.S.: *Visão computacional*. Universidade Federal da Bahia, Salvador, BA (2011)
10. Graciano, A.B.V.: *Modelagem e reconhecimento de objetos estruturados: uma abordagem estatístico-estrutural*. Ph.D. thesis, Curso de Ciência da Computação, Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, SP (2012)
11. Lara, A.C.: *Descritor de bordas e quantização espacial flexível aplicados a categorização de objetos*. Ph.D. thesis, Curso de Ciência da Computação, Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, SP (2013)
12. Deng, J., Berg, A.C., Li, K., Fei-Fei, L.: *What Does Classifying More Than 10,000 Image Categories Tell Us?*, pp. 71-84. Springer Berlin Heidelberg, Berlin, Heidelberg (2010), http://dx.doi.org/10.1007/978-3-642-15555-0_6
13. López, G.A.P.: *Aforapro: Reconhecimento de Objetos Invariante sob transformações afins*. Master's thesis, Curso de Engenharia Elétrica, Escola Politécnica da Universidade de São Paulo, São Paulo, SP (2011)
14. Farina, A.M.: *BioMobile: Sistema de Identificação de Usuários em Dispositivos Móveis na Plataforma Android Utilizando Reconhecimento de Faces a Partir de Vídeo*. Master's thesis, Curso de Ciências de Computação. Universidade Estadual Paulista Júlio de Mesquita Filho, São José do Rio Preto, SP (2012)
15. Jussiani, V.B.: *Combinação de Atributos de Textura e Cor para Avaliação Automática de Estrias Atróficas*. Master's thesis, Curso de Engenharia de Computação, Universidade Católica Dom Bosco, Campo Grande, MS (2015)
16. Achanta, R., Shaji, A., Smith, K., Lucchi, A, Fua, P., Susstrunk, S.: *Slic superpixels*. Tech. rep., École Polytechnique Fédérale de Lausanne (2010)
17. Mamani, G.E.H.: *Seleção supervisionada de características por ranking para processor consultas por similaridade em imagens médica*. Master's thesis, Curso de Ciência de Computação e Matemática Computacional, Instituto de Ciências Matemáticas e de Computação, São Carlos, SP (2012)

Herramientas para la detección y seguimiento de personas a partir de cámaras de seguridad

Leonardo D. Dominguez^{1,2}, Alejandro J. Perez¹,
Aldo J. Rubiales^{1,3}, Juan P. D'Amato^{1,2}, Rosana Barbuzza^{1,3}.

¹ PLADEMA, Universidad Nacional del Centro de la Provincia de Bueno Aires,

² Consejo Nacional de Investigaciones Científicas y Técnicas, CONICET

³ Comisión de Investigaciones Científicas, CICPBA

Resumen La inseguridad es un problema que afecta en mayor o menor medida a todas las ciudades del mundo. Las ciudades más informatizadas hacen uso de la video-vigilancia para combatirla, montando en muchos de los casos centros de monitoreo con cientos de cámaras. En su mayoría, estos centros cuentan con grupos de personas para realizar la tarea de observación, sin embargo, la velocidad de cómputo actual nos da la posibilidad de automatizar muchas de sus tareas diarias. En este trabajo, se presenta una plataforma de análisis de video que se está desarrollando en la UNCPBA para facilitar el seguimiento de una persona a través de diferentes cámaras, utilizando técnicas de proyección que convierten los puntos detectados desde las diferentes cámaras a un único espacio georeferenciado. Se presenta una discusión de los algoritmos utilizados para el seguimiento, algunos problemas propios que se suceden en este tipo de sistemas y los resultados preliminares obtenidos.

Keywords: Videovigilancia, Tracking multi-camara, Seguridad

1. Introducción

En los últimos tiempos, la seguridad se está convirtiendo en uno de los ejes fundamentales de la sociedad. La gran preocupación que generan los altos índices de violencia e inseguridad actuales [8] en países como Argentina, motivan el estudio de nuevas tecnologías que permitan vivir con mayor tranquilidad a los habitantes, sobre todo, de las grandes ciudades, en donde estas lamentables situaciones son más frecuentes. Según [12], entre el año 2000 y 2008, la tasa de población penitenciaria en América Latina, creció un 42 %, y según las tendencias en los buscadores WEB, en los últimos 10 años el interés de las personas en términos como “Robo”, “Inseguridad o Insecurity” se encuentran en constante aumento. En contrapartida, países como Estados Unidos, cuentan con agencias de investigación para la defensa de sus habitantes. En particular, la agencia DARPA, cuenta con más de 60 proyectos activos, de los cuales nos interesa mencionar a Satellite Remote Listening System y a Combat Zones That See, proyectos que tienen como objetivo registrar todo lo que se mueva mediante un sistema de videocámaras [14]. Otro caso paradigmático es el de Reino Unido,

que según [12], en el informe [4] se estimaba que en Londres un individuo podía ser registrado en un día normal por aproximadamente 300 cámaras. En general, los sistemas actuales están orientados en su mayoría como herramientas de monitorización, también conocidos como CCTV (Closed Circuit Television), que guardan las imágenes en un dispositivo de almacenamiento para su futura visualización o para ser utilizadas como evidencia. En sintonía, los propios fabricantes de cámaras de vigilancia ofrecen herramientas simples, que se limitan a conectar y grabar lo que las cámaras observan. Sin embargo, pensar en un sistema con ciertas automatizaciones, que sea capaz de vigilar un área, emitir y registrar alarmas, clasificar y contar personas e incluso poder seguir sus movimientos si es necesario, nos hace presuponer que será más ventajoso para los operadores, pues les permitirá focalizar su atención y estar alerta ante un evento importante pudiendo así seguir el protocolo que corresponda. Con un propósito similar, se ha iniciado el proyecto [16], el cual pretende a partir de una extensa red de sensores y dispositivos inalámbricos, controlar grandes áreas metropolitanas de forma inteligente. Sin lugar a dudas, con al auge de la tecnología en cuanto a captura de imágenes y video, es posible adquirir cámaras con buena definición a un bajo costo, lo cual permite que una herramienta como esta pueda ser utilizada no sólo en ámbitos de seguridad, sino también en lugares concurridos que requieran analizar los patrones de movilidad diaria, los comercios podrían establecer estrategias de marketing con mayores argumentos [15].

Entendiendo que la seguridad es un tema muy importante, en este trabajo, presentaremos una plataforma distribuida para la automatización de tareas de detección y seguimiento de personas u objetos en diferentes cámaras de video. Para esta tarea es necesario unificar los diferentes puntos de vista de cada cámara, para llevarlo a un único espacio geo-referenciado. Con este objetivo, se han desarrollado herramientas de calibración para obtener las transformaciones que permitan ubicar las detecciones en un mapa. Uno de los algoritmos principales es el que realiza la detección del movimiento, análogo a la substracción de fondo (SF) y detección de objetos. Lo innovador de la plataforma es que se encuentra preparada para poder contar con diferentes algoritmos de manera sencilla y su arquitectura distribuida le permite escalar para soportar muchas cámaras sin afectar el rendimiento global.

El trabajo se presenta de la siguiente forma. En la sección 2, se presentan las herramientas existentes con propósitos similares y un resumen del estado de arte. Posteriormente, en la sección 3 se mencionan dos de los detectores utilizados y cómo se mapean al plano. En la sección 4 se presentan algunos resultados preliminares de los recorridos de una persona junto con las problemáticas encontradas, y en la sección 5 se presentan las conclusiones y los trabajos que quedan pendientes para un futuro.

2. Estado del Arte

Actualmente existen una gran cantidad de sistemas de video vigilancia y en lo que respecta al seguimiento de personas hay mucha bibliografía reciente y es

un tema muy estudiado en estos últimos años, en parte gracias al avance del poder de cómputo disponible en los procesadores actuales.

Trabajos como [5], detallan los algoritmos más comunes para la detección y seguimiento de objetos, y en particular en [7], se realiza una comparación interesante entre los distintos detectores, en donde se concluye que la combinación de detectores puede ser muy útil para disminuir la tasa de falsos positivos mientras se mantiene el tiempo y la tasa de verdaderos positivos.

En el mercado, se destacan soluciones como Blue Iris a un costo de U\$S60 con soporte hasta 64 cámaras o EyeLine a un costo de U\$S250 sin límites de conectividad. Por otra parte se destacan soluciones Open Source como ZoneMinder o Ispy muy evolucionadas aunque todas se limitan a funcionar en un solo equipo. Las características de hardware que recomiendan todas las soluciones son: Intel Core i7, 8 gb de RAM o más, disco SSD, placa aceleradora Nvidia para la decodificación por hardware y Windows 8 o superior.

2.1. Legislación

El esfuerzo de las autoridades por brindar mayor seguridad para sus ciudadanos, puede caer en un reto legal. Hoy en día existen debates sobre la vulnerabilidad de la privacidad por el aumento excesivo de sistemas de vigilancia. En Argentina, todavía nos debemos un gran debate para este tema. Según el artículo [3] publicado en la 44 JAIIO 2015, el 87,5 % de las provincias no cuentan con una regulación adecuada o completa, y sus autores destacan la gran disparidad en el plazo de almacenamiento establecido de las imágenes previo a su destrucción, que va desde los 30 días en Santa Fe, a los 2 años en Corrientes y San Luis. Además, señalan que la mayoría de las regulaciones no se adaptan a los principios establecidos por la disposición 10/2015 de la DNPDP[1].

3. Detección y seguimiento de personas

3.1. Detección de movimiento

El primer paso para el seguimiento de personas, es poder determinar si hubo o no movimiento a partir de una serie de imágenes, lo cual se encuentra muy estudiado. Ya existen múltiples soluciones para poder calcular la variación entre imágenes consecutivas [11][13]. Los métodos varían en complejidad computacional y eficacia del resultado. Cuando las cámaras son exteriores, se suceden otros factores que además varían dependiendo del momento del día, la nubosidad y la época del año. Para contrarrestar estos efectos, se suelen utilizar ventanas de movimiento como se menciona en [10], cuyo objetivo consiste en aplicar una operación lógica, para procesar únicamente los movimientos que se detecten dentro de sus dimensiones y descartar todos aquellos que se produzcan por fuera. En general, la salida de este paso es un conjunto de píxeles (agrupados o no) que han sido marcados como representativos de objetos en movimiento.

Para la plataforma, se utilizaron las capacidades del paquete AForge (open source), con los detectores que incluye para hacer las primeras pruebas del sistema. Observando que todos informan el nivel de movimiento detectado en el video con valores entre 0 y 1, lo cual le permite al programador establecer un umbral para comparar el movimiento y definir diferentes tipos de alarmas, que es uno de los objetivos del proyecto. Hoy en día, se están probando otros algoritmos más sofisticados de sustracción de fondo, tal como VIBE[6], pero que escapan al alcance actual del artículo.

- Two Frame Difference (TF): Es el detector más simple y rápido. Se basa en encontrar la diferencia entre dos fotogramas consecutivos.
- Simple Background Modeling (BM): En contraste con el anterior, este detector se basa en encontrar la diferencia entre el fotograma actual y un fotograma definido que representa el fondo. También existen técnicas para actualizar el fondo a medida que avanza el tiempo.



Figura 1. A izquierda, detección de un objeto con sombra. A derecha, zona de mapeo

Cuando se trabaja con cámaras en el exterior, se suceden algunas situaciones climáticas que afectan el resultado de una detección, como por ejemplo la sombra. Para minimizar el error que provoca la sombra, se propone utilizar el pixel central del objeto detectado. Otros trabajos proponen utilizar la mínima coordenada "Y" del objeto detectado (se considera que es la ubicación de los pies), pero esto no siempre se corresponde en todos los puntos de observacion. Se pretende en un futuro mejorar estos algoritmos, aplicando filtros que primero eliminen la sombra, a partir del análisis de la variación de la iluminación.

3.2. Seguimiento en el mapa

Para poder establecer una relación entre las personas detectadas en cada cámara y el mapa satelital del área vigilada, es necesario calibrar cada una de las cámaras. La salida de este proceso es una matriz homográfica, que permite mapear cada pixel en un punto geo-referenciado. Esta matriz se obtiene de

manera manual, utilizando una serie de herramientas que permiten seleccionar varios puntos de referencia desde la imagen e inferir la matriz correspondiente, denominada HS. Los puntos se eligen sobre una region rectangular y plana.

Esta matriz es luego almacenada en la Base de Datos. Este proceso es sensible al error, por lo que se repite varias veces, calculando la matriz utilizando el promedio de puntos seleccionados. En tiempo real, por cada nueva detección en el espacio de la imagen, se aplica la transformación sobre el punto mas representativo del movimiento, para obtener así un punto sobre el espacio del mapa.

En la Figura 2 se puede observar en forma simplificada el camino recorrido por el proceso de detección y mapeo de una persona en un centro de monitoreo. Primero para lo cual se utilizó la detección de objetos obtenidas en el paso anterior y luego se aplica una transformación espacial sobre un mapa.



Figura 2. Recorrido de una detección hasta su visualización

3.3. Calibración

Dado que una matriz H_s relaciona puntos de dos imágenes, y teniendo en cuenta que se requiere de una diferente para calibrar cada cámara con cada mapa satelital, se implementó una herramienta que le permite al operador cargar dos imágenes (un mapa y un frame de una cámara) y establecer una referencia entre ambas a partir de un mismo cuadrilátero que pueda observarse en las mismas, de modo tal que se pueda calcular la matriz de perspectiva necesaria para adaptarlas.

El proceso de la misma está dividido en 3 solapas (Figura 3), Mapa, Cámara y Resultado. En la primera, se permite cargar el mapa y en la segunda la imagen de la cámara. Luego haciendo clic sobre ambas imagenes se pueden ir definiendo los vértices del cuadrilátero. La herramienta permitirá además, definir el punto donde se posiciona la cámara y calcular las medidas del cuadrilátero (ancho y alto) y su ángulo de rotación α respecto al eje X del satélite que nos provee el mapa, con el propósito de generar no solo la matriz homográfica de la cámara,

sino una matriz que contemple todas las transformaciones necesarias entre ambas imágenes.



Figura 3. Herramienta de calibración. De izq. a der.: Mapa, Cámara y Resultado

3.4. Integración de los datos

La plataforma desarrollada está concebida para escalar a múltiples cámaras, por lo que funciona de manera distribuida. En este sentido, basado en [9], se contempla tener varios equipos de cómputo (computadoras de escritorio o micro-PCs), todos conectados a un servidor central. Cada equipo de cómputo se conecta con una o varias cámaras IP y se encarga del procesamiento. El servidor coordina la detección de movimiento de todas las cámaras, y realiza las operaciones de transformación de los puntos que reciba de los puestos a los mapas. El servidor registra todo en una Base de Datos, la cual luego puede ser accedida por diferentes medios. Un tercer módulo es el encargado de la visualización sobre un mapa. Este módulo puede correr en algunos de los equipos e incluso en el servidor.

Las etapas básicas de este proceso son las siguientes:

- Un equipo 1, establece una conexión con una cámara IP y aplica un algoritmo de SF a las imágenes, indicándole además su estado actual a un servidor S.
- Un equipo 2, se conecta al servidor con la funcionalidad de visualización, suscribiéndose a la recepción de puntos para un mapa que abarca una determinada región satelital.
- Cuando el equipo 1 detecta movimiento, envía al servidor S, un mensaje que contiene el identificador de la cámara procesada y las coordenadas (x,y) .
- Cuando S recibe el mensaje, verifica si algún equipo está suscripto a los mensajes del área que vigila esta cámara. En caso de existir al menos un cliente, S busca la matriz de calibración y transforma el pixel (x,y) del mensaje al espacio del mapa. En caso que se encuentre activado, aplica un pos-procesamiento a la detección para obtener información extra, como por ejemplo una pre-clasificación.

De esta manera, con este pequeño esquema de mensajes, un servidor actuará como concentrador y conocerá en todo momento qué equipos se encuentran procesando señales de video y qué equipos desean recibir las coordenadas transformadas para un mapa.

3.5. Resolución de superposición de multi-cámara y falsos positivos

El área de observación de las cámaras muchas veces se superpone entre sí, por lo que un punto de movimiento puede ser informado por varios equipos a la vez. Al mismo tiempo, se pueden generar falsos positivos, producto de un movimiento de un objeto.

Para unificar estos puntos, y ser tratados como uno solo y reducir los falsos positivos, se debe aplicar un segundo procesamiento de los mismos.

Una de las propuestas que se plantea en este trabajo consiste en ordenar las capturas por intervalos de tiempo. Para que esto funcione correctamente, todos los equipos están sincronizados con un reloj único, para lo cual se utiliza la hora del Servidor. A continuación, por cada conjunto de puntos proyectados dentro de un intervalo, se los promedia (se eligió un intervalo por cada segundo), y luego se calcula la distancia euclídea del grupo con respecto al siguiente intervalo. Aquellos puntos que se encuentran a una distancia mayor de cierta tolerancia son descartados. Estos umbrales se basaron en [2] donde se expresa que en promedio una persona camina a 5 kilómetros por hora (o lo que es igual a 1,38 mts por segundo).

4. Resultados preliminares y discusiones

El seguimiento de una persona incluye dos etapas que se consideran críticas, como lo son la detección del objeto en la imagen de la cámara y el mapeo del mismo hacia el plano. Para las pruebas se utilizó una PC virtualizada de 4 núcleos y 2 GB de RAM con 4 cámaras conectadas y los videos contaban con una resolución de 640x480 pixeles. Las cámaras se ubicaron tal cual se observa en la Figura 4.

En esta configuración, existían varias zonas de oclusión, donde la persona no podía ser observada. Se definieron zonas de interés por cada cámara para obtener el área de análisis relevante. En "blanco", se marcan los puntos por donde pasó una persona.

Para efectuar las pruebas, se aplicó un filtrado de puntos a cada conjunto, en donde se lo agrupó por segundo, promediando las coordenadas x e y. Una vez definidos los conjuntos que contienen una única marca por segundo, se calculó la medida y se promediaron las distancias obtenidas. Se aplicó un filtrado tal como se explicó en la sección 3.5. Conociendo la escala del mapa, se definieron como umbrales máximos de movimiento en un segundo entre 1.38mts y 11.04mts, que corresponden a cuatro veces la velocidad promedio al caminar [2].

Por otra parte, con el fin de comparar la eficacia de los detectores, se calcularon las distancias que existen de sus resultados hasta un recorrido de referencia.

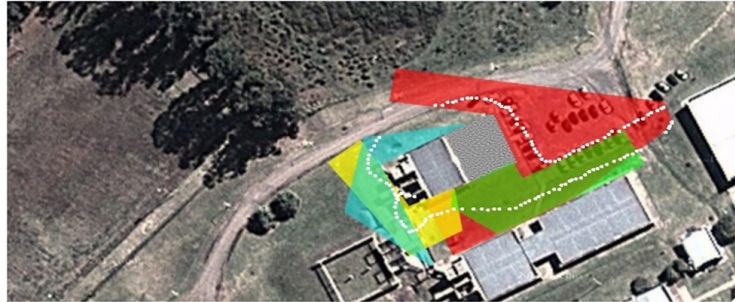


Figura 4. Áreas observadas por las cámaras

Para medir esta distancia, se utilizó el indicador eq.1 que mide como el promedio de las mínimas distancias desde todos los puntos del recorrido original hasta los segmentos consecutivos .

$$\sum_{j=1}^n \frac{1}{n} (\text{Min}_{pb}(\text{Distancia}(pA, pB(j), pB(j + 1)))) \quad (1)$$

donde $pA \in \text{PuntosOriginales}$; $pB \in \text{PuntosDelDetector}$

A su vez, se aplicó un filtro de ganancia del (1% al 99%) a la imagen de diferencia, previo a aplicar un umbral de binarización y obtener los píxeles de movimiento del objeto de interés. En estos casos, una menor ganancia eliminaba píxeles del objeto y una mayor ganancia aumentaba la cantidad. En cualquier caso, se modificaba el centro geométrico del objeto, utilizado posteriormente en la proyección.

Se generó la tabla en la (Figura 5) para analizar estas combinaciones, donde se contabilizaron la cantidad de puntos .

Detector	Umbral = 1.38	Umbral = 2.76	Umbral = 5.52	Umbral = 11.04
B.M. con G=1	1,615	1,598	1,729	1,712
B.M. con G=10	1,610	1,497	1,837	1,765
B.M. con G=30	1,448	1,549	1,678	1,702
B.M. con G=99	1,515	1,747	1,690	1,667
T.F. con G=1	1,561	1,435	1,894	2,204
T.F. con G=10	2,087	2,182	2,357	2,243
T.F. con G=30	1,429	1,914	2,666	2,639
T.F. con G=99	1,301	1,582	2,180	2,086

Figura 5. Continuidad (Cantidad de puntos totales luego de filtrar los puntos)

donde se midió la distancia del trayecto obtenido por cada método respecto al de referencia, para ambos detectores

En todos los casos, se observó la influencia de la sombra, desplazando el centro del objeto. Se observa que cuando el umbral es menor a los 5 mts. la distancia promedio es menor (dado que se descartan falsos positivos lejanos); en otro caso, aparecen puntos que no pertenecen a la persona, por lo que las distancias aumentan.

Los recorridos resultantes antes y después del filtrado fueron los que se ven en la Figura 6

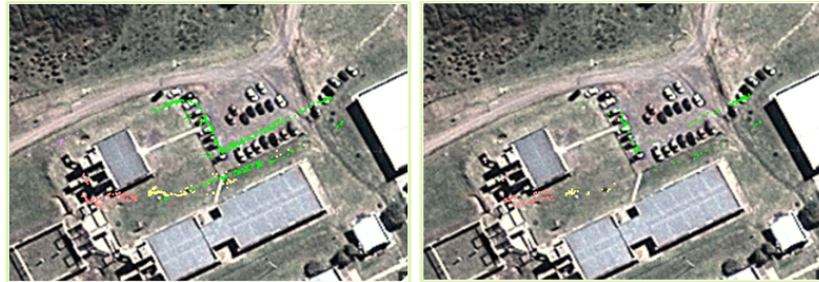


Figura 6. (izq.) Puntos obtenidos originalmente (der) Puntos luego del filtrado

Se observó que gran cantidad de muestras fueron filtradas, muchas producto de falsos movimientos; y los desplazamientos observados se han debido en su mayoría a la proyección de la sombra.

5. Conclusiones

En este trabajo se presentó una plataforma distribuida para la gestión de múltiples cámaras, con el soporte para el seguimiento de personas a través de las mismas. La plataforma permite aplicar diferentes técnicas de substracción de fondo para detectar objetos de interés y luego aplicar una proyección de los puntos, para ser visualizados en un espacio georeferenciado, más amigable para el usuario. La plataforma es muy configurable, permitiendo distribuir la funcionalidad de procesamiento, visualización y seguimiento en cualquier equipo.

También se comenzaron a analizar técnicas de filtrado de los datos, a fin de obtener los puntos de movimiento con mayor precisión. Para esto, se permite ingresar un recorrido de referencia (ya sea de forma manual o por GPS) y compararlo con los resultados obtenidos del análisis de imágenes. Los primeros resultados, no fueron del todo satisfactorios, ya que en un caso ideal deberían ser cercanos a 0; pero dan el pie para seguir profundizando y perfeccionando. Lo que se observó es la influencia de la sombra al momento de la captura, generando perturbaciones en las imágenes, que luego son potenciadas al ser proyectadas.

En un trabajo futuro, se pretende perfeccionar las técnicas de substracción de fondo que contemplen la aparición de sombras, a fin de obtener valores más precisos. También la idea es trabajar sobre el seguimiento de muchas personas de manera simultánea, a la vez que de integrar con otras técnicas de reconocimiento de objetos basadas en Deep-Learning; a fin de optimizar la clasificación de los objetos.

Referencias

1. Dirección nacional de protección de datos personales - disposición 10/2015. <http://www.infoleg.gob.ar/infolegInternet/anexos/240000-244999/243335/norma.htm>.
2. Velocidad promedio del desplazamiento humano. <https://es.wikipedia.org/wiki/Kil>
3. Cejas E. B. and González C. C. Estado de la normativa sobre video vigilancia en argentina y su relación con la protección de datos personales. *44 JAIIO*, 2015.
4. Watch Big Brother. The price of privacy: How local authorities spent £ 515m on cctv in four years. *A Big Brother Watch report, February*, 2012.
5. Legua C. C. Seguimiento automático de objetos en sistemas con múltiples cámaras. 2013.
6. Bouwmans D., Porikli F., B. Höferlin, and Vacavant A. *Background Modeling and Foreground Detection for Video Surveillance*. CRC Press, 2014.
7. Hall D., Nascimento J., Ribeiro P., et al. Comparison of target detection algorithms using adaptive background models. In *Visual Surveillance and Performance Evaluation of Tracking , 2nd Joint IEEE Int. Work.*, pages 113–120. IEEE, 2005.
8. Kessler G. *El sentimiento de inseguridad: sociología del temor al delito*. Siglo Veintiuno Editores, 2009.
9. Foresti G.L., Mähönen P., and Regazzoni C. S. *Multimedia video-based surveillance systems: Requirements, Issues and Solutions*, volume 573. Springer Science & Business Media, 2012.
10. Kruegle H. *CCTV Surveillance: Video practices and technology*. Butterworth-Heinemann, 2011.
11. Shaikh S. H., Saeed K., and Chaki N. *Moving Object Detection Using Background Subtraction*. Springer, 2014.
12. Dammert L., Salazar F., Montt C., and González P. Crimen e inseguridad: indicadores para las américas. *FLACSO-Chile/Banco Interamericano de Desarrollo (BID)*, 2010.
13. Piccardi M. Background subtraction techniques: a review. In *Systems, man and cybernetics, 2004 IEEE international conference on*, volume 4, pages 3099–3104. IEEE, 2004.
14. Shachtman N. Big brother gets a brain. *Village Voice*, 48(28):40, 2003.
15. Chandon P., Hutchinson J., Bradlow E., and Young S. H. Measuring the value of point-of-purchase marketing with commercial eye-tracking data. *INSEAD Business School Research Paper*, (2007/22), 2006.
16. Celtic Telecommunication Solutions. Husims - human situation monitoring system, 2012. https://www.celticplus.eu/wp-content/uploads/2014/09/HuSIMS-leaflet_lq.pdf.

Evaluating the spoken dialogue system of a conversational character: A Simulation Study

Yoselie Alvarado¹, Claudia Gatica², Veronica Gil Costa², and Roberto Guerrero¹

¹Laboratorio de Computación Gráfica (LCG)

²Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)

Universidad Nacional de San Luis,

Ejército de los Andes 950

Tel: 02664 420823, San Luis, Argentina

{ymalvarado, crgatica, vgcosta, rag}@unsl.edu.ar

Abstract. Currently, there are many applications like human-computer interactions in which speech technology plays an important role. In particular, embodied conversational character interfaces research has produced widely divergent results and it has tended to focus on the character's dialog capabilities associated with reasoning. Conversely, the present work attempts to evaluate a conversational character from interaction-related aspects. This paper describes an analysis of a conversational character spoken dialogue system using a discrete event simulator. The simulation model was implemented in the ARENA traditional simulator. Simulation results show that sometimes the response time from the speech conversational character can be too long to user, as well as environment noise is an important aspect of the system to be improved.

Keywords: Virtual Reality (VR), Virtual Humans, Conversational Characters, Natural User Interface (NUI), Speech Interface, Conversational Interaction, ARENA.

1 Introduction

Virtual reality technology has become a very popular technology, which embodies the newest research achievements in the fields of computer technology, computer graphics, sensor technology, ergonomics and human-computer interaction theory. Virtual reality and interactive technology have attracted a great deal of attention and much active research is currently being carried out in an effort to investigate their possible benefits in several areas. It is not a secret: virtual reality technology is an important technology to be paid attention, which will bring huge impact to our life and work [1–4].

New developments in the fields of speech recognition, natural language processing, and computer graphics have given rise to the emergence of more sophisticated computer interfaces with multimodal interaction [5].

On one hand, the evolution in speech technologies allow the development of new systems with several purposes, including voice search, personal digital assistant, gaming, living room interaction systems, and in-vehicle infotainment systems as the most popular applications in this category [6].

On the other hand, embodied conversational character emerged as an specific type of multimodal interface, where the system is represented as a person conveying information to human users via multiple modalities such as voice and hand gestures, and the internal representation is modality-independent, both propositional and non-propositional. Embodied conversational character answers questions and performs tasks through interaction in natural language-style dialogs with users contrasting the traditional view of computers. Many people believe that such interfaces have great potential to be beneficial in human-computer interaction for a number of reasons. Conversational character could act as smart assistants, much like travel agents or investment advisors [7–10].

Conversational character applies rich style of communication that characterizes a human conversation. Researchers have built embodied multimodal interfaces that add dialogue and discourse knowledge to produce more natural conversational characters. For example, *Peedy the parrot* is an embodied character that allows users to verbally command it to play different music tracks. While most research on embodied conversational characters has concentrated on the graphical representation and conversational capabilities of the virtual character, others investigated the question of whether auditory embodiment can provide cues that influence user behaviors and ultimately affects the learning performance of users interacting with a virtual character to learn about biology, mathematics, geography, literature, etc. [11–14].

Over last decade, virtual character has been increasingly used for educational purposes. The rationale behind this emerging trend is the belief that information technology can be utilized as a powerful means to assist learners with the acquisition of general knowledge, literacy, narrative competence, social skills like teamwork and negotiation capabilities, logical and spatial reasoning, eye-hand coordination and fine motor control, among others [15–17].

Conversational character must allow the user (speaker) to watch for feedback and turn requests, while the character (listener) can send these at any time through various modalities. The interface should be flexible enough to track these different threads of communication in the appropriate way to each thread. Different threads have different response time requirements; some, such as feedback and interruption occur on a sub-second timescale [18].

It is evident that human-computer communication is extremely complex just as human-human communication. Its analysis implies to turn to several disciplines like as psychology, sociology, biology, among others for foundation of computational model. These disciplines provide very precious qualitative and quantitative information that are indispensable to consider. Moreover, the evaluation of single modalities from conversational character often can be useful for system performance analysis. In this case, simulation models are a powerful

tool for both understand the system behavior and to detect possible bottlenecks [19–22].

Thus, as the conversational interaction is a vital aspect for an embodied conversational character, in the present work we present a simulation approach as a real-time modeling for the speech modality.

2 Conversational Character System

The spoken dialogue subsystem of a semantic conversational character is the system simulated. The *character* feature means that system’s interface has a visual embodiment and the *conversational* feature means that the system simulates a conversational skill like a human being so the system shows listening and speaking ability. Finally, the conversational character is *semantic* because it is able to perform reasoning through a query to a *Dbpedia*’s query module for users’ answers [23, 24].

More specifically, the implemented approach is a question-answering virtual character driving responses to user’s inquiries (See fig. 1). The system, named CAVE-VOX, is thought to be used in inquiry applications through the analysis of the user’s keyword natural language utterance and the generation of the appropriate response; previous works describe the whole system [25, 26].



Fig. 1. CAVE-VOX system.

System’s human-computer interaction involves real-time synchronization of several aspects like visualization, speech recognition, keywords detection, speech synthesis, among others. Thus a dialogue system between user and virtual character that exploits knowledge provided by structured data (ontologies) in order

to help users in an specific information search was built. Then, this question-answer character has a collection of responses relevant to a particular topic. The driven query must be an specific query about a previous chosen topic. The character plays an appropriate role according with the selected topic and answers to user's queries through data stored in a *Dbpedia* data base.

2.1 Speech system

In this work, the spoken dialogue subsystem of a semantic conversational character was simulated. As the real-time interaction is important for this analysis the conversational module was only considered, this system's modality involves simulating inherent human skills like as listening and talks.

The interaction between the character and the user is performed in real-time through user's voice (by speech to text conversion) and virtual character's voice (by text to speech translation), adding simple gestures and expressions, and lip synchronization.

In order to make the evaluation and considering the original system, the main involved components are:

- *Speech recognition (listening)*: while interacting, a human user talks to the system, which transforms the user's speech to a textual representation by using an **Automatic Speech Recognition**(ASR) module. The module takes as input the user's speech utterance that comes from microphone and gives a resultant text from parsing the word string produced by speech recognition and forms an internal semantic representation based in keywords contained into a grammar.

Each word said by the user is analyzed for the recognizer and compared through a phonetic transcriber. All the picked up words by the microphone are contrasted with words defined by rules in a grammar. According with the grammar, possible sentences are analyzed and compared with the received sentence, then every word gets a confidence value. The ASR module takes this confidence value to determine if a word is accepted or rejected.

If the word is accepted, then it is stored as a valid keyword. If the word is rejected, then the system must inform about the failure to user. From previous analysis with users, it is known that the rejection rate is approximately 50%.

When all the words are accepted it allows for a query as a text string consisting of keywords.

Because the process performed by the ASR is completely computational and is performed in parallel, it is not possible for the user to know the time required to execute this operation and it is usually represented in nanoseconds. Therefore, for the simulations performed in this work, we consider that the ASR processing time is 0 seconds.

- *Reasoning (thinking)*: this module generates a response based on the input, the current state of the conversation and the dialog history. For this, it

extracts the meaning of the utterance from keywords, manages the dialog flow and produces the appropriate actions for the target domain in on-line *Dbpedia*.

This approach gives complete control over the virtual persona's knowledge and expressions to the scriptwriter who creates the responses. It allows the writer to specify the character of the virtual persona, what information it can deliver and the form of that delivery. When an interactor comes up to the conversational character and asks it a question, the system driving the character analyzes the interactor's question and selects the appropriate response from *Dbpedia* collection.

Finding an answer takes between 10 and 120 seconds according with the user's query. If this process faults, then the system notifies to the user about the fault. From previous analysis, it is known that the fault rate is approximately 10%.

- *Speech synthesis (talking)*: It is a **Text-To-Speech** (TTS) module carrying out the generation of the synthetic output voice from the text that comes as a response from the Reasoning Module. This module is performed on three main situations: if one word is rejected, then the conversational character says "*Excuse me, I didn't hear you very well*" which takes about 3 seconds, if the *Dbpedia* query faults, then character says "*Sorry. It's not possible to retrieval the elements you requested*" which takes about 7 seconds, and if the conversational character has an answer from *Dbpedia*, then the delay is between 40 and 300 seconds according with its length.

3 Modeling and Simulation with ARENA

Modeling and simulation provide the basis for efficient solving of various problems related to the operation of complex systems like analysis, optimization and management, industry problems (like mining process, etc.). Simulation is considered to be one of the most effective technologies for the analysis and planning of logistics systems [27].

ARENA simulation software, is a general propose simulation tool enabling the construction of models over a series of modules or basic components organized hierarchically. The ARENA simulation software has high level of modeling supporting graphical design. It also includes a lower level of modeling including specific details as arrival times, service time, scheduling of processes, etc. [28].

A model is developed using modules that are part of the basic processes. In ARENA, modules are the flowchart and data objects that define the process to be simulated. All information required to simulate a process is stored in modules. The dynamics associated with the processes can be viewed as nodes in a network by which entities circulate causing a change in the system state. The entities with attributes and variables compete for the services provided by the resources. Entities are items (like trucks, mineral, etc.) that are being served or produced [29].

Figure 2 shows a simple model build with ARENA. The CREATE module is the starting point for entities in a simulation model. Entities are created using a schedule or based on a time between arrivals. Entities then leave the module to begin processing through the system. The entity type is specified in this module. The PROCESS modules are intended as the main processing method in the simulation. They include the resource by which entities compete. A resource retained by an entity must be released at some point in the model. Otherwise, a deadlock can occur. The DECIDE module allows for decision-making processes in the system. Finally, the DISPOSE modules are ending point for entities in a simulation model [30].

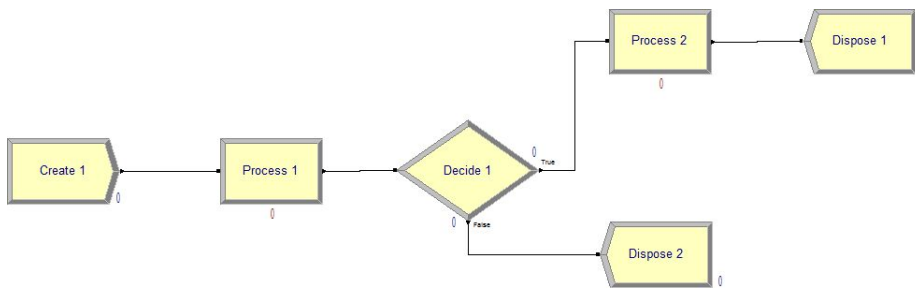


Fig. 2. Basic modules used in ARENA.

After the model is built, we can run simulations to obtain different metrics and statistics like resources utilization, waiting time, etc.

4 Conversational Character Modeling

The simulation model covers a subset of a conversational character. ARENA v.10 was used for modeling the real-time simulation of the spoken dialogue system. As ARENA is an entity-driven application, only the tasks that directly impact the entities are modeled. In the model, an entity represents a query made by the user.

Our resulting model contains the three basic components of a conversational system:

- *Speech recognition (listening)*: this system's stage modeling uses a CREATE module which creates entities (queries based on a time between arrivals), a PROCESS module intended to represent the process performed by ASR resource (because it is not possible for the user to know the time required to execute this operation then the simulated time-delay is zero), and a DECIDE module representing the process which decide as to whether or not something heard by the recognizer is a valid word (according with this problem the success rate is approximately 50%).

- *Reasoning (thinking)*: this component is modeled with a PROCESS module representing keyword searching in the *Dbpedia* resource (with a uniform delay between 10 and 120 seconds), and a DECIDE module determining as to whether or not a keyword is in *Dbpedia* (according with this problem the success rate is approximately 90%).
- *Speech synthesis (talking)*: it is represented by means of three different PROCESS modules according with the three main established situations (3 seconds, 7 seconds and a uniform delay between 40 and 300 seconds respectively). All of these processes use TTS resource and all of them complete at DISPOSE module.

Considering the system’s components described before, our simulation model is showed in figure 3.

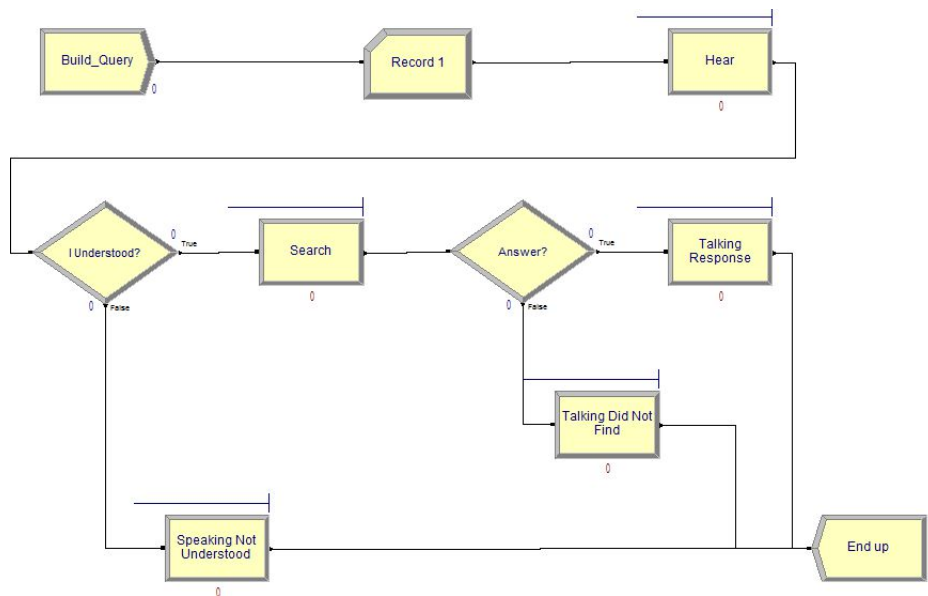


Fig. 3. Conversational Character modeling in ARENA.

5 System Evaluation

Once the simulation model of the conversational character has been built and verified, it can be used to analyze the system performance. After the simulation’s running, the ARENA simulator recorded several data related to the model parameters. ARENA reports include category by replication, entity times, entity number, queue times, resource usage, among others.

In this case, 20 replications were performed. From every replication, information about entities times and resources usage was collected: entity parameter represents the query from user and resources parameters are ASR, TTS and *Dbpedia* tools.

The simulation model generates outputs including the performance measure used in the users' experiments considering the process realized by the conversational character from the user's utterance to the appropriate response synthesized by the subsystem for a particular situation.

Query is a the unique entity in the simulated system. In this system the query analysis is vital because a query time allows to learn about response times of the system. The analyzed time for the entities (query) is from the query creation to the query output. An average time from all repeats was considered to obtain minimum, maximum and average times of entity. Table 1 shows the resulting values in seconds.

	Minimum	Maximum	Average
Value	3,0	359,2	103,4

Table 1. Times entity: Query.

From these values, it is concluded that the processing time necessary to attend the demand of a user is 103 seconds, meaning near 2 minutes in average, and the maximum time is 359 seconds meaning near 6 minutes. Minimum value represents the case when speech recognition fails.

In analysis of resources, resource usage was considered. ASR and TTS resources are used in each query only once, then their constant values are not important for system's performance analysis. This is different from *Dbpedia* resource where its analysis requires to be considered. According with ARENA outputs, *Dbpedia* usage is about 30% as a minimum, 80% as a maximum and 42,3% in average. It is important to emphasize that if *Dbpedia* resource is not used this is due to speech recognition failure.

6 Conclusions and Discussion

Latest research had been oriented to create embodied computer-animated characters that produce accurate auditory and visible speech, as well as realistic facial expressions, emotions, and gestures. The invention of such characters has a tremendous potential to benefit virtually all individuals in natural user interface. Conversational interaction is an important aspect to enable and improve the human-computer interaction.

This paper involved the evaluation of the spoken dialogue system of a conversational character with a discrete event simulator ARENA.

We described character's speech system and its simulation model that was developed to study conversational interaction performance. We introduced the general and most important issues one has to take care of when starting to

simulate with simulator ARENA. We discussed the modules and parameters in detail with the main idea to obtain quantitative results from simulation.

At the moment, analysis is limited to entities and resources parameters. Even though, the analyses does not cover all the aspects of the user query, we believe it has several elements to improve. For example, from entity point of view the simulation results show that sometimes the extensiveness of the speech from the conversational character can be too long to users. A more exhaustive study is need for determining if a long answer is produced by reasoning or something else. For the analysis of resources, *Dbpedia* usage is not ideal because sometimes speech recognition fails due to many factors influence in a correct recognition of speech as: speaker's clarity, microphone's quality, user's accent, environment noise, etc. Several of these aspects can be improved by upgrading microphone's quality.

In this work, we focused on human-computer conversational interaction in real-time and how to do it more natural and intuitive to users allowing proficient user-interactivity in real-time, meaningful feedback and learning through an interface.

References

1. XinXing Tang, editor. *Virtual Reality - Human Computer Interaction*. InTech, 2012.
2. M. Chan. *Virtual Reality: Representations in Contemporary Media*. Bloomsbury Publishing, 2014.
3. T. Parisi. *Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile*. O'Reilly Media, Incorporated, 2015.
4. Kurt Squire. Changing the game: What happens when video games enter the classroom? *Innovate: Journal of Online Education*, 1(6), August 2005.
5. Cecilia Sik Lanyi, editor. *The Thousand Faces of Virtual Reality*. InTech, 2014.
6. Dong Yu and Li Deng. *Automatic Speech Recognition: A Deep Learning Approach*. Springer Publishing Company, Incorporated, 2014.
7. J. Cassell, T. Bickmore, L. Campbell, H. Vilhjálmsón, and H. Yan. Embodied conversational agents. chapter Human Conversation As a System Framework: Designing Embodied Conversational Agents, pages 29–63. MIT Press, Cambridge, MA, USA, 2000.
8. M. Mancini. *Multimodal Distinctive Behavior for Expressive Embodied Conversational Agents*. Universal Publishers, 2008.
9. Q. Chen, P. Torroni, S. Villata, J. Hsu, and A. Omicini. *PRIMA 2015: Principles and Practice of Multi-Agent Systems: 18th International Conference, Bertinoro, Italy, October 26-30, 2015, Proceedings*. Lecture Notes in Computer Science. Springer International Publishing, 2015.
10. B. Endrass. *Cultural Diversity for Virtual Characters: Investigating Behavioral Aspects across Cultures*. EBL-Schweitzer. Springer Fachmedien Wiesbaden, 2014.
11. Niels Ole Bernsen and Laila Dybkjr. *Multimodal Usability*. Springer Publishing Company, Incorporated, 1st edition, 2009.
12. Gene Ball, Dan Ling, David Kurlander, John Miller, David Pugh, Tim Skelly, Andy Stankosky, David Thiel, Maarten V Dantzich, and Trace Wax. Life-like computer characters: The persona project at microsoft research. *Software agents*, pages 191–222, 1997.

13. Sharon Oviatt, Courtney Darves, and Rachel Coulston. Toward adaptive conversational interfaces: Modeling speech convergence with animated personas. *ACM Trans. Comput.-Hum. Interact.*, 11(3):300–328, September 2004.
14. R.J.S. Sloan. *Virtual Character Design for Games and Interactive Media*. CRC Press, 2015.
15. Andrea Corradini, Klaus Robering, and Manish Mehta. *Conversational characters that support interactive play and learning for children*. INTECH Open Access Publisher, 2009.
16. David Griol, José M Molina, Zoraida Callejas, and Ramón López-Cózar. La plataforma educagent: agentes conversacionales inteligentes y entornos virtuales aplicados a la docencia. 2011.
17. Marissa Milne, Martin Luerksen, Trent Lewis, Richard Leibbrandt, and David Powers. Embodied conversational agents for education in autism. *A comprehensive Book on Autism Spectrum Disorders*, page 387, 2011.
18. Dominic W Massaro, Ying Liu, Trevor H Chen, and Charles Perfetti. A multilingual embodied conversational agent for tutoring speech and language learning. In *INTERSPEECH*, 2006.
19. J.O. Turner, M. Nixon, U. Bernardet, and S. DiPaola. *Integrating Cognitive Architectures into Virtual Character Design*. Advances in Computational Intelligence and Robotics. IGI Global, 2016.
20. Y. Zhang, P. Zhang, and D.F. Galletta. *Human-computer Interaction and Management Information Systems: Foundations*. Taylor & Francis, 2015.
21. Benjamin Weiss, Ina Wechsung, Christine Kühnel, and Sebastian Möller. Evaluating embodied conversational agents in multimodal interfaces. *Computational Cognitive Science*, 1(1):1, 2015.
22. Lee W Lacy. *Interchanging Discrete event simulation Process Interaction Models using the Web Ontology Language-OWL*. PhD thesis, University of Central Florida Orlando, Florida, 2006.
23. Yu-Lin Chu and Tsai-Yen Li. *Realizing semantic virtual environments with ontology and pluggable procedures*. INTECH Open Access Publisher, 2012.
24. Mohamed Morsey, Jens Lehmann, Sören Auer, Claus Stadler, and Sebastian Hellmann. DBpedia and the Live Extraction of Structured Data from Wikipedia. *Program: electronic library and information systems*, 46:27, 2012.
25. Y. Alvarado, N. Moyano, D. Quiroga, J. Fernández, and R. Guerrero. *Augmented Virtual Realities for Social Developments. Experiences between Europe and Latin America*, chapter A Virtual Reality Computing Platform for Real Time 3D Visualization, pages 214–231. Universidad de Belgrano, 2014.
26. N. Jofré, G. Rodríguez, Y. Alvarado, J. Fernández, and R. Guerrero. Virtual humans: Conversational characters for a cave-like environment. In *XX Congreso argentino de ciencias de la computación*, pages 937–946. Universidad Nacional de La Matanza, Octubre 2014.
27. Mustafa Rawat and Steven Vaccaro. Implementing a distributed logistics simulation using arena and hla. In *Proceedings of the 2009 Spring Simulation Multiconference*, page 62. Society for Computer Simulation International, 2009.
28. T. Altiok and B. Melamed. *Simulation Modeling and Analysis with ARENA*. Elsevier Science, 2010.
29. M.D. Rossetti. *Simulation Modeling and Arena*. Wiley Series in Modeling and Simulation Series. Wiley, 2015.
30. W.D. Kelton, R.P. Sadowski, and N.B. Swets. *Simulation with Arena*. McGraw-Hill Education, 2015.

Un nuevo descriptor para la identificación de personas mediante caracteres simples

Verónica Aubin(1), Marco Mora(2), and Matilde Santos(3)

(1)Universidad Nacional de La Matanza, Argentina

(2)Universidad Católica del Maule, Chile

(3)Universidad Complutense de Madrid, España

vaubin@unlam.edu.ar, mora@spock.ucm.cl, msantos@ucm.es

Abstract. En este artículo se presenta un nuevo descriptor para la verificación de la identidad de personas en base al análisis de imágenes en escala de grises de caracteres manuscritos individuales y simples. El descriptor corresponde a los coeficientes B-Spline de la curva de posición relativa de los puntos de mínimo valor de gris dentro del carácter. Estos puntos corresponden a los píxeles de menor valor de gris sobre la línea recta perpendicular a los puntos del esqueleto morfológico del trazo. La posición relativa se computa como la distancia euclidia entre el punto de mínimo gris y su correspondiente en el esqueleto morfológico del trazo. Se utilizó un clasificador multiclase, basado en Máquinas de Vectores Soporte de salida binaria, para evaluar la capacidad de discriminación del descriptor propuesto. Se utilizó una base de datos con 50 muestras de 6 símbolos simples realizadas por 50 personas. La experimentación de la base de datos muestra resultados muy satisfactorios, con un promedio de aciertos del 97%, y permiten pensar que es factible desarrollar un método de identificación de personas en base al descriptor presentado.

Keywords: Reconocimiento de personas, Análisis off-line, Trazos manuscritos, Puntos de mayor presión.

1. Introduction

La escritura manuscrita como patrón biométrico de comportamiento ha cobrado un renovado interés por parte de los investigadores en los últimos años [1]. A pesar de avanzar hacia el sueño de un futuro puramente digital, el estudio de la escritura usando imágenes digitales ha conservado su lugar debido a sus aplicaciones en la vida real: en el análisis forense de documentos, la clasificación de los archivos históricos [2], la verificación de firmas [3], estudios sobre la correlación entre la escritura y diferentes trastornos neurológicos [4], así también como en diferentes campos como la seguridad y control de acceso [5] [6].

La escritura a mano es una de las técnicas biométricas primitivamente utilizadas para autenticar un individuo. Los trazos manuscritos de una persona tienen una gran variabilidad; no obstante, existen características invariantes que permiten el reconocimiento manual o automático del autor. En el modo verificación, el

sistema valida la identidad de una persona comparando el rasgo biométrico capturado en la entrada con su propia plantilla biométrica previamente almacenada en la base de datos. En general, el usuario indicará su identidad mediante un número de identificación personal, un nombre de usuario o algún tipo de código. Posteriormente el sistema realizará una comparación uno a uno para determinar si el individuo es quien dice ser.

Muchos de los trabajos relacionados con esta investigación provienen del problema de la verificación de firmas. La verificación off line de firmas es un tema bien documentado y se ha abordado con muchos enfoques diferentes. Una serie de documentos cubren los avances en este campo [7] [3]. Recientemente en [8] se presenta un estudio de la literatura, hasta la fecha, sobre la identificación y verificación off line del autor de escrituras a mano considerando las distintas características y enfoques de clasificación, en diferentes idiomas y alfabetos. En [9] se presenta un análisis de imágenes off-line de escritura a mano basado en textura para la identificación del escritor; cercana a nuestra propuesta. La técnica que usan divide una letra en fragmentos pequeños y considera cada fragmento como una textura. En [10] se consideran caracteres bengalí aislados para la verificación del escritor. Se presenta el rendimiento de diferentes características texturales. Una colección de 500 documentos en bengalí de 100 escritores se utilizan en este sentido. La combinación de características obtiene un mejor rendimiento. La evaluación de los resultados muestra que el método es efectivo y se puede aplicar en la base de datos de gran tamaño. En [11] se plantea la identificación de personas a través de características dinámicas extraídas de la imagen en escala de grises de un carácter. El descriptor presentado considera el patrón de la distribución de presión a lo largo del trazo que es discriminante entre individuos, es decir considera el nivel de gris de los píxeles del trazo y la ubicación en el mismo. Se encontró que, la distancia relativa, medida en una perpendicular al esqueleto del trazo, entre la línea que une los puntos más oscuros de la imagen respecto del esqueleto, es característica del autor, y varía de un individuo a otro.

Al momento de implementar un sistema biométrico deben considerarse aspectos prácticos, tales como el rendimiento computacional y el espacio de almacenamiento. El uso de técnicas de selección de características son necesarias para reducir el espacio de almacenamiento y mejorar el tiempo de compilación.

En este artículo se presenta un nuevo descriptor para la identificación de personas, basado en la extracción de características pseudo-dinámicas de imágenes en escala de grises de caracteres manuscritos individuales y simples. El descriptor corresponde a los coeficientes B-Spline de la curva de posición relativa de los puntos de mínimo valor de gris dentro del carácter. Estos puntos corresponden a los píxeles de menor valor de gris sobre la línea recta perpendicular a los puntos del esqueleto morfológico del trazo. La posición relativa se computa como la distancia euclidia entre el punto de mínimo gris y su correspondiente en el esqueleto morfológico del trazo. Es de destacar la importancia de que el descriptor propuesto extrae características de un trazo simple, en lugar de un trazo complejo como puede ser la firma de un individuo, lo que permite tomar muchas muestras de un mismo documento. Además de lo anterior, el descriptor presenta

la particularidad de ser invariante a la rotación, lo que aporta una ventaja importante dentro del procesamiento de imágenes. La clasificación se realizó con una SVM multiclase, contemplando el problema de clases desbalanceadas, para evaluar la capacidad de discriminación del descriptor propuesto. Se analizan los resultados obtenidos de la clasificación para las muestras realizadas por 50 personas, correspondientes a la base de datos creada especialmente con 6 símbolos simples.

La estructura de este trabajo es la siguiente. La sección 2 presenta conceptos de aproximación B-spline aplicados en el trabajo. La sección 3 presenta el descriptor propuesto. La sección 4 explica en detalle la metodología empleada para la validación de los resultados y la base de datos creada especialmente. La sección 5 presenta los resultados de la evaluación objetiva del descriptor. Finalmente, en la sección 6, se presentan las conclusiones del trabajo.

2. Conceptos de aproximación B-spline

La teoría de las B-spline es muy amplia y se expone en detalle en [12]. En ésta sección sólo se desarrollan los elementos necesarios de dicha teoría para la comprensión del uso de las B-spline en esta investigación.

La B-spline es una combinación lineal de $n+1$ puntos de control $c_1, \dots, c_n, n \geq k$, con funciones bases $N_{i,k}$. Estas funciones son polinomios formados a partir del parámetro u tal como se muestra en la siguiente expresión:

$$f(u) = \sum_{i=0}^L c_i N_{i,k}(u) \quad (1)$$

siendo k el grado de la curva B-spline y L se corresponde con el número de segmentos en que se divide el espacio de aproximación.

Las bases B-spline $N_{i,k}$ pueden definirse según la siguiente expresión recursiva:

$$N_{i,j}(t) = \frac{t - t_i}{t_{i+j} - t_i} N_{i,j-1}(t) + \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}} N_{i+1,j-1}(t) \quad (2)$$

donde el corte de la recursividad corresponde a la expresión

$$N_{i,0}(t) = \begin{cases} 1 & t_i \leq t \leq t_{i+1} \\ 0 & \text{otros casos} \end{cases} \quad (3)$$

Dados m puntos (x_i, y_i) de una curva, se puede expresar el problema de aproximación de la curva mediante el siguiente sistema de ecuaciones lineales:

$$y_i = \sum_{j=0}^{n-1} c_j N_{j,k}(x_i) \quad (4)$$

donde los coeficiente c_j son desconocidos.

Matricialmente el sistema de ecuaciones se expresa de la siguiente manera:

$$Y = N \times C \quad (5)$$

donde:

$$Y = [y_0 y_2 y_3 \dots y_{m-1}]^T \quad (6)$$

$$N = \begin{pmatrix} N_{0,3}(x_0) & N_{2,3}(x_0) & \dots & N_{n-1,3}(x_0) \\ N_{0,3}(x_1) & N_{2,3}(x_1) & \dots & N_{n-1,3}(x_1) \\ \dots & \dots & \dots & \dots \\ N_{0,3}(x_{m-1}) & N_{2,3}(x_{m-1}) & \dots & N_{n-1,3}(x_{m-1}) \end{pmatrix} \quad (7)$$

$$C = [c_0 c_2 c_3 \dots c_{n-1}]^T \quad (8)$$

El sistema planteado está sobredeterminado, es decir, tiene más ecuaciones que incógnitas. Este sistema sobredeterminado se resuelve a través de la matriz pseudoinversa por izquierda (L) según la siguiente expresión:

$$L = M^{-1} N^T \quad (9)$$

siendo $L_{(n \times m)} \cdot N_{(m \times n)} = I_{(n \times n)}$. Para $m > n$ puede calcularse $L_{(n \times m)}$ construyendo la matriz $M_{(n \times n)}$ como:

$$M = N^T N \quad (10)$$

Si el determinante de (M) $\neq 0$, existe M^{-1} y puede escribirse:

$$M^{-1} N^T N = M^{-1} M \quad (11)$$

La solución del sistema sera:

$$C = LY \quad (12)$$

3. Presentación del Descriptor

El descriptor de este trabajo corresponde a los coeficientes de la aproximación B-spline de la curva que representa la posición relativa de los puntos de mínimo gris dentro del grafema. Los puntos de mínimo gris corresponden al pixel de menor valor de gris sobre la línea perpendicular al esqueleto del grafema. La posición relativa corresponde a la distancia euclídea entre el punto de mínimo gris y el borde del trazo, como se puede observar en la figura 1(a). Considerando las distancias de todos los puntos de mínimo gris se conforma una señal que es característica para cada individuo. A esta señal o curva se la denomina, de aquí en adelante, como señal de posición relativa de los puntos de mínimo gris.

En general las señales de distancia de posición relativa tienen distinta cantidad de puntos dependiendo de la imagen. Lo anterior implica que es necesaria una etapa de normalización de la longitud de la curva, representada en la figura 1(b). Esta normalización se realizó redimensionando la señal a la longitud

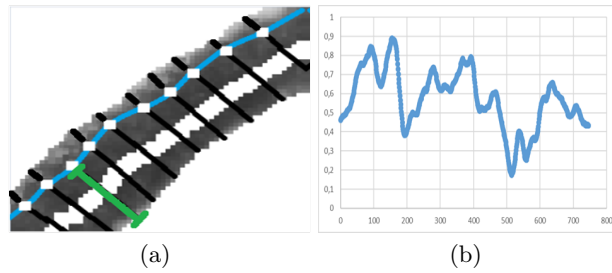


Figura 1. Descriptor: (a) señal de posición relativa de los puntos de mínimo gris (b) Normalización de la señal caracter C

promedio de todas las señales de la base de datos. Luego de la normalización anterior se observa que las señales tienen una longitud demasiado grande para implementar una etapa de clasificación supervisada (sobre 700 puntos aproximadamente). Para reducir la dimensionalidad del descriptor se recurre al cómputo de los coeficientes de la interpolación B-spline de la señal de distancia relativa normalizada.

Como es ampliamente sabido, si se utilizan pocas bases la aproximación de interpolación tiene un error elevado respecto de la señal original. A medida que se van aumentando las bases el error disminuye y la aproximación se va ajustando cada vez más respecto de la original. No obstante lo anterior llega un punto que el error deja de disminuir de manera significativa, reduciéndose el error de manera muy marginal. Para determinar la cantidad adecuada de coeficientes

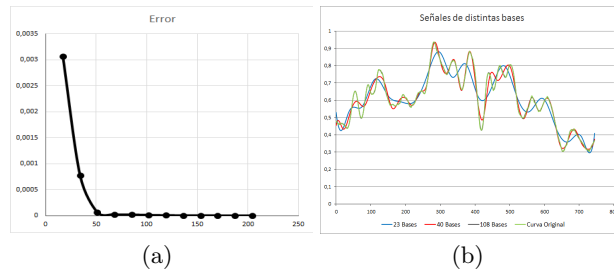


Figura 2. Error: (a) Disminución del Error (b) Señales de distintas bases y curva original

se comienza a aumentar de forma paulatina la cantidad de bases hasta que la disminución del error es despreciable. Para mostrar este fenómeno se presenta la figura 2. Se escogió aleatoriamente una de las señales de distancia relativa de todo el conjunto disponible y con esta señal se realizan las aproximaciones aumentando las bases. En la figura 2(a) se observa que basta con 74 coeficientes para reconstruir nuevamente la señal original con un muy bajo error. En la figura

2(b) se presentan 3 casos de aproximaciones para 23 (azul), 40(rojo) y 108(gris) bases y la curva original(verde). Se observa que la aproximación con la mayor cantidad de bases se superpone casi perfectamente a la curva original.

4. Materiales y Métodos

Para evaluar la capacidad de discriminación de personas del descriptor propuesto se implementó un clasificador multiclase, basado en SVM de salida binaria, entrenado con Validación Cruzada y considerando el problema de desbalanceo de clases usado en [13]. La base de datos de símbolos simples de donde obtener las muestras para la experimentación se detalla a continuación .

4.1. Base de Datos

Un total de 15.000 imágenes de grafemas forman la base de datos construida especialmente. Las imágenes son a color de 24 bits y de mediana resolución, aproximadamente de 800 por 800 píxeles. Se consideraron las 6 grafemas que se muestran en la figura 3, por ser rasgos simples con trazos ascendentes, descendentes y curvos en distintas direcciones, que dan lugar a los gestos gráficos que revelan las características personales de su autor. 50 personas realizaron 50 muestras de cada una de los grafemas escogidos, de acuerdo con los modelos presentados. Las muestras se recogieron bajo condiciones controladas, los cola-

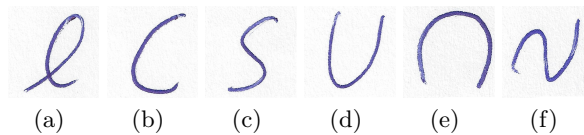


Figura 3. Símbolos de la base de datos.

boradores escribieron sobre una hoja de papel de $75(g/m^2)$, utilizando 5 hojas del mismo tipo como base de apoyo y el instrumento de escritura fue un bolígrafo bic trazo grueso de color azul, cuya bolita está fabricada en tungsteno de 0,7 mm. La captura de las imágenes se realiza mediante un escáner convencional, lo que permite realizar una captura masiva de imágenes utilizando un dispositivo de uso cotidiano.

4.2. Modelo de clasificación

A continuación se presenta en detalle la metodología de entrenamiento y clasificación. El reconocimiento de personas por un carácter simple, planteado en este artículo, es un problema multiclase. Las muestras correspondientes a un

grafema, realizadas por una misma persona, forman una clase. Por consiguiente en la base de datos presentada se definen 50 clases diferentes para un símbolo. Para resolver el problema se construyó un sistema de múltiples clasificadores binarios (SVM). Cada clasificador distingue entre una de las clases y el resto como segunda clase, técnica conocida como uno-contra-todos (one-versus-all) [14]. Para formar un grupo balanceado, se decidió optar por eliminar casos de la clase mayoritaria, consiguiendo una relación de 1 a 5. El conjunto queda de 295 muestra, 50 de la persona elegida como clase positiva y 5 muestras de cada una de las restantes personas, clase negativa. SVM necesita dos etapas: el entrenamiento y la fase de pruebas. El conjunto de entrenamiento se utiliza para ajustar los parámetros del modelo, y el de prueba, para evaluar los parámetros de los mismos. Para garantizar que los resultados sean independientes de la partición entre datos de entrenamiento y prueba se utilizó en el modelo validación cruzada aleatoria con 10 iteraciones, dividiendo las muestras de datos balanceadas en un 70 % para el entrenamiento y 30 % para el test. Los datos en el modelo planteado, no son linealmente separables a través de un hiperplano óptimo en el espacio de entrada, por lo que se construye un sistema con función Kernel de base radial (RBF). Siguiendo el desarrollo de [15], la expresión del hiperplano de separación L_D es la siguiente:

$$L_D = \sum_j \alpha_j - \frac{1}{2} \sum_j \sum_k \alpha_j \alpha_k y_j y_k x'_i x_k \quad (13)$$

el problema de SVM consiste en maximizar el funcional L_D sujeto a las siguientes restricciones:

$$\begin{aligned} 0 &\leq \alpha_j \leq C \\ \sum_j \alpha_j y_j &= 0 \end{aligned} \quad (14)$$

donde α_j son los multiplicadores de Lagrange, y C es un parametro que permite mantener acotados dichos multiplicadores. Los α_j no nulos corresponden a los vectores de soporte y determinan frontera de decisión. En particular, la expresión de la función de Base Radial es la siguiente:

$$K(x, y) = \exp\left(\frac{-\|x-y\|^2}{2\sigma^2}\right) \quad (15)$$

donde σ es el parámetro de la función kernel.

Se realiza el cómputo de los parámetros del SVM con optimización aleatoria. Se itera 100 veces el entrenamiento del clasificador SVM con validación cruzada. Por cada iteración se obtiene la pérdida, error cuadrático medio, de los 10 folds. Se calculan los promedios por iteración y se obtienen los parámetros óptimos del modelo, correspondientes al promedio mínimo de las pérdidas por iteración.

Se entrena el clasificador SVM de los parámetros computados con validación cruzada. Para lo cual se hace una nueva partición y se calcula el promedio de las pérdidas y se elige como modelo el que tiene pérdidas más cercanas al promedio. Se evalúa este modelo SVM. Se calcula la matriz de confusión para el grupo de muestras balanceado considerado. Para obtener mayor independencia de los

datos de entrada se forman 20 grupos de muestras balanceados por persona, que se usan para entrenar y evaluar el modelo, obteniéndose como resultado una matriz de confusión por grupo. Luego se calcula la matriz de confusión promedio para cada persona. Se obtienen los siguientes indicadores a partir de la matriz de confusión:

- $TP = \text{AciertosPositivos}$
- $TN = \text{AciertosNegativos}$
- $FP = \text{FalsosPositivos}$
- $FN = \text{FalsosNegativos}$
- $ACIERTOS = TP + TN$
- Sensibilidad o la verdadera tasa positiva

$$TPR = \frac{TP}{TP+FN}$$
- Especificidad o la verdadera tasa negativa

$$TNR = \frac{TN}{FP+TN}$$
- $RHO = \frac{TP}{TP+FP+FN}$

5. Resultados

En esta sección se presentan resultados numéricos que permiten evaluar y comparar el funcionamiento en la verificación de personas de la señal de distancia relativa y de los coeficientes de la aproximación B-spline.

En la tabla 1 se observan los resultados obtenidos para la letra C de los 50 individuos de la base de datos, tanto para la señal de distancia relativa como para los coeficientes B-spline. Cada fila corresponde al promedio de la evaluación del conjunto de test de 20 grupos balanceados. La última fila corresponde al promedio de las 50 personas. Como se observa de la última fila, cada SVM en promedio reconoce correctamente la identidad del 97% de los individuos para ambos descriptores.

La tabla 2 presenta los resultados para todos los símbolos. Se muestran los valores en promedio obtenido para cada uno de los indicadores considerados a partir de la matriz de confusión. Se observa que el porcentaje de reconocimiento utilizando todos los símbolos es superior a 97%.

Persona	% Aciertos Reducción	% Aciertos Descriptor	Persona	% Aciertos Reducción	% Aciertos Descriptor
1	98	98	26	96	97
2	96	97	27	96	97
3	98	97	28	96	97
4	96	95	29	95	95
5	98	98	30	97	98
6	97	96	31	97	96
7	98	97	32	98	99
8	95	95	33	99	98
9	97	98	34	100	100
10	98	99	35	96	97
11	97	97	36	96	97
12	97	98	37	95	93
13	98	98	38	97	97
14	98	97	39	98	97
15	98	97	40	98	98
16	96	94	41	98	98
17	97	97	42	94	93
18	99	98	43	98	98
19	98	97	44	98	97
20	96	97	45	99	96
21	95	95	46	97	98
22	99	97	47	98	97
23	96	96	48	98	97
24	98	98	49	96	96
25	98	99	50	94	92
			Promedio	97	97

Tabla 1. Símbolo C

Simbolo	TP	TN	FP	FN	ACIERTOS %	TPR	TNR	RHO
C	43	244	1	7	97	85	100	0.83
~	44	244	1	6	97	87	100	0.85
e	42	244	1	8	97	85	100	0.83
∩	42	244	1	8	97	84	100	0.82
s	44	244	1	6	98	87	100	0.86
u	42	244	1	8	97	84	100	0.82

Tabla 2. Promedio de los resultados por símbolo

6. Conclusión

Este artículo ha presentado un nuevo descriptor para identificación de personas usando trazos manuscritos. El descriptor propuesto utiliza los coeficientes de la aproximación B-spline de la curva que representa la posición relativa de los puntos de mínimo gris dentro del grafema. Además de lo anterior, se considera una imagen de un trazo manuscrito simple, en lugar de un trazo complejo como la firma de un individuo. Para evaluar el funcionamiento del descriptor se ha realizado un estudio con un clasificador multiclase, basado en SVM de

salida binaria, entrenado con Validación Cruzada y considerando el problema de desbalanceo de clases. Se obtuvo que, en promedio, el descriptor reconoce correctamente la identidad del 97% de los individuos.

Referencias

1. Schomaker, L.: Advances in writer identification and verification. In: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). Volume 2., IEEE (2007) 1268–1273
2. Arabadjis, D., Giannopoulos, F., Papaodysseus, C., Zannos, S., Rousopoulos, P., Panagopoulos, M., Blackwell, C.: New mathematical and algorithmic schemes for pattern classification with application to the identification of writers of important ancient documents. *Pattern Recognition* **46** (2013) 2278–2296
3. Hafemann, L.G., Sabourin, R., Oliveira, L.S.: Offline handwritten signature verification-literature review. arXiv preprint arXiv:1507.07909 (2015)
4. Smekal, Z., Mekyska, J., Rektorova, I., Faundez-Zanuy, M.: Analysis of neurological disorders based on digital processing of speech and handwritten text. In: *Signals, Circuits and Systems (ISSCS), 2013 International Symposium on*, IEEE (2013) 1–6
5. Horster, P.: *Communications and Multimedia Security II*. Springer (2016)
6. Vielhauer, C.: *Biometric user authentication for IT security: from fundamentals to handwriting*. Volume 18. Springer Science & Business Media (2005)
7. Impedovo, D., Pirlo, G., Russo, M.: Recent advances in offline signature identification. In: *2014 14th International Conference on Frontiers in Handwriting Recognition*. (2014)
8. Halder, C., Obaidullah, S.M., Roy, K.: Offline writer identification and verification—a state-of-the-art. In: *Information Systems Design and Intelligent Applications*. Springer (2016) 153–163
9. Hannad, Y., Siddiqi, I., El Kettani, M.E.Y.: Writer identification using texture descriptors of handwritten fragments. *Expert Systems with Applications* **47** (2016) 14–22
10. Halder, C., Obaidullah, S.M., Paul, J., Roy, K.: Writer verification on bangla handwritten characters. In: *Advanced Computing and Systems for Security*. Springer (2016) 53–68
11. Aubin, V., Doorn, J., Kaplan, G.: Nuevos descriptores para la Identificación de Personas basados en la Simetría del Trazo. In: XVIII Congreso Argentino de Ciencias de la Computación. (2013) Available in <http://sedici.unlp.edu.ar/handle/10915/31655>.
12. Farin, G.: *Curves and surfaces for computer-aided geometric design: a practical guide*. Elsevier (2014)
13. Aubin, V., Mora, M., Santos, M.: A new descriptor for people recognition by handwritten strokes analysis. *International Conference on Pattern Recognition Systems ICPRS-16* (2016) 14–20 doi:10.1049/ic.2016.0038.
14. Gomez-Moreno, H., Gil-Jimenez, P., Lafuente-Arroyo, S., Vicen-Bueno, R., Sanchez-Montero, R.: Color images segmentation using the support vector machines. *Recent Advances in Intelligent Systems and Signal Processing* (2003) 151–155
15. Matlab: *Statistics and Machine Learning Toolbox*. The MathWorks, Inc. (2015) Available in <http://www.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html>.

Experiences accelerating features selection in Viola-Jones algorithm

Germán Lescano^{1,2}, Pablo Santana-Mansilla^{1,2}, Rosanna Costaguta¹

¹Instituto de Investigación en Informática y Sistemas de Información (IISI)
Facultad de Ciencias Exactas y Tecnologías (FCEyT)
Universidad Nacional de Santiago del Estero (UNSE)

²Consejo Nacional de Investigaciones Científicas y Tecnológicas (CONICET)
{gelescano,psantana,rosanna}@unse.edu.ar

Abstract. Faces and facial expressions recognition is an interesting topic for researchers in machine vision. Viola-Jones algorithm is the most spread algorithm for this task. Building a classification model for face recognition can take many years if the implementation of its training phase is not appropriately optimized. In this study, several settings for implementing the training phase are analyzed. The aim was to share our experiences when we try to accelerate the training phase using one computer with a graphical processing unit (GPU). For each setting, the execution times were analyzed and compared with previous studies. Although we don't contribute to break new ground in topic or methodology, we decide to share our experience in order to show an antecedent working with a cheap GPU with the aim that this can be useful to another for to make comparisons.

Keywords: Adaboost, Viola-Jones Algorithm, feature selection, CUDA

1 Introduction

Face and facial expressions recognition is an interesting topic for researchers in machine vision [9]. An important stage in a face recognition algorithm is the building of a classification model that can discriminate faces. Building a classification model require a training phase during which a sample of images is analyzed with the aim of extracting those features that best describe a face.

Viola and Jones [11] proposed an algorithm that can detect faces in real time. This algorithm can be implemented on a wide range of small low power devices, including hand-helds devices and embedded processors. However, a drawback of this algorithm is that the training phase is extremely time-consuming.

In this work, we propose and analyze five settings to implement the training phase of Viola-Jones algorithm. Each setting tries to reduce the execution times when working on a single computer. Two settings involve the use of sequential computing and the other three involve the use of parallel computing, specifically CUDA architecture. CUDA is a parallel computing platform and programming model invented by NVIDIA [12]. It enables dramatic increases in computing performance by harnessing

the power of the graphics-processing unit (GPU). In order to reduce execution times, we had focus on feature selection because this process has a notable impact on training times.

This paper is organized as follows. In section 2, the purpose, the utility and the training phase of Viola-Jones algorithm are described. In section 3, five settings of development for training face are described. In section 4, our experimental results are showed. In section 5, we compare our proposal respect other alternatives we found in the literature. In section 6, conclusions and the future research directions are presented.

2 Revision of the training phase in Viola-Jones Algorithm.

The Viola-Jones algorithm describes a framework for object detection. It is widely used in a variety of software and hardware applications that incorporate elements of computer vision, like the face detection module in video conferencing, human-computer interaction, and digital photo cameras [6].

Viola and Jones [11] propose a variant of Adaboost algorithm [2] for the training phase that is related with the selection of a small number of features that best describe a face. These features are known as weak classification functions and are combined to build a stronger classifier.

The features used by Viola and Jones are reminiscent of Haar basis functions [8]. Figure 1 shows five simple features usually employed. These features are defined by two, three or four rectangles. To compute a feature, the sum of pixels within the white rectangles should be subtracted from the sum of pixels within the black rectangles.

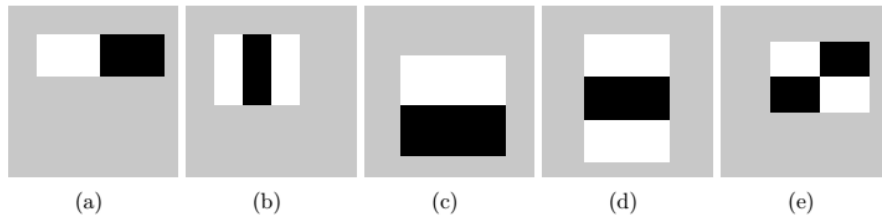


Fig. 1. Haar-Like patterns frequently employed [11].

Supposing f is a feature, θ is a threshold, $p \in \{-1,1\}$ is the polarity that indicates the direction of the inequality and x is a sub-window of an image, a weak classifier to detect a face in x can be defined by Equation 2.

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{si } pf(x) < p\theta \\ 0 & \text{en otro caso} \end{cases} \quad (2)$$

Each iteration of the boosting algorithm is designed to select the single rectangle feature which best separate the positive (face images) and negative examples (not face images). For each feature, the weak classifier determines the optimal threshold of

classification function, such that the minimum number of examples is misclassified. Figure 2 shows the boosting algorithm proposed by Viola and Jones [11].

Boosting Algorithm

- Given example images $(x_1, y_1), \dots, (x_n, y_n)$ where $y_i=0, 1$ for negative and positive examples respectively..
- Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i=0,1$ respectively. Where m is the number of positives examples and l the number of negatives examples.
- For $t = 1, \dots, T$ (T weak classifiers)

1. Normalize the weights, $w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$
2. Select the best weak classifier with respect to the weighted error:

$$\varepsilon_t = \min_{f,p,\theta} \sum_i w_i |h(x_i, f, p, \theta) - y_i|$$

Define $h_i(x)=h(x, f, p, \theta_t)$ where f, p, θ_t are the minimizers of ε_t .
3. Update the weights: $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$
 Where $e_i=0$ if example x_i is classified correctly, $e_i = 1$ otherwise, and

$$\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$$
 - The final strong classifier is:

$$C(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{otherwise} \end{cases}$$

Where $\alpha_t = \log \frac{1}{\beta_t}$

Fig. 2. Boosting Algorithm [11]

Viola-Jones algorithm to select weak classifier is depicted in Figure 3. This pseudo-code is an interpretation proposed by Morelli and Padovani [5] and its operation can be summarized as follow. For each feature, the images into the samples are sorted by feature value in ascendant way. The Adaboost optimal threshold for that feature can then be computed in a single pass over this sorted list. On each iteration over the sorted list, four sums are evaluated for each element: the total sum of positive examples weights T^+ , the total sum of negative example weights T^- , the sum of positive weights below the current example S^+ and the sum of negative weights below the current example S^- . Also for each feature, an error is computed using the Equation 3. This value represents the error that we would produce if the element were considered the threshold for the feature. Once all errors have been computed, the lowest one is selected.

$$\varepsilon = \min(S^+ + (T^- - S^-), S^- + (T^+ - S^+)) \quad (3)$$

```

selectedFeature =  $\emptyset$ 
selectedFeatureError =  $\infty$ 
selectedThreshold = 0
 $T^+ \leftarrow$  total sum of positive examples weights (1)
 $T^- \leftarrow$  total sum of negative examples weights (2)
for all  $f \in$  features do
   $X^{(f)} \leftarrow$  training images sorted by  $f$  value (3)
   $e_f = \infty$ 
   $\theta_f = 0$ 
  for  $i = 1$  to  $N$ 
     $S_i^+ \leftarrow$  the sum of positive weights below the current example
     $x_i^f$  (4)
     $S_i^- \leftarrow$  the sum of negative weights below the current example
     $x_i^f$  (5)
     $e_i = \min(S_i^+ - (T^- + S_i^-), S_i^- - (T^+ - S_i^+))$  (6)
    if  $(e_i - e_f)$  then
       $e_f = e_i$ 
       $\theta_f = f(x_i^f)$ 
    end if
  end for
if selectedFeatureError >  $e_f$  then
  selectedFeature =  $f$ 
  selectedFeatureError =  $e_f$ 
  selectedThreshold =  $\theta_f$ 
end if
end for

```

Fig. 3. Algorithm for weak classifiers selection [5].

3 Experimental Settings for Implementing the Training Phase

Five experimental settings were proposed to implement the training phase of Viola-Jones algorithm. The C language was chosen to code each experimental setting because it enables us to make decision of programming at low-level and it is compatible with CUDA.

3.1 First Experimental Setting

The implementation that was done with this experimental setting was sequential. We allocated in memory a matrix that stores the identifier of each training file, the category of the image (face or not face), its weight and the category (face or not face) assigned by the classifier algorithm. In addition, we use as many files as training images in order to save the features of the images. Each file has 162336 lines (this number correspond to the total quantity of features that can be generated for an image of 24 x 24 pixels) and each line registers the details of one feature.

In Figure 3, the instructions that can have in memory the data necessary for computing them were tagged with (1), (2), (4), (5) and (6). Meanwhile, the instruction tagged with (3) has a lot of access to hard disk because it needs data saved in files. In instruction (3) the sorting task is made through the Bubble sort method, so we have a loop that iterates over 162336 features. In each iteration, the values of a feature are extracted from each training image file. These values are employed to re-sort the training images. The access to details of a feature in the file is sequential.

3.2 Second Experimental Setting

This experimental setting is similar to the first one in the sense that it operates with data saved on files. The difference is that with this second setting we create one file to save the values of all features on a unique place. Each line of this file contains all values of a feature extracted from each training image. This change reduces in a 99,99% the amount of access to files, from 2.435.040.000 (162336 features x 15000 sample images) to 162336, during the sorting task. In this second setting, we changed the Bubble method for the Quick Sort method because the last one generally has best performance during sorting tasks. The way of accessing to the feature values was also changed. Sequential access was changed for random access through the *fseek* function.

3.3 Third Experimental Setting

This experimental setting continues with the use of a unique file in order to store for each feature the values that are observed in each one of the sample files. However, we employ the Thrust library. Thrust is a parallel algorithms library that enhances programmer productivity while enabling performance portability between GPUs and multicore CPUs [13]. This library was used for sorting, operation tagged with (5) in Figure 3, and in (4) and (6) operations of this figure for accomplish them through a reduction operation.

3.4 Fourth Experimental Setting

In this experimentation, we changed the way of applying the sum operations involved in the process of weak classifiers selection. The operations to sum the weight of positive samples (4) and negative samples (5) of Figure 3 were implemented in the

same procedure so as to take advantage of the processing cycle, and the procedure was parallelized based on the algorithm proposed by Harris [3]. Small changes were applied to the algorithm in order to make two sum operations at the same time.

This experimental setting does not sort the values of the samples with respect to a particular feature. For each value in the samples with respect to a particular feature, the calculation of $\varepsilon = \min(S^+ + (T^- - S^-), S^- + (T^+ - S^+))$ was made in fully parallel way. This way 15000 processing threads were launched, one for each sample.

3.5 Fifth Experimental Setting

On previous experimentations, the data matrix was stored in a disk file, but on this experimental setting, all data are stored in the memory of the computer. To be sure data are not paged for the operative system we use `cudaHostAlloc` instruction that allocates a buffer of page-locked host memory [7].

The concept of stream it is also used. A CUDA stream represents a queue of GPU operations that are executed in a specific order [7]. On each execution, a stream is responsible for processing a row of the matrix, asynchronously loading the data into memory. Two kernels are launched at the same time when data are loaded, one kernel sorts the files of the samples considering the value of the analyzed feature. Since the order of sample files can vary according to the feature and that kernels execution is asynchronous, it is necessary to use other matrix to keep a reference to the order with regard to the feature that the stream analyzes. In this second matrix, each row represents a feature and it stores indexes of the files corresponding to the samples. This matrix is also stored in pinned memory and it is copied in asynchronous way into the streams.

Once the sorting kernel is launched, the following step consists on launching a kernel for performing operations (4), (5) and (6) of Figure 3 for each value of sample files for a particular feature. The results of these calculations are copied asynchronously to an array that stores, for each feature, the mistake that would commit if the value in the sample for this feature was chosen as threshold value. This array is stored in pinned memory.

At the end of the execution of all the streams, the `min_element` function of Thrust library is used to find the smallest mistake in the above mentioned array. This final step allow one to get the selected feature, the threshold value and its corresponding mistake.

4 Results

Table 1 shows the results of executing the experimental settings designed in a computer with the following hardware configuration: i7 processor of 3.4 Ghz and GPU GeForce GT 730 with 2 GB of global memory. The numbers into the first column of Table 1 are operations tagged in Figure 3.

In the first setting, the amount of access has a notable influence in execution times because if we have 162336 features and 15000 training image, then we will require

2435040000 accesses to files. In the second setting, the amount of accesses to files reduces in 99.99% and this enables the reduction of the execution times. However, in the third setting, the execution times go to the bad. In the third setting, the times for making the sorting task are reduced but we lose performance when computing the error for obtaining the threshold for a feature. In the fourth setting, the execution times are improved again and they are better than second setting. This was achieved because in the fourth setting we did not make the sorting task and the execution times for calculating the threshold was reduced. Overall, we got the best performance in the fifth setting. If we compare fifth setting with setting number 4, the fifth setting represents an increasing of about 11x in speed. It seems reasonable to assume that this improvement in performance was obtained thanks to the use of streams and pinned memory.

Table 1. Comparison of time needed to execute the different operations of the method for selecting the weak classifier

Operations	Time (milliseconds)				
	Exp. Setting 1	Exp. Setting 2	Exp. Setting 3	Exp. Setting 4	Exp. Setting 5
(3)	938993.904	3.437	0.540	--	44.381 (8192 streams)
(4), (5) y (6)	0.046	0.046	0.601	0.019	
Approximate time to evaluate a feature	939682.127	696.716	9090.446	519.515	
Approximate time to find a weak classifier	4.84 years	31.42 hs	17.08 days	23.43 hs	2 hs

5 Discussion

A bibliographical exploration regarding to time reduction when building images classifiers with boosting algorithm allowed us to identify research works such as Huang and Shi [4], Abualkibash et al. [1] and Tsai et al. [10]. Huang and Shi [4] report the results that are showed in Table 2 when working with 65230 features and 18676 samples. In their experiments Huang and Shi used computers with a 1.8 Ghz processor.

Abualkibash et al. [1] describe the same experiment that was made by Huang and Shi [4]. Abualkibash et al. [1] utilized computers equipped with quad-core processors but they did not give details about processing capacity of CPUs. Table 3 shows the results reached by Abualkibash et al. [1].

Tsai et al. [10] experimented with a GPU Nvidia Tesla K20c. This GPU, according to technical documentation, has 2496 cores, 706 Mhz of memory frequency, and 5 GB of global memory. Tsai et al. [10] made three experiments: the first one with 1119 features and 19575 samples; the second experiment with 6090 features and 19161

features; and the last experiment with 10640 features and 19140 samples. The results of these three experiments can be seen in Table 4.

Table 2. Comparison between execution time of alternative 5 and execution time during experiments conducted by Huang and Shi [4]

Time to select a feature	
(Huang et al. 2010)	Alternative 5 implemented in Geforce GT 730 of 2GB memory
3.31 minutes (when working with 2 computers)	38.52 minutes (when working with 8192 streams)
1.97 minutes (when working with 4 computers)	

Table 3. Comparison between execution time of alternative 5 and execution time during experiments conducted by Abualkibash et al. [1]

Time to select a feature	
(Abualkibash et al. 2013)	Alternative 5 implemented in Geforce GT 730 of 2GB memory
24.6 seconds (when working with 6 computers)	38.52 min (when working with 8192 streams)
6.4 seconds (when working with 21 computers)	
5.2 seconds (when working with 26 computers)	
4.8 seconds (when working with 31 computers)	

Table 4. Comparison between execution time of alternative 5 and execution time during experiments conducted by Tsai et al. [10]

Experiments	Time to select a feature	
	Tsai et al. [10]	Alternative 5 implemented in Geforce GT 730 of 2GB memory
1119 features 19575 samples	0.788 seg	1.413 min with 1119 streams
6090 features 19161 samples	1.157 seg	7.313 min 4096 streams
10640 features 19140 samples	1.217 seg	6.2497 min Con 8192 streams

Considering results that are showed in Tables 2, 3 and 4 one could conclude that the fifth experimental setting proposed in this paper is not a good option to reduce the time of classifiers building. Nevertheless, one should consider that the fifth alternative

of solution was tested in a computer with a GPU of limited processing capacity. For this reason, it would be interesting to run the fifth solution in a computer with better hardware capacities and then compare it with the results achieved by Huang and Shi [4], Abualkibash et al. [1] and Tsai et al. [10]. For example, a GPU like the one used by Tsai et al. [10] not only has more computing capacity but also it is possible to execute more streams.

Considering it was wanted to execute the fifth experimental setting in a computer with more computing capacity and that the laboratory at the university had not have more powerful computers than the described in point 4 (i7 processor of 3.4 Ghz and GPU Geforce GT 730), it was decided to hired the computer instance g2.2xlarge from amazon.com. The instance g2.2xlarge has: 8 virtual CPU Intel Xeon E5-2670; 15 GB of RAM; and a GPU Nvidia Grid K520 with 4 GB of RAM, 797 Mhz of frequency and 1536 cores. Table 5 shows a comparison of time needed for executing the experiments of Table 4 (the same amount of features and sample files) in the hired instance and the GPU GeForce GT 730. It was expected that execution in the GPU available in g2.2xlarge instance would be faster than the execution in GeForce GT 730. Nonetheless, the results do not confirm the previous assumption. A reason might be the virtualization effect although this needs further research.

Table 5. Comparison of execution time when implementing solution 5 in GeForce GT 730 and in instance g2.2xlarge.

Experiments	Time to select a feature (Alternative 5 implemented in GeForce Gt 730)	Time to select a feature (Alternative 5 implemented in GPU of g2.2xlarge)
1119 features 19575 samples	1.413 min	4.1789 min
6090 features 19161 samples	7.313 min	22.039 min
10640 features 19140 samples	6.2497 min	38.472 min

6 Conclusions

This work analyzed the implementation of a training process for generating a classifier with the capacity of face recognition. The aim was to reduce the time needed to train the classifier using a single computer. The focus was the process for selection of weak classifiers because this stage is the most invoked during classifiers building and it is the most demanding in terms of execution time. With the several alternatives implemented, sequential and parallel through CUDA architecture, it was possible to achieve substantial improvements.

The use of GPUs for developing and accelerating applications is a feasible alternative because any programmer can acquire one and there are cheap GPU. In addition, there is an effort made by manufacturer of GPU for development software that helps

programmers to use them, for example NVIDIA with her CUDA platform. With the use a GPU, applications can compute many data without wasting too much time.

Further research may involve the measurement of the performance of fifth experimental setting in a computer with bigger computing capacity in its GPU and exploring a multi-GPU approach. Collaterally, we believe that further research is needed for evaluate if virtualization affect in the performance GPU.

7 References

1. Abualkibash, M.; ElSayed, A.; Mahmood, A.: Highly scalable, parallel and distributed Adaboost algorithm using light weight threads and web services on a network of multi-core machines. *International Journal of Distributed and Parallel Systems (IJDPS)*, 4(3): 29-40, May 2013.
2. Freund, Y.; Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1): 23-37. August 1997.
3. Harris, M.: Optimizing parallel reduction in CUDA. Reporte técnico. 2007. Disponible en: http://docs.nvidia.com/cuda/samples/6_Advanced/reduction/doc/reduction.pdf
4. Huang, Z.; Shi, X.: A distributed parallel AdaBoost algorithm for face detection. 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS). Vol 1: 147-150, Oct 2010.
5. Morelli A., Padovani S.: Detección y Reconocimiento de Cara. Tesis de Licenciatura en Ciencias de la Computación. Universidad de Buenos Aires. 2011.
6. Obukhov, A.: Haar Classifiers for Object Detection with CUDA. In: Wen-Mei W. Hwu (Ed.), *GPU Computing Gems*. 517-544. Burlington, MA 01803, USA, 2011.
7. Sanders, J.; Kandrot, E.: CUDA C on multiple GPUs. In: *CUDA by Example. An Introduction to General-Purpose GPU Programming*. 213-236. Boston, MA 02116, USA, 2011.
8. Papageorgiou, C.; Oren, M.; Poggio, T.: A general framework for object detection. *International Conference on Computer Vision*. 555-562, 04 January - 07 January, 1998.
9. Taheri, S.; Patel, V.; Chellappa, R.: Component-Based Recognition of Faces and Facial Expressions. *IEEE Transactions on Affective Computing*, 4(4): pp. 360-371, October-December 2013.
10. Tsai, P.; Hsu, Y.; Chiu, C; Chu, T.: Accelerating AdaBoost algorithm using GPU for multi-object recognition. 2015 IEEE International Symposium on Circuits and Systems (ISCAS), 738-741, May 2015.
11. Viola P., Jones M.: Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2): 137–154, May 2004.
12. NVidia CUDA technology, http://www.nvidia.com/object/cuda_home_new.html
13. Thrust, <http://thrust.github.io/>

Non-Verbal Communication for a Virtual Reality Interface

N. Jofré, G. Rodriguez, Y. Alvarado, J. Fernández, and R. Guerrero

Laboratorio de Computación Gráfica (LCG) - Universidad Nacional de San Luis,
Ejército de los Andes 950
Tel: 02664 420823, San Luis, Argentina
{npasinetti, gbrodriguez, ymalvarado, jmfer, rag}@unsl.edu.ar

Abstract. The steady growth of technology has allowed to extend all forms of human-computer communication. Since the emergence of more sophisticated interaction devices, Human Computer Interaction (HCI) science has added the issue of Non-Verbal Communication (NVC).

Nowadays, there are a lot of applications such as interactive entertainments and virtual reality requiring more natural and intuitive interfaces. Human gestures constitute a great space of actions expressed by the body, face, and/or hands.

Hand Gesture is frequently used in people's daily life, thus it is an alternative form to communicate with computers in an easy way.

This paper introduces a real-time hand gesture recognition and tracking system to identify different and dynamic hand postures. In order to improve the user experience, a set of different system functions into a virtual world had been implemented so interaction can be performed by the user through a data glove device.

Keywords: Non-Verbal Communication, Human Computer Interaction, Virtual Reality (VR), Computer Graphics.

1 Introduction

Some time ago scientists predicted the impending transition in the ways humans would experience the communication of information. Virtual Reality (VR) has allowed the creation of interaction's environments that facilitate new contexts of exchange and communication of information. Currently, VR applications consist in user immersion into a computer-generated environment, resulting in a natural idea to improve the impression of living in a simulated reality [1][2].

It is well known that the use of VR techniques does not alter the physical body, however affects perceptions and interactions of the human being operating in a virtual environment in real time using their senses and natural abilities. At the beginning, VR used the classical interaction devices such as joystick, keyboard, mouse, among others; in the same way they were used in video games. Nowadays, technological advances have resulted in the need for evolution of existing devices and consequently, the emergence of new simple and intuitive

devices to improve user interaction with VR environments [3]. In this context, *Natural User Interfaces* (NUI) appears as an attractive solution to the need raised.

These interfaces are a new paradigm in HCI for a system or application, where the main goal is making use of a natural communication to humans. In fact, capturing information must be in real time achieving a direct interaction without using a peripheral as an intermediary for the entry of information [4]. In consequence, this paradigm exploits the skills that most users have already acquired through living an experience in an everyday world while interacting with other humans [5]. This interaction is what the world knows as *Non-Verbal Communication* (NVC).

While Verbal Communication (information exchange using words) is important, humans had relied on NVC for thousands of years before they developed the capability to communicate with words. The NVC takes many forms or modalities like *facial expression*; *gaze* (eye movements and pupil dilation); *gestures*, and other *bodily movements*; *body posture*; *bodily contact*; *spatial behaviour gesture*; *clothing* and *appearance*; *non-verbal vocalizations*, and *smell* [6].

The main NVC related research areas are *Proxemics*, *Kinesics* and *Paralinguistics*. *Proxemics* analyses the chosen body distance and angle during interaction. *Kinesics* is the study of what is called body language, all body movements except physical contact, which includes gestures – movements of the limbs, postural shifts and movements of some parts of the body like hands, head or trunk. Finally, *Paralinguistics* comprises all non linguistic characteristics related to speech like the selected language or the tone of voice or voice inflexions, among others [6][7].

At the moment, VR implements the NVC by using virtual characters so that the user can interact with environment. Considering that a developed VR system must collect gestural, positional, sound and biometric user information; the use of more sophisticated devices must be done, achieving a NUI system. Thus, the virtual participants respond to the sensory data as if it were real, where response is at every level from physiological through to cognitive one [8][9].

In this context, this work presents a VR system based on NUI principles. The system allows the simulation of gestures and animations of a human hand based on the recognition of their movements, using a data glove as an input device. A recogniser and tracker system methods enables to provide a natural interaction for the user.

2 Concepts on the Generic Non-Verbal Communication Framework

NVC is the main channel through which is experienced inner life of others. Is a significant and ubiquitous aspect of life, so it is not surprising to observe that computing technology expects to integrate seamlessly the daily life and naturally identify the automatic understanding and synthesis of NVC, i.e, as adept computers to the natural modes of human operation and communication [10].

Thus, to get an interface between computers and some of the most important aspects of human psychology such as emotions and social attitudes.

In particular, the gesture interaction is one of the research topics that belong to the scientific field of HCI and have been examined thoroughly within the past years. Their objective is to recognise meaningful expressions of humans with any part of their body (hands, arms, head, upper body). The spectrum of applications have given significant results in many areas such as artificial intelligence, robotics, medicine, sign language, video games, performing arts, among others [11][12].

2.1 NUI based on Virtual Reality

All interactive system should provide the user with an interface through which it can move and interact in the virtual environment. The interface's design should include features like functionality, ease of use, intelligence and adaptation; so it is reasonable to think that the same user can personify a character in the environment together with other objects or virtual characters. In order to achieve the above, it is necessary to understand before some basic VR environments personification's concepts like as *user perspective*, *modelling*, *rendering* and *physics*.

First Person (FP) view, the most common *user perspective*, allows the user to perceive the environment through the eyes of the character, observing his around up-close, giving a clear view of the scenario in front of him. This FP feature establishes a "user-character" relationship to provide the most immersing feel and improving learning abilities stimulating his visual and auditory capabilities [13].

The VR applications development process require to support scientific *modelling* and *rendering* processes. Both processes are designed to ensure that the final product is appealing and, as much as possible, is real to the user's eyes, allowing the developer focusing on the target instead of centering his attention on environment aspects.

Considering the *physics* aspect, objects in any VR application should behave as in reality including their physical characteristics, so it is necessary to simulate aspects like gravity and collision avoiding crossing between solid objects, then achieving a behavior like the impact of a ball on the ground and soft objects deformations, among others.

All of the above concepts should not affect the performance, in particular *real-time*, since the user acts according to what it is displayed while interacting with the application. Taking into consideration the performed user action, the graphical application refreshes the display, returning a feedback to the user.

Formally, experts in NUI approve that an interface is natural if it obeys the following considerations [14]:

- Create an experience that gives the feeling of being an extension of the body.
- Create an experience that is natural for both expert users and new users.
- Create an experience that is authentic to the environment.

- Create a user interface that takes into account the context, been aware of right metaphors, visual cues, feedback and input and output methods.

Most of current computer games and VR applications usually employ a “person” within the scenario who unfolds through commands specified by the user through a device. This “person” is called *Avatar* and consists of a character whose behaviour is explicitly controlled by an external animator using control commands.

Considering VR applications, NUI concepts and finally Gesture Interaction, the interface should provide to the user with certain facilities for a more bearable interaction, i.e., using the gestural medium to develop within the scenario. In the interaction through an avatar, each gesture is a control command associated to a process trying to match the incoming data from data glove with the modelled avatar [15][16]. In general, gestures can be used for both as symbolic language and as multi-dimensional control tools. Within the standard primitives for the virtual behavior, the following multi-dimensional control tools were defined by [17][18]:

- **Navigation**, the displacement of the user in the virtual space and the “cognitive map” he/she builds of it.
- **Selection**, the action of user pointing to an object.
- **Manipulation**, the user modification of an object’s state; and
- **System control**, the dialogue between the user and the application.

Added to the defined Navigation and System control tools a *gestural protocol* is needed. Theoretically, a protocol is defined as a set of rules set out in the communication process between two systems. In an HCI protocol, the user and computer are the two involved systems where communication is carried out by means of human gestures.

There exist a diversity of methods, techniques, interfaces and applications that are used for several purposes. In every instance different equipment is used, requiring specific hardware and software to get and process the needed data based on the strategy of the researcher and the needs of each project.

Many existing works use specific devices for gesture recognition and hand’s movements tracking, nevertheless one of the most interesting ways to get what was mentioned is by using data gloves [19].

3 Gestural Interface System Features

Gesture recognition and hand movements tracking technology consists of devices capable of tracking the human body movements and communicating the data to computer [20]. Its potential is to change the way users interact with computers by eliminating input devices such as joysticks, mice and keyboards and allowing the unencumbered body to give signals to the computer through gestures (such as finger pointing) or movements.

At the beginning, many works of the mentioned issues used a regular camera as a device for its developments, based on image processing technique [20]. However, nowadays, there exist special equipment that the user can attach to his body.

This study focuses on the gestural interface of a system where the interaction device is a data glove consisting of an integrated sensor textile glove. This device has been chosen under the need for natural interaction by the user using the hands into a virtual world [21][19]. The system's interface allows a real-time interaction where the user communicates through an *avatar*.

The following explains the basic features offered by the system to give more sophistication and naturalness to the interface. It has been developed according to the mentioned multi-dimensional control tools (Section 2.1), excepting the selection tool.

- **Navigation.** Most of the known and implemented navigation options used by game and VR applications developers are: “walk”, “stop”, “run”, “jump”, change different views of the scenario, among others. Particularly, this system has implemented the “walk”, “stop” and change of view of the scenario (See Fig. 1 a)). All of them are associated to a set of gestures (protocol) that performs each action. As an extra functionality of the system, it is possible to remotely control an air-plane toy included in the scenario (See Fig. 1 b)). As it is possible to do two different types of navigation (avatar navigation and air-plane navigation), it is necessary to perform a gestural protocol to swap between them.
- **Manipulation.** While a user interacts naturally on scenario, has also the need to perform handling actions in it. Basically, these actions are to be able to touch the virtual elements and grab them. Some involved factors for implementing this functionality are: physics treatment, proximity to objects (to grab) and avatar hand position. (See Fig. 1 c))
- **System Control.** This control tool relates to the user-application dialogue (Section 2.1). In this approach user-application dialogue implies to control the system by using simple and natural gestures. This feature is achieved through navigation and manipulation tools.

As a consequence of the personification of an avatar by the user it would be nice to be able to visualise user hands and avatar hands as synchronised as possible. Thus, this system implements the next feature:

Real-time Imitation. This involves that avatar's hands can mimics user's hands through a refined model management, physics treatment and rendering tuning (frame rates) (See Fig. 2). Realistic visualisation of the replicated gestures needs the model representation of hands and mapping of human hands' motion into the model which is not straight-forward due to different sizes and joint structures of bones for the human hands. It is necessary to consider that some movements are not feasible by the user then the avatar should not be able to do it.

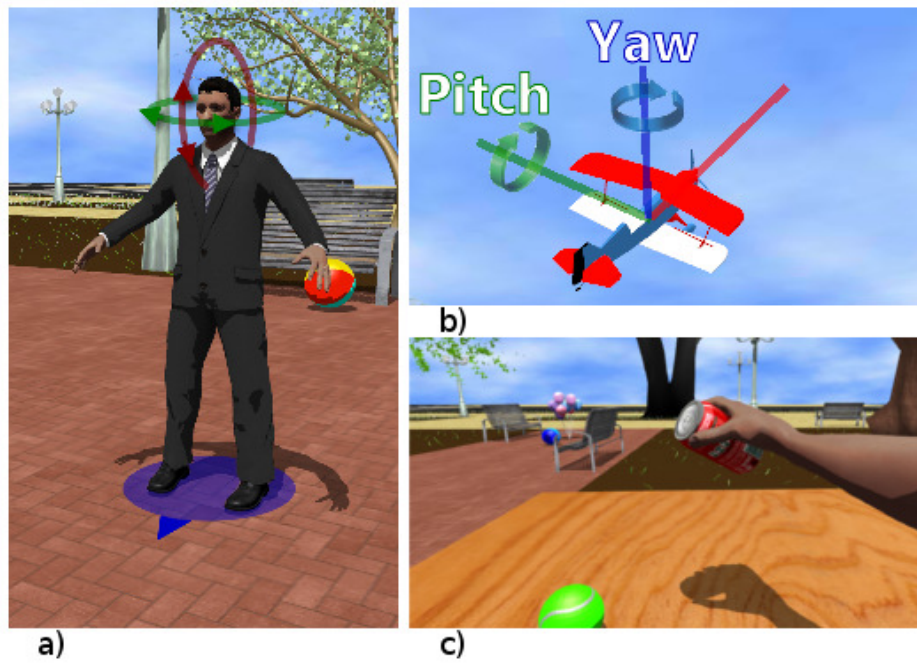


Fig. 1. System's Features: a) Change of View Avatar, b) Airplane Navigation, c) Grasping in a work's table.



Fig. 2. Real time Imitation Feature.

4 Issues Concerning the Interface

The developed system is a system that operates in real time. Since the interaction should appear as natural as possible, it is necessary to consider certain particular features of the operating modules, which will make the proper and efficient operation of the system.

The used input device was an integrated sensor textile glove *DG5 Vhand Data Glove 2.0* distributed by *DGTech Engineering Solutions* [22].

4.1 Data Acquisition (Sampling)

The data glove is a sensor device with five embedded bend sensors which enables to accurately measure the finger movements, while embedded 3 axes accelerometer allows to sense both the hand movements and the hand orientation (roll and pitch). Direct acquisition is performed by the different sensors to capture performer actions. On figure 3, the three former columns show the captured values for the fingers sensors.

4.2 Data Processing (Mapping)

The mapping procedure consists in a function going from bending, orientation and acceleration domains to values within the avatar model domain. Acquired data are matched to model data getting a transition from physical detected movement (real domain) to model avatar image movement (virtual domain). On figure 3, the last column shows the resulting values from the mapped captured values on third column (input data).

Data Glove Fingers	Domain	Input Data	Hand mesh (output)		
			Proximal phalange	Middle phalange	Distal phalange
Thumb	[0.0 – 100.0]	17.87	(0.99, -0.04, 0, 0.08)	(0.99, 0, 0, 0.08)	(0.99, 0, 0, 0.08)
Index	[0.0 – 100.0]	24.90	(0.99, -0.06, -0.06, -0.06)	(0.99, -0.06, -0.06, -0.06)	(0.99, -0.06, -0.06, -0.06)
Middle	[0.0 – 100.0]	53.01	(0.95, 0.21, 0.21, 0.30)	(0.95, 0.21, 0.21, 0.30)	(0.95, 0.21, 0.21, 0.30)
Ring	[0.0 – 100.0]	43.75	(0.97, 0.10, 0.10, 0.20)	(0.97, 0.10, 0.10, 0.20)	(0.97, 0.10, 0.10, 0.20)
Pinky	[0.0 – 100.0]	33.59	(0.99, 0.04, 0.04, 0.09)	(0.95, 0.21, 0.21, 0.30)	(0.99, 0.04, 0.04, 0.09)

Fig. 3. An instance of Input Data and Deviation for Avatar Fingers Real-Time Imitation.

4.3 Gesture Recognition

Each gesture is determined by tracking mapped data. Gesture recognition takes place by matching the tracked data with some gestural values set. Valid gesture values are previously stored on a data base.

4.4 Gestural Protocol Detection.

The real time interface is responsible for controlling whether the user has executed any of the developed protocols. Each gestural protocol provided by the system arrange a set of gestures that are important for its detection. These gestures should be performed sequentially. Thus, once the gestural protocol was determined, the system will generate a particular event. Some of the implemented protocol are: forward walking, back-forward walking, stop walking, revolver, grasp, among others.

5 Conclusions and Future Works

Virtual Reality has become a commonplace in our increasingly technological world in recent years. An explosion has occurred in our understanding of Virtual Reality, Virtual Environments and in the technologies required to produce them in the last decade. Virtual Worlds and Virtual Environments are produced for people, for users to interact with computer and with complex information sets.

In Human-Computer Interaction, the development of avatars has significant potential to enhance realism, automation capability, and effectiveness across a variety of issues. As a form of communication, a theoretical model delivering new insights on the process of 3D avatar's design must consider verbal and non-verbal communication.

Particularly, NVC meets all the requirements of the HCI phenomenon, such as a functional, procedural, dynamic, irreversible, and meaningful character, and a presupposed systematicity which qualifies it for scientific study. Thus, non-verbal behaviour appears to be a primary factor in human-avatar interaction as much as it is for human-human interaction.

In particular, in human-avatar interaction, gestures add a welcome feeling of activity to the otherwise joyless ones of pointing and clicking. Gestural interfaces point out the invisible nature of gestures as a simple medium of NVC.

At present time, we are working on adding user technology that enables the use of human body as interface. This work is aimed to offer a form of natural interaction, based on the knowledge and practice of human beings through the several and numerous movements you can do with your hands.

The use of hands as an interaction medium requires a series of processes that naturally added immersion and interaction to the involved VR system.

Specifically, the development is based on the use of data gloves in order to incorporate attractive features for interaction within the virtual world. The conducted research and work is mostly based on gesture and real-time movements hand recognition. The included features are: hand tracking, navigation on stage and object manipulation.

The system allows the specification and recognition of dynamic gestures defined by sequences of dynamic and static hand poses (in order to expand the expressivity potential of the user and not to feel restricted). Each gesture has its own recognition process that tries to recognise the gesture using the incoming pose data. The recognition processes for each gesture operate in parallel.

The detected gestures are very easy to learn and do not require any special attention from the user.

By the moment, some systems aspects are beta version. Thus, future works will be oriented to improve environment issues, such as hardware platform. Even though the system allows a user interactivity in real-time, meaningful feedback and learning through an interface. We consider that the system should provide the necessary multi-VR media [23] and multi-platform structure and optimise the handling of 3D objects in the scenario using new libraries of physics or reprogramming of existing ones.

We think that the path leading to a natural and realistic inclusion of nonverbal communication in virtual environments is long and challenging, but crucial for the quality of interactions within these environments.

References

1. R. Van Eck. *Interdisciplinary Models and Tools for Serious Games: Emerging Concepts and Future Directions*. Premier reference source. Information Science Reference, 2010.
2. M. Chang, R. Kuo, Kinshuk, G.D. Chen, and M. Hirose. *Learning by Playing. Game-based Education System Design and Development*. LNCS sublibrary: Information systems and applications, incl. Internet/Web, and HCI. Springer, 2009.
3. J. Blake. *Natural User Interfaces in . Net*. Manning Pubs Co Series. Manning Publications Company, 2012.
4. Luis Hernández Ibez. Interfaces naturales para contenidos digitales interactivos en museos. la experiencia de galicia dixital. *Revista del Centro de Investigación de la Universidad la Salle*, 8(29):37–42, 2014.
5. Daniel Wigdor and Dennis Wixon. *Brave NUI world: designing natural user interfaces for touch and gesture*. Elsevier, 2011.
6. L. Sonneborn. *Nonverbal Communication: The Art of Body Language*. Communicating with confidence. Rosen Pub., 2011.
7. Adriana Peña Pérez Negrón and Angélica de Antonio Jiménez. Nonverbal communication to support collaborative interaction in collaborative virtual environments for learning. 2008.
8. M Fabri, DJ Moore, and DJ Hobbs. Expressive agents: Non-verbal communication in collaborative virtual environments. *Proceedings of Autonomous Agents and Multi-Agent Systems (Embodied Conversational Agents)*, 2002.
9. Marco Gillies, Mel Slater, et al. Non-verbal communication for correlational characters. 2005.
10. Mark L Knapp, Judith A Hall, and Terrence G Horgan. *Nonverbal communication in human interaction*. Cengage Learning, 2013.
11. Laura Dipietro, Angelo M Sabatini, and Paolo Dario. A survey of glove-based systems and their applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(4):461–482, 2008.
12. Martin G Helander. *Handbook of human-computer interaction*. Elsevier, 2014.
13. François Dominic Laramée. *Game design perspectives*. Charles River Media, Inc., 2002.
14. Francisco González, Pedro C Santana, Paulina Calderón, Aarón Munguía, and Martín Arroyo. Fallbox: a computer game with natural interaction through head tracking. *MexIHC 2010*, 2010.

15. Yikai Fang, Kongqiao Wang, Jian Cheng, and Hanqing Lu. A real-time hand gesture recognition method. In *2007 IEEE International Conference on Multimedia and Expo*, pages 995–998. IEEE, 2007.
16. MA Moni and ABM Shawkat Ali. Hmm based hand gesture recognition: A review on techniques and approaches. In *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pages 433–437. IEEE, 2009.
17. William R Sherman and Alan B Craig. *Understanding virtual reality: Interface, application, and design*. Elsevier, 2002.
18. Marcio S Pinho, Doug A Bowman, and Carla M Freitas. Cooperative object manipulation in collaborative virtual environments. *Journal of the Brazilian Computer Society*, 14(2):54–67, 2008.
19. Prashan Premaratne. *Human Computer Interaction Using Hand Gestures*. Springer, Singapore, 2014.
20. Antonio Camurri and Gualtiero Volpe. *Gesture-Based Communication in Human-Computer Interaction: 5th International Gesture Workshop, GW 2003, Genova, Italy, April 15-17, 2003, Selected Revised Papers*, volume 2915. Springer Science & Business Media, 2004.
21. Mohamed Alsheakhali, Ahmed Skaik, Mohammed Aldahdouh, and Mahmoud Alhelou. Hand gesture recognition system. *Information & Communication Systems*, 132, 2011.
22. <http://www.souvr.com/soft/uploadsoft/200808/2008081809050244.pdf>.
23. Y. Alvarado, N. Moyano, D. Quiroga, J. Fernández, and R. Guerrero. *Augmented Virtual Realities for Social Developments. Experiences between Europe and Latin America*, chapter A Virtual Reality Computing Platform for Real Time 3D Visualization, pages 214–231. Universidad de Belgrano, 2014.



XIII Workshop Tecnología Informática Aplicada en Educación (WTIAE)

Comité Científico:

Baldasari Sandra (España)
Cataldi Zulma (Argentina)
Coello Coello Carlos (México)
Collazo César (Colombia)
Cukierman Uriel (Argentina)
Dapozo Gladys N. (Argentina)
Depetris Beatriz (Argentina)
Gonzalez Carina (España)
Madoz Maria Cristina (Argentina)
Malberti Riveros María Alejandra (Argentina)
Manreso Cristina (España)
Rueda Sonia (Argentina)
Russo Claudia (Argentina)
Sanz Cecilia (Argentina)



XIII Workshop Tecnología Informática Aplicada en Educación (WTIAE)

ID	Trabajo	Autores	
9101	Assessing Trust in a Classroom Environment, applying Social Network Analysis	Antonieta Kuz (UTN), Mariana Falco (UTN), Roxana S. Giandini (UNLP)	304
9115	Modelo del estudiante para la Evaluación Académica Usando Técnicas de Lógica Difusa	Constanza Huapaya (UNMdP), Delia Esther Benchoff (UNMdP), Marcela Gonzalez (UNMdP), Francisco A. Lizarralde (UNMdP), Cristian Remón (UNMdP), Leonel Guccione (UNMdP)	314
9120	Incorporación de herramientas de trabajo colaborativo en el entorno AulasWeb: una propuesta de modificación del selector de actividades	Alejandro Gonzalez (UNLP), Maria Cristina Madoz (UNLP), Leandro Romanut (UNLP)	324
9141	Evaluación y medición de la calidad percibida de MOOC	Marisa Tumino (UAP), Juan Manuel Bournissen (UAP)	334
9154	El discurso en entornos virtuales de aprendizaje: categorías de análisis de los desempeños escritos	Néstor Blanco (UNLZ), Nora Cuello (UNLZ), Gabriela Gonzalez (UNLZ), Paula Penco (UNLZ)	344
9166	Herramientas de Autor para la creación de Objetos de Aprendizaje. Estado del arte	Lucía Violini(UNLP), Cecilia Sanz (UNLP)	353
9172	Modeling Students through Analysis of Social Networks Topics	M. Emilia Charnelli (UNLP), Laura Lanzarini (UNLP), Javier Diaz (UNLP)	363
9191	Metodologías de Ensamblaje de Objetos de Aprendizaje. Análisis bajo la lupa del concepto de Objetos de Aprendizaje subyacente.	Gustavo Javier Astudillo (UNLPAM), Cecilia Sanz (UNLP), Liliana Patricia Santacruz-Valencia (URJC)	372
9195	Bot guía para un ambiente virtual inmersivo	Alejandro Altube (UM), Iris Sattolo (UM), Liliana Lipera (UM)	382



ID	Trabajo	Autores	
9208	Transformación de Exámenes de Ambientes E-Learning en Videojuegos Web	David Curras(UAI), Mariano Davobe (UAI), Alejandro Sartorio (UAI), Marcelo Vaquero (UAI), Luciano Diamand (UAI)	392
9260	Integrando modelo de aprendizaje supervisado al análisis del desempeño de alumnos en cursos virtuales sobre plataformas Moodle	Rodrigo Jacznik (UTN FRLP), Mariano Tassara (UTN FRLP), Ivan DUva (UTN FRLP), Guillermo Baldino (UTN FRLP), Roxana Giandini (UTN FRLP), Leopoldo Nahuel (UTN FRLP)	404
9283	Modelo de Proceso para Uso Didáctico en la Enseñanza de Ingeniería del Conocimiento	Ezequiel IScordamaglia(UNLA), Dario Rodriguez (UNLA), Hernán Amatriain (UNLA), Ramon Garcia Martinez (UNLA)	414
9285	Conjunto de Herramientas de Ayuda a la Presencialidad	Dario Rodriguez (UNLA), Sebastian Martins (UNLA), Ezequiel Baldizzoni (UNLA), Santiago Bianco (UNLA), Ramon Garcia Martinez (UNLA)	423
9302	Modelo de seguimiento de una actividad de foro en la enseñanza universitaria. Aplicación en un estudio de caso.	Cecilia Sanz (UNLP), Maria Alejandra Zangara (UNLP), María Paula Dieser (UNLPAM)	431
9309	4WindGame: Un juego de Armado de Piezas Asistido con una impresora 3D	Alejandro De Castro (UAI), Federico Brussa (UAI), Alejandro Sartorio (UAI), Marcelo Vaquero (UAI), Daniel Tedini (UAI)	441
9315	Sistema para el Control, Administración y Programación de Robots Educativos	Germán Leandro Osella Massa(UNNOBA), Cecilia De Vito (UNNOBA), David Fernandez (UNNOBA), Monica C. Sarobe (UNNOBA), Claudia Russo (UNNOBA)	452

Assessing Trust in a Classroom Environment, applying Social Network Analysis

Antonieta Kuz¹, Mariana Falco¹, Roxana Giandini^{1,2}

¹ LINSI, Facultad Regional La Plata, UTN
60 esq. 124, La Plata, Buenos Aires, Argentina
{akuz, mfalco}@frlp.utn.edu.ar

² LIFIA, Universidad Nacional de La Plata
50 esq. 120, La Plata, Buenos Aires, Argentina
giandini@lifia.info.unlp.edu.ar

Abstract. This paper builds the study of social networks and their bonds with trust, through the methods of detection and description of structural properties. Networks of relationships can be mapped and managed to enhance organizational learning. Social Network Analysis (SNA) is detecting and interpreting patterns of social ties among actors. Due to trust is a crucial factor among the interaction between actors, we use NodeXL© tool to perform a case study in a classroom environment. We relate the trust measures to the clustering coefficient, centrality, and strong and weak ties which are computable characteristics of a social network. We find that more connected classes, and classes with stronger friendship relationships, enjoy more trust.

Keywords: Social Network Analysis, trust, NodeXL©, Classroom Climate, Graphs, Students.

1 Introduction

The interest in Social Networks has been increasing and evolving across a wide variety of fields and researches, such as Physics, Psychology and Computer Science. Social Network Analysis (SNA) was developed in a relatively non-technical manner from the structural concerns. Anthropologists like Radcliff-Brown started to develop a concept of social structure and a web of social life. Social networks have also been studied by Milgram's small world research [1]. SNA is a methodological tool for the analysis and study of Social Networks. The main goal of SNA is detecting and interpreting patterns of social ties among actors [2].

The specific objectives of this research lead to introducing of certain elements that help establish the conceptual frame-work of the social network. Moreover, we introduce a methodological tool for the practical analysis of social network in the classroom environment. In order to make this proposal practical, we present a case study developed in a tool called NodeXL© [3] that allowed us to create visualizations of social networks and to assist us in our analysis. In particular, we present and explore trust structures in social networks, and the influence in the network of the

weak bounds. Trust is a component which acts under uncertainty in many social interactions. We will intend to define the qualitative way of relations between different actors and their consequences under trust and weak bounds. The advantage of the use of NodeXL© is that it permits a qualitative analysis by studying networks over patterns and visualizations, that makes it easy and intuitive.

The remainder of this paper is structured as follows. Firstly, in the related work section, we describe the state of art of SNA and trust among social relations. Then, the main structure and basic concepts within the social networks are described. This section is followed by the explanation of trust and its powerful relation with SNA. Then, we contextualize NodeXL©, a powerful SNA tool which allows us to perform the study case in the next section. Also, a study of the trust among ties in a classroom through different metrics is detailed. Then, we present the results. Finally, in the last section, we describe the main conclusions and the future work.

2 Related work

Several approaches have been published about Social Network Analysis, trust and weak bounds. Among others, the work of Robert A. Hanneman [4] considers Network Analysis as a method for describing and analyzing a web of links among entities, including people. Marc A. Smith [5] adds SNA features to the familiar Excel spreadsheet with NodeXL©. Formally, a social network is formed by actors and their relationships which their content, as may be trust. Indeed, Golbeck [6] considers trust is a concept in computing systems while Sibel Adali [7] treats trust as a social tie between a trustor and trustee. In general, the difference between these proposals and ours is that we are suggesting the association between SNA and trust for a specific environment, such as a classroom, in a practical way.

Weak ties, introduced in the paper written by Granovetter [8] refer to relationships with acquaintances outside their social circle. He studied the tie between the jobseeker and the person who supplied the information that led to the job. Granovetter also estimated that new information comes via weak ties, those that are local bridges. He showed that the weak ties play an important role in the dissemination of trust, because strong ties by themselves generate fragmentation, as subgroups in a community become isolated from each other, and weak ties allow for student community integration, connecting these subgroups.

3 SNA and Trust

Newman [9] defines a social network as “*a set of people or groups of people with some pattern of contacts or interactions between them*”. Part of SNA [10] studies all the interactions between individuals and organizations, and flows of information. The analysis of the relations between actors allows delimiting the dynamics of flow circulation between actors located in different places in the network. Most of the bounds among actors have a purpose or may have it because there is some interest. A

trust network is a social network in which persons are connected via explicit trust relationships.

3.1 Trust

Castelfranchi & Falcone [11] suggest that it's crucial to distinguish at least between two kinds and meanings of trust: (a) Trust as psychological attitude of X towards Y relative to some possible desirable behavior or feature. (b) Trust as the decision and the act of relying on, counting on, depending on Y. Also, they say that there are five inter-related categories: the construct, the trustee, actions and behaviors, results and outputs of behavior and the risk.

Also, with the growth of Internet, Facebook is gaining ground over other social networks. People share and reflect their whole lives online and become more comfortable, sometimes even blending their personal and professional life. Through the Facebook's privacy, users can choose who are going to include in their list of friends and who does not. In this context, it is one of the great challenges be able to assess what information to trust and who to trust [12].

Trust is a dynamic phenomenon defined as a person's willingness to be vulnerable in order to obtain benefits from someone or something. For example, trust develops between two people through shared experiences and exchange of favors, their exchanges of lower risk before engaging in higher risk exchange. In a social network, when two actors are directly connected they may have a certain degree of confidence, but for actors that are not directly connected there does not exist trust information by default. However, it is possible to obtain information and knowledge from those links that connect the network and this can be used for inferring how much one may trust others. Trust relationships are special types of social relationship. Trust edges are weighted with numeric values.

3.2 Practical trust measures

Trust is rather than a purely cognitive or emotional phenomenon a facet of the relationship and the trust construction requires reciprocal self-disclosure and affirmative responses. In a social network each user has his/her friends. Close friends have closer ties and hence higher trust, so our approach takes into consideration the social ties and trust in the social network [13]. Thus, SNA brings the explanation of behaviour of relations that requires an analysis of how the actors are connected to one another considered in a particular environment with contextual factors [7]. We now present structure-based metrics that support the identification of trustworthy individuals. We note that these metrics provide critical cues about trust, but they are not the only indicators of trustworthiness.

Clustering Coefficient. Is the ratio of how many actual edges there are between neighbours to how many potential edges that are between neighbours. To determine the diameter of a network [14], we search the distance between every pair of vertices. The diameter is equal to the greatest distance. Also is defined as the actual number of

connections among his neighbours divided by the maximum potential number of such connections.

Centrality. It contributes to our understanding of communication flows within social structures [15]. In general, centrality refers to actors who are in a central position within a social network. The centrality measure on which this research focuses is closeness centrality due to its function as a global evaluator of centrality. As a measure of reachability, closeness is a centrality measure that explains an actor's ability to easily connect to others. That is, closeness centrality explains the ties within a network placing an actor in a central location such that the persons access to information and could potentially pass that information along to others people. The average of the direct and indirect links to all others in the network determines an actor's closeness centrality [16].

Strong and weak ties. In SNA's terms, the ties or interpersonal ties are defined as information-carrying connections between people or individuals. The weak ties are more present in social networks in society as well as those most responsible for the transmission of information through these networks. Mark Granovetter [8] pointed out that the emphasis on weak ties lends itself to discussion of relations between groups and analyse segments of the social structure that are not easily defined in terms of unit groups.

The most of intuitive notions of the strength of an interpersonal tie should be satisfied by the following definition: the strength of a tie is a (probably linear) combination of the amount of time, the emotional intensity, mutual confiding and the reciprocal services which characterize the tie. The strength of the interpersonal ties can be defined as the combination of the amount of time spent in interaction, the emotional intensity, the intimacy and the reciprocal services which characterize the tie. Individuals are sometimes influenced by others with whom they or tenuous or even random relationships.

The influences are labelled weak ties. Weak ties are assumed to be responsible for dissemination of information, effective role in social cohesion. So, individuals with many weak ties are best placed to diffuse ideas of those ties ill form local bridges between different networks. Weak ties appear very suited for the transfer of knowledge, but obtaining an effective transfer of knowledge requires extensive personal contact and trust. Transferring trust is less difficult through strong ties. The strength of the relations can be indicated by its multiplicity, that appears because actors are bound by different types of ties, and this make actors stay close to each other by reinforcing their bounds.

3.3 A tool called NodeXL

NodeXL© is an extensible toolkit for network overview, discovery and exploration implemented as an add-in to the Microsoft Excel 2007 spreadsheet software. Thus, is a tool for network interactive visualization that leverages the widely available Excel application, as the platform for representing generic graph data, performing advanced

network analysis and visual exploration of networks [17]. We used this software tool, which allows us to show how a classroom is from within, by using certain metrics from SNA. NodeXL© is designed to enable Excel users to easily import, to clean, analyse and visualize network data. Extends the existing graphing features of the spreadsheet with the added chart type of “network”, thus lowering the barrier for adoption of network analysis [5].

4 Case Study: Network mapping and data extraction strategy

Because trust is a social construct, it is valid to apply this concept to various fields. One of them is the classroom environment. The classroom is a learning space in which students take the first steps into what is known as socialization. In this area are conducted countless relationships, in which ties between students are created. Furthermore, the classroom environment allows to measure social trust. Indeed, trust give rise to cooperation and directly affects the effectiveness and success of teamwork.

We use this case study to show how the combination of social network and qualitative analysis can offer interesting evaluation of trust and also, design data extraction forms to accurately record the information that researchers obtain from the primary studies. In order to apply these concepts in the classroom, it is worth emphasizing certain topics that make entanglement between SNA and ties in the classroom. To represent the classroom, we use graphs in which each node (vertex) represents a student. We consider a class consists of 30 students and a teacher named Mr. Hugh. Therefore, the dataset implemented for the present study case is as follows: Abraham, Adam, Adele, Benjamin, Bonnie, Brooke, Carol, Cedric, Connor, Danielle, Daphne, Diana, Edmond, Ellen, Elsa, Frankie, Frederic, Gavin, Georgia, Harry, Helen, Jeremy, Katy, Madison, Maggie, Maxwell, Olivia, Ralph, Randy and Rose.

It is feasible to mention that to protect the privacy of the students involved, we change the real names with fancy labels names. Furthermore, it was presented to the students a set of questions such as "*Who would you choose to study for the test?*" or "*With whom would you sit in the classroom?*", among others, in order to indicate how they feel in relation to their peers, and any other relevant relational information such as strength of the tie. These types of test allow us to determine the degree to which individuals are accepted or rejected in a group, discover relationships between individuals and reveal the structure of the group itself. The questions may be of selection or perception type, which in turn, each may be positive or negative.

We have selected the negative questions because we can use them to differentiate between the rejected and abandoned children, to identify the social polarization of a large number of rejections, to determine the leaders and subgroups. In other words, all of these concepts fulfilled a method of social information retrieval. The content of these questions will allow us to know classroom climate and then determine the levels of trust that exist there. A good education individually and/or collectively is directly related to the human future and/or communities. In this context, it has been aimed to relate trust concept with teaching. Mitchell and Forsyth [18] claim that trust has significant implications for everyone connected with schools, and trust can be a vital

We are using a principle known as “*triadic closure*” in which, if nodes B and C have a friend A in common, then the formation of an edge between B and C produces a situation in which all three nodes A, B, and C have edges connecting each other — a structure with reference to a triangle in the network. The term “*triadic closure*” comes from the fact that the B - C edge has the effect of “closing” the third side of this triangle [19]. Considering that the clustering coefficient is the fraction of pairs of neighbors who are themselves neighbors, it is feasible to mention the following example. The clustering coefficient of node A (Adele) in Figure 1 is 1/17 (because there is only the single B-C (Georgia-Gavin) edge among the pairs of friends). In general, the clustering coefficient of a node ranges from 0 (when none of the node’s friends are friends with each other) to 1 (when all of the node’s friends are friends with each other), and the more strongly triadic closure is operating in the neighborhood of the node, the higher the clustering coefficient will tend to be.

Through the analysis of trust by the triadic closure concept, it is achievable to delimitate the values related to each tie of the triangle. Thus, Gavin trusts Adele with 0.6; Adele trusts Georgia with 0.7; and from the tie needed to make the triangle closure we can say that Georgia trusts Gavin with 0.5. And this is the tie, the link that allows the flow of communication among them. Therefore, the structure of a classroom environment depends on the existence of trust relations between the members and the level of trust they represent.

As such we pointed out that one reason why Georgia and Gavin are more likely to become friends, when they have a common friend Adele, is simply based on the opportunity for Georgia and Gavin to meet: if Adele spends time with both Georgia and Gavin, then there is an increased chance that they will end up knowing each other and potentially becoming friends. A second, related reason is that in the process of forming a friendship, the fact that each of Georgia and Gavin is friends with Adele (provided they are mutually aware of this) gives them a basis for trusting each other that an arbitrary pair of unconnected people might lack.

4.2 Centrality: Results.

One property of closeness centrality is that it tends to give high scores to individuals who are near the center of local clusters (aka network communities), in an overall larger network. Students who are highly connected to others within their own cluster will have a high closeness centrality. This condition is shown in Figure 1, in which the students that obtain a high value are Elsa and Adam. Actors who are trusted will be more often used as information transfer conduits. Students possessing high levels of closeness centrality will maintain those positions as long as trust levels do not decline. On this occasion, there is no incentive from a vulnerability standpoint to find alternate routes of communication. Subsequently, actors are likely to utilize the most efficient communication routes through these trusted actors.

Schein's original definition of centrality is that centrality is “*denotes the person's objective position as measured by the degree to which company secrets are entrusted to him, by ratings of others of his position, and by his actual power*” [20]. The Schein's definition provides two aspects of interest to the present discussion. First, centrality involves trust (see Fig. 1). Without trust, one will likely have a decrease in

his or her ability to successfully occupy a central position in a network. In addition, centrality is, in part, a function of "*ratings of others of his position*". Thus, an actor's centrality is partially a function of his or her ability to be trusted, which in turn, leads to others rating the actor as a central player, such as Adam and Elsa (see Fig. 1).

The six directed ties to Adam present the following trust values (see Fig. 1). Ellen trust Adam with 0.7, Helen with 0.8; Danielle with 0.3; Carol with 0.6, Ralph with 0.7 and Katy with 0.5. Distinguish the fact that the first trust bond Adam may lose if something may happen is Danielle because she had the lowest value among the group that chooses him directly. By comparing to the subgroup leads by Elsa, we find that is a truly united and cohesive group of students. The trust values of all the direct ties to Elsa show us that they really trust in their leader. Consequently, whether a person trusts a given leader depends heavily on whether there exist intermediary personal contacts who, from their own knowledge, assure him that the leader is trustworthy, and who can, if necessary, intercede with the leader or his lieu-tenants on his behalf. Trust in leader is integrally related to the capacity to predict and affect their behavior [8]. Parenthetically, it is significant to mention that the made selections by both Adam and Rose possess high levels of trust. Thereby, Adam trust Olivia with 0.8 and Elsa trust Rose with 1.0. Second, the phrase "*secrets are entrusted to him*" denotes the transfer of confidential information to those who possess centrality. These conditions are precisely those necessary to create the context for successful opportunism. Therefore, in low trust situations, opportunism is perceived to be a significant threat, otherwise central actors will not occupy a position of centrality. When this is the case, closeness centrality marks the most efficient routes of communication. Closeness centrality measures actual routes, but not necessarily the most efficient routes.

4.3 Strong and weak ties: Results.

Considering two arbitrarily selected individuals – call them A and B, and the set S composed of all students with ties to either or both of them – call them C, D, E, etcetera. As Granovetter [8] pointed out in his research, the hypothesis which enable us to relate dyadic ties to larger structures is: the stronger the tie between A and B, the larger the proportion of individuals in S to whom they will both be tied, that is, connected by a weak or strong tie. This overlap is predicted to be least when their tie is absent, most when it is strong, and intermediate when it is weak. From the proposed relationship, it is feasible to specify that stronger ties involve large time commitments.

In consideration of Figure 1, we are considering Adam and Olivia as 'A' and 'B' respectively, and Maxwell as the 'C' student. Therefore, if Adam-Olivia and Adam-Maxwell ties exist, then the amount of time Maxwell spends with Olivia depends (in part) on the amount that Adam spends with Olivia and Maxwell, respectively. If Maxwell and Olivia have no relationship, common strong ties to Adam will probably bring them into interaction and generate one. Implicit here is idea that the more frequently persons interact with another, the stronger their sentiments of friendship for one another are apt to be. As such, the stronger the tie connecting two individuals, the more similar they are.

Weak ties are crucial in binding groups of strong ties together. They bring circles of networks into contact with each other, strengthening relationships and forming new bonds between existing relationship circles. As such, in Figure 1, even though there is no direct tie between Georgia and Olivia, it exists a mediator that allows the flow of information between them: Adele. The relationship of the students with their weak ties should be maintained and cultivated, knitting them networks together to encourage information free flow between the different parts of the networks. One of the most remarkable feature in Figure 1 is the presence of a bidirectional tie between Elsa, one of the leaders, and Rose. We emphasize the structure within this tie because is the only case among both subgroups in which two students choose one to another. This type of relationship is feasible be seen when the contributing nodes are friends. In contrast, the weak ties are just the acquaintances.

Conclusions

According to constructivist approach, it is important to use the techniques which allow student-teacher and student-student interaction such as discovery learning, cooperative learning, and team works. It is clear that to build teams in a class first, a trusting environment should be created. In this research we present some measures that have different meaning and give important information for the network analysis. These metrics are likely to help analysts identify trustworthy information more quickly than manual inspection, and more generally contribute to the science of representing the behavior of students interacting through social networks. It is feasible, through NodeXL©, performing the trust analysis within the network and to elaborate the results from three metrics, such as clustering coefficient, centrality, and weak and strong ties. We find that trust levels are higher in more connected communities as may be a classroom.

When trust is absent, people are reluctant to work closely together, and collaboration is more difficult. Consequently, trust promotes communication, motivation, cooperative behavior within classes and it obtains a healthier school climate. For efficient team members reciprocal and approximately equal trust levels should be preferred. Also, counselors may use these networks as signals and take necessary precautions for risky personal relationships. Trusting relationships among classmates promote knowledge exchange. In addition to formal learning activities during school time, learning process continues via these informal knowledge exchanges. For extending this research, we will consider other metrics and we will compare the trust levels of different classes, which allow us to determine the various climate of the classroom. Future work may also consider comparing the contributions of the metrics to trustworthiness or information value metrics generated by human experts, and refining the metrics by including various aspects of expert judgments.

References

1. Milgram, S. (1967), 'The Small World Problem', *Psychology Today* 2, 60–67.

2. Searle, J. R. (1995) *The Construction of Social Reality*. The Free Press
3. Hansen, D., Shneiderman, B. & Smith, M. A. (2010), *Analyzing Social Media Networks with NodeXL: Insights from a Connected World*, Morgan Kaufmann Publishers Inc., San Francis-co, CA, USA.
4. Hanneman, R. A., ed. (1988), *Computer-assisted theory building: modeling dynamic social systems*, Sage Publications, Inc., Thousand Oaks, CA, USA.
5. Smith, M., Hansen, D. & Gleave, E. (2009), *Analyzing Enterprise Social Media Networks*, in 'Computational Science and Engineering, 2009. CSE '09. International Conference on', Vol. 4, pp. 705–710.
6. Golbeck, J. (2009), 'Trust and nuanced profile similarity in online social networks', *ACM Trans. Web* 3(4), 1–33.
7. Adali, S. (2013), *Trust as a Computational Concept*, in 'Modeling Trust Context in Networks', Springer Briefs in Computer Science, Springer New York, pp. 5–24.
8. Granovetter, M. (1973), 'The Strength of Weak Ties', *The American Journal of Sociology* 78(6), 1360–1380.
9. Newman, M.E.J. (2010) *Networks, An Introduction*. Oxford University Press.
10. Scott, J. (2000), *Social Network Analysis: A Handbook*, Sage Publications.
11. Castelfranchi, C., & Falcone, R. (2010). *Trust theory: A socio-cognitive and computational model* (Vol. 18). John Wiley & Sons.
12. Alesina, A., La Ferrara, E. (2000) Who trusts others? *Journal of Public Economics* 85, pp. 207-234, Elsevier.
13. Mobius, M., Szeidl, A. 'Trust and cooperation in social networks'. European Summer Symposium in Economic Theory (ESSET). Study Center Gerzensee, 3-14 July 2006
14. Diestel, R. (2005), *Graph Theory*, Springer.
15. Droege, S., Dong, L.S. (2007) 'The centrality efficiency index: a new social network analysis measure'. *Review of Business Research*, International Academy of Business and Economics ISSN: 1546-2609.
16. Freeman, L. C. (1979) 'Centrality in Social Networks: I. Conceptual Clarification. *Social Networks*, Vol 1, pp 215-239.
17. Sorin, A. M. (2011), 'Analyzing Social Media Networks with NodeXL: Insights from a Connected World by Derek Hansen, Ben Shneiderman, and Marc A. Smith', *Int. J. Hum. Comput. Interaction* 27(4), 405–408.
18. Mitchell, R.M. and P.B. Forsyth, (2004). 'Trust, the Principal, and Student Identification', UCEA Conference Proceedings for Convention, David C. Thompson and Faith E. Crampton, Editors, The Paper Presented at the Annual Meeting of The University Council for Education Administration Kansas City, Missouri.
19. Easley, D., Kleinberg, J. 'Networks, Crowds, and Markets: Reasoning about a Highly Connected World'. Chapter 3. Cambridge University Press, 2010. Complete preprint online at <http://www.cs.cornell.edu/home/kleinber/networks-book/>
20. Schein, E. H. 'The individual, the organization, and the career: A conceptual scheme', *Journal of Applied Behavioral Science*, Vol 7, pp 401-426, 1971.

Modelo del estudiante para la Evaluación Académica Usando Técnicas de Lógica Difusa

Constanza Huapaya¹, Leonel Guccione¹, Guillermo Lazurri¹,
Delia Esther Benchoff¹, Marcela Gonzalez², Francisco Lizarralde¹ y Cristian
Remon¹

¹Fac. de Ingeniería, Dpto. de Matemática, Juan B. Justo 4302

²Fac. De Psicología, Complejo Universitario - Funes 3250

Universidad Nacional de Mar del Plata, Mar del Plata, Argentina

{constanza.huapaya,leonel.guccione,francisco.lizarralde,ebenchoff,remoncristian}
@gmail.com, mpgonza@mdp.edu.ar

Abstract. En este artículo se presenta un modelo del estudiante híbrido cuyo objetivo es proponer perfiles cognitivos de estudiantes de ingeniería. Estos perfiles se construyen en base a dos variables lingüísticas: *modalidad de aprendizaje* y *nivel de conocimiento*. Estas variables se estiman a partir de datos obtenidos con dos modelos: personalización (estilos de aprendizaje de Felder y perturbación). A partir de los estilos de aprendizaje, mediante un sistema de inferencia difusa, se obtiene la *modalidad de aprendizaje*. El modelo de perturbación se basa en el dominio a enseñar formado por tópicos de granularidad media, estructurados como un árbol. El progreso del conocimiento de cada estudiante sobre cada tópico es manipulado con modelos difusos. La información dada por estos conjuntos, usando un sistema de inferencia difusa, se obtiene la variable *nivel de conocimiento* basado en valores de estereotipos clásicos. Para ilustrar el modelo propuesto en este artículo se describe un ejemplo sobre un tema específico.

Keywords: modelo del estudiante, evaluación académica, lógica difusa, estudiante de ingeniería.

1 Introducción

El avance de la tecnología computacional ha provocado que el aprendizaje basado en computadora haya crecido en forma sostenida en la última década [1]. Sin embargo, todavía existen problemas cuando se comparan estos sistemas de enseñanza/aprendizaje con las clases impartidas en el aula. Algunos de los inconvenientes que se presentaron son la falta de soporte adaptativo para sostener la flexibilidad en la representación de los usuarios.

La adaptación busca facilitar el aprendizaje ajustándose a las características individuales, estimar el avance e inferir el nivel de conocimiento del estudiante. Un enfoque exitoso es el *modelado del estudiante* [2]. Self [3] plantea que éste es un proceso orientado a representar aspectos cognitivos del estudiante tales como el análisis del rendimiento, examinar sus errores, representar sus objetivos, identificar el

conocimiento previo y adquirido, y describir sus características personales. En consecuencia, un factor crucial para el diseño de un sistema educativo es la construcción de un modelo que represente efectivamente al estudiante. El modelado del estudiante puede definirse como el proceso de reunir información relevante a fin de inferir el estado cognitivo actual y representarlo de manera tal que sea accesible y útil para la adaptación.

Como resultado de este proceso se define un *modelo del estudiante*. Este modelo involucra una tarea compleja que se caracteriza por el estudio de objetos intangibles como la personalidad, habilidades cognitivas y preferencias individuales que influyen en el aprendizaje, esto es, un modelo del estudiante sirve para representar y explorar las características cognitivas individuales del estudiante [2,4].

El estado cognitivo no puede ser observado y medido directamente como la altura de una persona. Durante la adquisición de la información para estimarlo aparecen algunas restricciones como la incertidumbre y datos incompletos (la información sobre la actividad del estudiante que se recoge varía en cuanto a cantidad y calidad). Esta incertidumbre, es causada principalmente, por la naturaleza dinámica del aprendizaje así como por el enfoque interpretativo que hace el docente sobre la actividad del estudiante [1]. A pesar de las dificultades señaladas, el diseño del modelo del estudiante debería orientarse a la administración de datos que permita manipular (analizar e inferir) el estado cognitivo sobre datos crudos (notas, tiempos de las respuestas, evaluaciones especiales, etc). La Lógica Difusa es una de las herramientas más idóneas para manipular la incertidumbre [5,6]. En [7] se aprecia un modelo del estudiante definido con Lógica Difusa. Como datos relevantes que puede contener nuestro modelo del estudiante se encuentran el estilo de aprendizaje y el logro alcanzado por el alumno. Esta información es la base para la toma de decisiones en el diagnóstico cognitivo.

2 Descripción del modelo

2.1 Representación del conocimiento y su evaluación

El primer paso dado en la construcción del modelo fue definir un subconjunto de tópicos del dominio a enseñar/evaluar. Estos fueron organizados en una estructura de árbol donde el nodo raíz es el tema central a enseñar y los sucesivos hijos fueron subdivisiones de los tópicos hasta llegar a las hojas que representan los temas que serán tratados para su posterior evaluación con pruebas especiales. Un ejemplo de este árbol se aprecia en la figura 1. El dominio elegido fue el Lenguaje de programación assembler, del cual solo se propusieron algunos tópicos. Estos tópicos son elegidos por el experto en el área que se desea estimar el diagnóstico cognitivo.

Luego de incorporar los resultados de las pruebas, en el modelo overlay puro [8] cada nodo del árbol es marcado como aprendido o no aprendido. El modelo overlay representa el conocimiento del estudiante bajo la hipótesis de que un estudiante posee conocimiento parcial, pero correcto con respecto al área del conocimiento en estudio. La diferencia entre el conocimiento del experto y del estudiante es la falta de habilidades y conocimiento del estudiante. La finalidad de este modelo es tratar de

eliminar esa diferencia tanto como sea posible. El dominio se descompone en un conjunto de elementos (que pueden ser conceptos o tópicos).

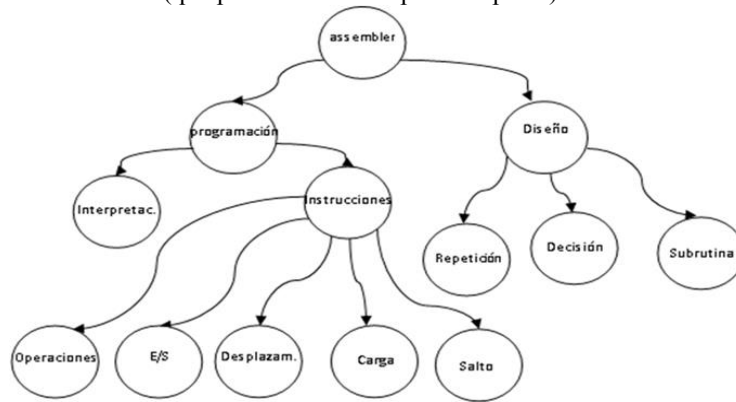


Fig. 1. Representación parcial del dominio en forma de árbol.

En su versión moderna, el modelo representa el grado de dominio que tiene el estudiante sobre cada tópico usando una medida cualitativa. En particular puede utilizarse un modelo de perturbación [9]. Este modelo incorpora el conocimiento incorrecto del estudiante.

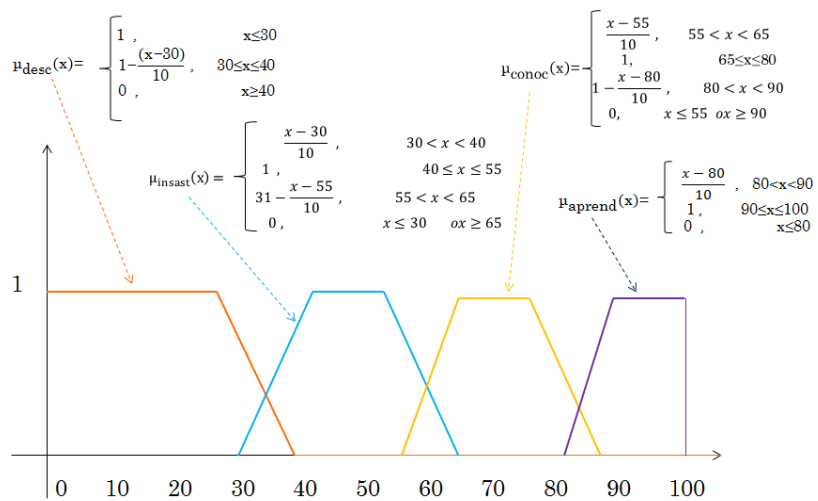


Fig. 2. Ejemplo de funciones de pertenencia respecto al nivel de conocimiento en cada nodo

Por ejemplo, para establecer la medida cualitativa de un modelo de perturbación, pueden utilizarse los siguientes cuatro conjuntos difusos a fin de describir el conocimiento del estudiante en cada nodo del dominio: desconocido, insatisfactoriamente conocido, conocido y aprendido. Las funciones de pertenencia a

cada conjunto y sus gráficos correspondientes se aprecian en la figura 2. Las calificaciones de las pruebas varían entre 0 y 100. A cada nodo se le asocia una 4-upla formada por los valores de cada una de las funciones de pertenencia descritas más arriba: $(\mu_{desc}(x), \mu_{insast}(x), \mu_{conoc}(x), \mu_{aprend}(x))$ a fin de expresar el conocimiento del estudiante sobre el concepto en evaluación, esto es, para cada valor de x , se evalúan las cuatro funciones de pertenencia.

Para ilustrar su uso en un tópico como “estructura de control de decisión” se presentan los ejemplos expuestos en la figura 3. En 3(a) se expresa que el estudiante no aprendió el tópico; en 3(b) posee un 20 % de conocimiento insatisfactorio, un 20% de conocimiento y 60 % de aprendizaje en el tópico y en 3(c) se considera que el aprendizaje de la estructura de decisión es completo.

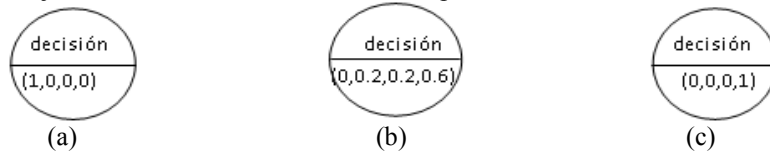


Fig. 3. Ejemplos de 4-uplas en la evaluación de un tópico

En la figura 4 se aprecia el modelo de perturbación en un instante de la evaluación donde se muestran los tópicos del dominio con sus correspondientes 4-uplas. Las hojas del árbol están en tono gris y son las evaluaciones de los estudiantes examinados con diversas pruebas. Luego, con un programa específico de cálculo que se ha desarrollado, se activa el resto de los elementos del árbol hasta llegar al nodo central (esto es, assembler). La activación se ha calculado, en este caso, como promedios de los hijos de cada nodo.

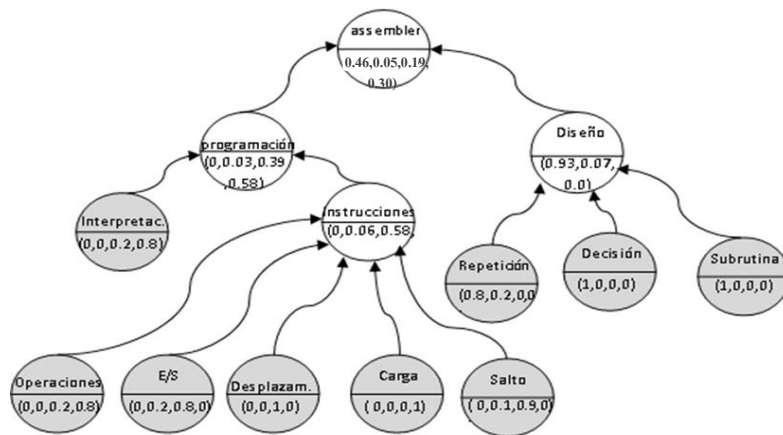


Fig. 4. Ejemplo del modelo de perturbación para un estudiante

2.2 Núcleo del modelo del estudiante

Para estimar el diagnóstico cognitivo se ha diseñado un núcleo del modelo del estudiante compuesto por dos módulos: un *modelo de perturbación* y un *modelo de personalización* (ver figura 5). Como ya vimos, el primero de ellos actualiza su contenido a medida que el sistema recibe nueva evidencia, i.e., nuevos datos sobre la evaluación del estudiante. La información producida por este modelo interactúa con el modelo de personalización (los estilos de aprendizaje y estereotipos de cada alumno) para construir un perfil cognitivo dinámico. El estereotipo de cada estudiante se modifica también con la nueva información producida en las sucesivas iteraciones.

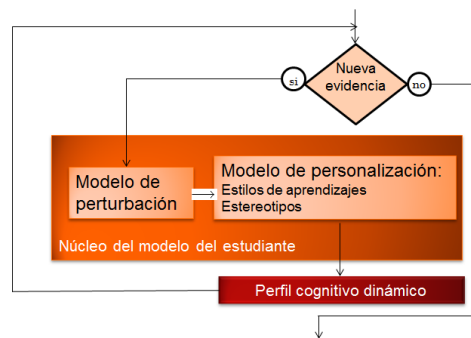


Fig. 5. Modelo del estudiante

El *modelo de personalización* se basa en los estilos de aprendizaje. Un estilo de aprendizaje es el modo que caracteriza a un individuo cuando adquiere, retiene y recupera información. Los estudiantes muestran diferentes fortalezas y preferencias cuando adquieren conocimiento. Esto es, poseen distintos estilos de aprendizaje. Aquí se ha adoptado el modelo de Felder [10,11,12], principalmente, porque su investigación está dirigida a los estudiantes de ingeniería. Este autor conjuntamente con B. Soloman [13] creó un cuestionario en línea para estimar los estilos de aprendizaje. Se ha tomado el cuestionario a los alumnos de asignaturas de las carreras de ingeniería de la Universidad Nacional de Mar del Plata y procesado los resultados en la página de la Universidad del Estado de North Carolina. El instrumento en línea estima las preferencias en cuatro dimensiones (activo/reflexivo, sensitivo/intuitivo, visual/verbal, y secuencial/global). Se han seleccionado dos dimensiones Procesamiento de la Información (Activo/ Reflexivo) y Entendimiento (secuencial/global) por considerarlas más apropiadas a la naturaleza de las asignaturas elegidas (ciencias básicas). La fundamentación de esta elección es la siguiente:

Estilo de aprendizaje Activo/ Reflexivo: esta dimensión distingue entre modos activos y reflexivos del procesamiento de la información por parte del alumno. Los estudiantes activos aprenden mejor trabajando activamente con el material de aprendizaje, resolviendo problemas planteados y probando nuevas soluciones. Por el contrario, los alumnos reflexivos se inclinan por pensar y reflexionar sobre el material de aprendizaje.

Estilo de aprendizaje Secuencial/Global): en esta dimensión los estudiantes son caracterizados de acuerdo a su entendimiento. Los estudiantes secuenciales aprenden con pequeños pasos incrementales y por lo tanto tienen un progreso lineal del aprendizaje. Tienden a seguir caminos lógicos paso a paso para encontrar la solución. Por el contrario, los alumnos globales usan una modalidad holística de pensamiento y aprenden a saltos. Como para los estudiantes globales es importante la visión global, a estos les interesan los resúmenes y el conocimiento amplio, mientras que a los estudiantes secuenciales están más interesados en los detalles.

Cada dimensión posee un rango que varía de 0 a 11 (ver figura 6). A modo de ejemplo, en la dimensión activo/reflexivo, si el resultado del cuestionario toma un valor entre 0 a 3 (en cualquiera de las dos direcciones), el estudiante se encuentra bien balanceado entre las dos dimensiones que figuran en los extremos de la escala; si el puntaje en la escala es de 5 a 7, tiene una moderada preferencia por el extremo al que se acerca y si el puntaje en la escala es de 9 a 11, tiene una fuerte preferencia por el extremo donde se encuentra.

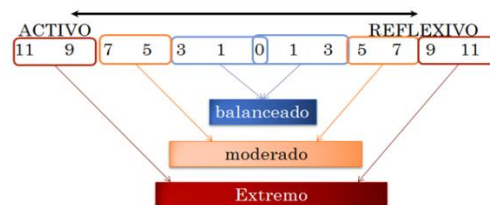


Fig. 6. Ejemplo de la síntesis de los estilos de aprendizaje de Felder para la dimensión Procesamiento de la Información

El razonamiento basado en estereotipos trabaja con una primera caracterización sobre el estudiante y es usado para modelar al estudiante inicialmente [14]. También tenemos la definición de [15]: “un estereotipo representa una colección de atributos de personas .que permite al sistema realizar un gran número de inferencias plausibles basadas en un número pequeño de observaciones”. Aquí se ha utilizado una clasificación de estereotipos de estudiantes como grupos que comparten características comunes en el aprendizaje. Se han definido los siguientes estereotipos clásicos para medir el nivel de conocimientos: novicio, medio, experto.

2.3 Proceso de Inferencia

El proceso para predecir un perfil cognitivo de cada estudiante consta de los siguientes módulos (ver figura 7). Se parte del modelo de perturbación y de los estilos de aprendizaje de Felder. Estos dos módulos constituyen las entradas al sistema de inferencia difusa. El proceso de inferencia implica definir las variables lingüísticas de entrada, de salida y reglas de inferencia difusa. Como salidas del sistema se obtiene un perfil cognitivo por estudiante.

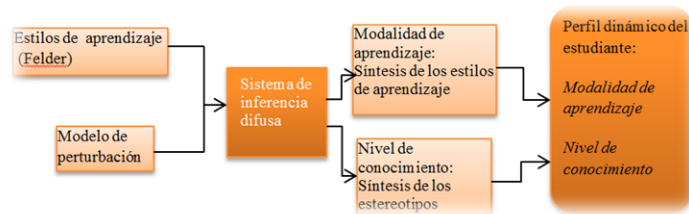


Fig. 7. Inferencia difusa en el modelo del estudiante

Para los estilos de aprendizaje se han utilizado las dos dimensiones de Felder/Silverman descritas previamente: **Procesamiento de la información** y **Entendimiento**. Cada una de ellas serán las variables lingüísticas de entrada y poseen tres términos lingüísticos cada una (siendo estos conjuntos difusos). Procesamiento de la información posee como términos lingüísticos *activo*, *balanceado* y *reflexivo*. Entendimiento posee como términos lingüísticos *secuencial*, *balanceado* y *global*. Las reglas del sistema de inferencia realizan una síntesis de las dos dimensiones de Felder. A partir de esa síntesis se define una de las variables lingüísticas de salida **Modalidad de aprendizaje**. Esta posee como términos lingüísticos *extremo*, *moderado* y *balanceado*. Se han definido 9 reglas difusas para la inferencia. En la figura 8 se aprecia la inferencia del sistema realizado con un FIS. Se ha utilizado el sistema FISPRO (<http://www7.inra.fr/mia/M/fispro/>).

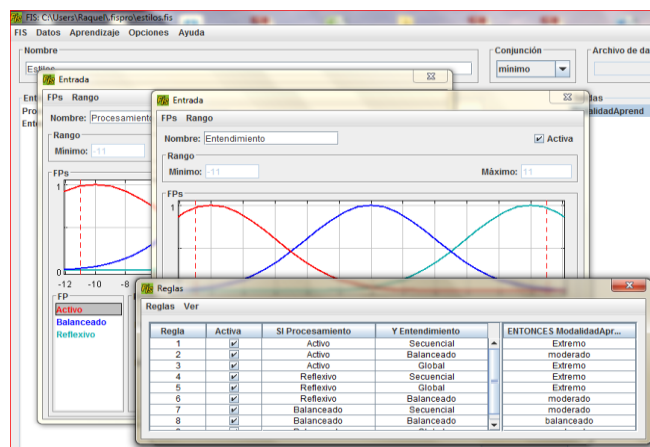


Fig. 8. Sistema de inferencia para la Modalidad del aprendizaje. En la pantalla delantera se aprecian las reglas de inferencia y detrás dos pantallas con las variables de entrada y sus tres términos lingüísticos definidos como conjuntos difusos

El modelo de perturbación proporciona la entrada al sistema de inferencia para obtener el nivel de conocimiento. Esta entrada usa la 4-upla asignada a cada nodo del árbol que representa el modelo de perturbación: (no-aprendido, insatisfactoriamente conocido, conocido, aprendido). Las componentes de la 4-upla constituyen las cuatro variables lingüísticas de entrada: **no-aprendido** (con los

términos lingüísticos *bajo, medio, alto*), **insatisfactorio** (*bajo, medio, alto*), **conocido** (*bajo, medio, alto*) y **aprendido** (*bajo, medio, alto*) y una variable de salida con los estereotipos clásicos en el **nivel de conocimiento** (*novicio, medio, experto*). En la figura 9 se exponen parte de las reglas difusas para inferir el nivel de conocimiento.

Finalmente el perfil cognitivo dinámico queda formado por los resultados del par: (modalidad de aprendizaje, nivel de conocimiento) obtenidos a partir de los dos modelos de inferencia difusa explicitados previamente. A modo de ejemplo, un perfil es (moderado, intermedio) en un momento del aprendizaje. El diagnóstico cognitivo se construye con la evolución en el tiempo de estos perfiles.

Regla	Peso	Activa	SI NoAprend...	Y Insatisfact...	Y Conocido	Y Aprendido	ENTONCES NivelConocimiento
1	2	<input checked="" type="checkbox"/>	alto	alto	medio	bajo	novicio
2	1	<input checked="" type="checkbox"/>	medio	medio	medio	medio	medio
3	1	<input checked="" type="checkbox"/>	alto	medio	bajo	bajo	novicio
4	1	<input checked="" type="checkbox"/>	alto	medio	medio	bajo	medio
5	1	<input checked="" type="checkbox"/>	bajo	bajo	medio	alto	experto
6	1	<input checked="" type="checkbox"/>	medio	bajo	alto	medio	medio
7	1	<input checked="" type="checkbox"/>	bajo	medio	alto	medio	medio
8	1	<input checked="" type="checkbox"/>	bajo	bajo	alto	alto	experto
9	1	<input checked="" type="checkbox"/>	alto	alto	bajo	bajo	novicio

Fig. 9. Algunas reglas de inferencia difusa del sistema FIS para el Nivel de Conocimiento

A partir de los casos individuales se analizaron los perfiles más frecuentes. Estos resultados ayudaron a corregir la enseñanza de algunos tópicos (p.e. algoritmia básica).

2.4 Resultados

A fin de ilustrar el modelo propuesto, se analizaron los resultados de la actividad de 30 estudiantes en una asignatura del ciclo básico de la carrera de Ingeniería Informática de la Universidad Nacional de Mar del Plata. Basado en el dominio que presentado en la figura 1, se evaluaron los estudiantes con dos pruebas distintas. Con los resultados de la evaluación se construyó una tabla ilustrada en la tabla 1, donde se analizó la evaluación del nodo principal sobre Lenguaje assembler.

Tabla 1. Modelo de perturbación para el tópico “Lenguaje assembler” para tres estudiantes.

Nivel de conocimiento / Estudiantes	NoAprendido	Insatisf. conocido	conocido	aprendido	Resultado de la inferencia para el nivel de conocimiento
Ariel G.	0.46 (medio)	0.05 (bajo)	0.19 (bajo)	0.30 (bajo)	novicio
Erik B.	0 (bajo)	0 (bajo)	0.04 (bajo)	0.96 (alto)	experto
Lucia C.	0.08 (bajo)	0.18 (bajo)	0.4 (medio)	0.34 (medio)	medio

En la mencionada tabla solo se muestran los primeros tres estudiantes. Se aprecia las funciones de pertenencia predominantes, entre paréntesis, correspondientes al valor dado por el modelo de perturbación y , en la última columna, los valores del *nivel conocimiento*, como estereotipos típicos, inferidos con las reglas difusas. En la figura 4 se ilustra el modelo de perturbación de un estudiante cuya evaluación está expresada en la fila 1 (correspondiente al estudiante Ariel G.).

Para procesar los datos sobre los estilos de aprendizaje en el sistema FisPro los valores de las dimensiones $[11,10,\dots,0, 1,\dots,10,11]$ se transformaron en $[-11,\dots,11]$. En la tabla 2 se ilustra el procesamiento de los estilos de aprendizaje de los mismos estudiantes de la tabla 1. Se muestran las funciones de pertenencia predominantes (entre paréntesis) correspondientes al valor dados por los estilos de aprendizaje y , en la última columna, los valores de la *modalidad del aprendizaje* inferidos con las reglas difusas.

Tabla 2. Estilos de aprendizaje de Felder (procesamiento y entendimiento)

Estilos de aprendizaje Estudiantes	Procesamiento	Entendimiento	Resultado de la inferencia de la <i>modalidad del aprendizaje</i>
Ariel G.	-5 (activo)	2 (balanceado)	moderado
Erik B.	1 (balanceado)	-7 (secuencial)	moderado
Lucia C.	-6 (activo)	7 (global)	extremo

A partir de las tablas 1 y 2 completas con los datos de los 30 estudiantes del grupo analizado, se computó los perfiles de mayor frecuencia. Los resultados de este análisis se muestran en la tabla 3.

Tabla 3. Los perfiles cognitivos más frecuentes a mitad del dictado del curso

Perfiles	Modalidad de aprendizaje	Nivel de conocimiento	Porcentaje
1	moderado	medio	33 %
2	balanceado	medio	17 %
3	balanceado	novicio	12 %
4	moderado	experto	10 %
5	extremo	medio	7 %
otros			21 %

Sobre el total del grupo se pudo discriminar los alumnos sin ningún conocimiento previo (72 %) a partir del análisis de la primera componente de la 4-upla. Los alumnos que volvieron a cursar (15 %) fueron detectados a partir del dominio insatisfactorio del tópico principal. Estos alumnos que perdieron la asignatura en exámenes finales (10%) se encontraron considerando la tercera componente y , finalmente, hubo un 3% de estudiantes con pases de otras carreras de Informática quienes sobresalieron en la cuarta componente.

Conclusión

El sistema de diagnóstico cognitivo presentado aporta información adicional sobre la evaluación del estudiante cuando se lo compara con la metodología tradicional de

estimación del logro. Se emplean datos sobre los estilos de aprendizaje y del progreso de cada estudiante. El modelo de perturbación proporciona detalles sobre la evolución del aprendizaje en cada nodo y puede ser visualizado a lo largo de un tiempo determinado y en nodos con distintos niveles de granularidad. Este modelo provee información cuantitativa y cualitativa. El modelo de personalización contribuye con información sobre la modalidad del aprendizaje. La dupla (modalidad de aprendizaje, nivel de conocimiento) conforma un perfil cognitivo.

Estos perfiles mejoran la información aportada por las tradicionales notas de una forma más realista. El modelo expuesto en este artículo será usado en el desarrollo de un ambiente de diagnóstico cognitivo en temas de las ciencias básicas de ingeniería con el objetivo fundamental de mejorar la adaptación de las pruebas de evaluación.

Referencias

1. Chrysafiadi K.; Virvou M.: Student modeling approaches: A literature review for the last decade. *Expert Systems with Applications*. 40 (11), 4715--4729. (2013).
2. Chrysafiadi K.; Virvou M.: *Advances in Personalized Web-Based Education*. Springer Cham Heidelberg (2014).
3. Self, J.A.: Student models in computer-aided instruction, *International Journal of Man-Machine Studies* 6(2), 261--276. (1974).
4. VanLehn K. "Student modeling". En *Foundations of Intelligent Tutoring Systems*, eds.: Polson M.C. y Richardson J.J. Lawrence Erlbaum. (1988)
5. Zadeh L. *Computing with Words. Principal Concepts and Ideas*. Springer. (2012).
6. Zadeh L. The concept of a linguistic variable and its application to approximate reasoning - I, *Information Sciences*, vol. 8, no. 3, pp. 199-249, (1975).
7. Huapaya C. Lizarralde F., Gonzalez M., Guccione L., Arona G. y Vivas J. "Modelo del Estudiante basado en la Lógica Difusa", WICC 2014 XVI Workshop de Investigadores en Ciencias de la Computación, (2014).
8. Stansfield, J. C., Carr, B., y Goldstein, I. P.. "Wumpus advisor I: A first implementation of a program that tutors logical and probabilistic reasoning skills". At Lab Memo 381. Cambridge, Massachusetts: Massachusetts Institute of Technology. (1976).
9. Martins, A. C., Faria, L., Vaz de Carvalho, C., y Carrapatoso, E. User modeling in adaptive hypermedia educational systems. *Educational Technology & Society*, 11(1), 194--207. (2008).
10. Felder, R. M. "Are Learning Styles Invalid? (Hint: No!)" On-Course Newsletter, September 27, recuperado de : <http://www.oncourseworkshop.com/Learning046.htm>. (2010)
11. Felder R.M., Spurlin J.E., "Applications, Reliability, and Validity of the Index of Learning Styles." *Intl. Journal of Engineering Education*, 21(1), 103--112. (2005).
12. Felder, R.M., & Silverman, L.K. Learning and teaching styles in engineering education [Electronic Version]. *Engr. Education*, 78(7), 674-681. Recuperado el 24 de Julio del 2009 de: <http://www4.ncsu.edu/unity/lockers/users/f/felder/public/Papers/LS-1988.pdf>. (1988)
13. Felder R. y Soloman, B.A., Index of learning styles questionnaire. Recuperado el 10 de marzo de 2016 de: <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>
14. Kay J. "Stereotypes, students models and scrutability". *Proceedings of the 5th International Conference on Intelligent Tutoring Systems* 19-30. (2000)
15. Rich E. "User Modeling via Stereotypes". *Cognitive Science* 3, 329-354. (1979).

Incorporación de herramientas de trabajo colaborativo en el entorno AulasWeb: una propuesta de modificación del selector de actividades

Alejandro Héctor González¹, Cristina Madoz¹, Leandro Matías Romanut², Rodolfo Bertone¹

¹Instituto de Investigación en Informática III – LIDI – Facultad de Informática UNLP

²Dirección de Educación a Distancia y Tecnologías UNLP

{agonzalez,cmadoz,pbertone}@lidi.info.unlp.edu.ar

leandro.romanut@presi.unlp.edu.ar

Abstract. Este artículo está basado en el trabajo de investigación de una tesina de grado de la Facultad de Informática de la UNLP. Durante el proceso de desarrollo se realiza una investigación de los denominados LMS (Learning Management System - Sistema de Gestión de Aprendizaje), como Moodle (AulasWeb y Qoodle), Sakai y WebUNLP; se evalúan distintos aspectos y en particular, se analizan las herramientas de trabajo colaborativo que poseen los LMS estudiados. A partir de este estudio y de una encuesta realizada a profesionales en el uso de los entornos se desarrollan instructivos que, acompañados de un personaje virtual, orientan a los docentes en la inclusión de actividades de trabajo colaborativo. Así también se muestran los primeros resultados logrados a través de la realización de una experiencia piloto en un curso sobre el entorno AulasWeb, basado en Moodle.

Keywords: entornos virtuales de enseñanza y aprendizaje, trabajo colaborativo, personajes virtuales, b-learning

1 Introducción

A partir de la experiencia, en varias ediciones, de la capacitación de profesores que se lleva adelante en la Dirección de Educación a Distancia y Tecnologías de la UNLP, se observa que los docentes poseen poca experiencia en el uso de entornos virtuales, y al momento de la creación y diseño de las aulas, se encuentran con variados recursos y actividades para utilizar sin tener en claro el objetivo pedagógico con el cual incorporarlos.

La propuesta del trabajo que se presenta tuvo como objetivo estudiar los LMS: Moodle (AulasWeb y Qoodle), Sakai y WebUNLP, utilizados en distintas universidades, analizando diversos factores que están relacionados a cómo surgieron, cómo están organizados a nivel sistema, qué roles de usuario poseen, qué herramientas proveen para realizar la creación y diseño de las aulas, entre otras características. En particular, se profundiza el estudio de las herramientas colaborativas provistas por estos LMS a través de un cuadro comparativo donde se destacan las características de cada una. A su vez, se propuso generar un conjunto de

instructivos que incluyan posibles situaciones pedagógicas para utilizar las actividades de trabajo colaborativo. Estos instructivos se crearon basados en encuestas realizadas a profesionales en el uso y coordinación de aulas virtuales y en un análisis de uso de herramientas de trabajo colaborativo en las aulas virtuales del entorno AulsWeb. Los instructivos están acompañados de un personaje virtual que colabora como guía en la creación y diseño de las aulas virtuales.

2 Marco teórico

La incorporación de las TIC en la educación, produce una serie de cambios y transformaciones en las formas en que se representan y llevan a cabo los procesos de enseñanza y aprendizaje.

Existen diversas herramientas TIC con las cuales se pueden diseñar y llevar a cabo las prácticas educativas dentro del aula virtual.

Desde una visión constructivista, explica Barriga, “el aula”, lejos de referirse a un conjunto de recursos físicos, consiste en un sistema interactivo en el cual ocurren una serie de transacciones comunicativas. Este sistema genera un ambiente particular de trabajo para la construcción del aprendizaje, determinado por una serie de reglas de organización y participación, este ambiente se denomina “ambiente de aprendizaje”. [1]

Por su parte, Bates explica que la modalidad educativa puede cambiar de acuerdo a la cantidad de e-learning que interviene en el proceso. En la modalidad presencial, los estudiantes comparten un mismo espacio físico y tiempo, la formación es sincrónica y en el mismo lugar y es la más evolucionada dada la historia misma de la educación. [2]

2.1 Modalidades educativas

Las modalidades educativas están relacionadas con el ambiente en el que se desarrollan los procesos de enseñanza y aprendizaje. Definen la utilización de los medios, de los recursos educativos y establece las acciones de los agentes del proceso -estudiantes, tutores, coordinadores-. [3]

A medida que incrementa el uso de e-learning en las modalidades de enseñanza aparecen dos términos: el b-learning y el aprendizaje distribuido.

González define al b-learning (Aprendizaje híbrido o combinado) como la modalidad que combina la enseñanza presencial con la no presencial de tal manera que ambas experiencias de aprendizaje se vuelven imprescindibles para completar con éxito los objetivos de aprendizaje. [4][5]

Al involucrar de manera total el e-learning, se hace referencia a la educación a distancia, dado que involucra diferentes formas de comunicación.

Algunos autores definen a la educación a distancia como:

“La enseñanza a distancia es un sistema tecnológico de comunicación bidireccional (multidireccional), que puede ser masivo, basado en la acción sistemática y conjunta de recursos didácticos y el apoyo de una organización y tutoría, que, separados

físicamente de los estudiantes, propician en éstos un aprendizaje independiente (cooperativo).” [6]

Por su lado, Mena, Diez y Rodríguez señalan a la educación a distancia como una modalidad educativa que, mediatizando la mayor parte del tiempo la relación pedagógica entre quienes enseñan y quienes aprenden a través de distintos medios y estrategias, permite establecer una particular forma de presencia institucional más allá de su tradicional cobertura geográfica y poblacional, ayudando a superar problemas de tiempo y espacio.[7][8]

De las definiciones anteriores puede pensarse que la educación a distancia es un proceso educativo que mantiene una relación pedagógica entre los docentes y los estudiantes a través de distintos recursos y estrategias, siendo este proceso mediado por tecnologías. Los entornos de aprendizaje y enseñanza apoyan la comunicación y el intercambio.

3 Entornos Virtuales de Enseñanza y Aprendizaje

Un LMS (Learning Management System - Sistema de Gestión de Aprendizaje) es un software online que permite administrar, distribuir, monitorear, evaluar y apoyar las diferentes actividades previamente diseñadas y programadas dentro de un proceso de formación completamente virtual (e-learning) o de formación semi-presencial (b-learning), explica Cañellas Mayor. [9]

3.1 Análisis de los Entornos Virtuales de Enseñanza y Aprendizaje

Algunos de los entornos de enseñanza y aprendizaje que se utilizan para abordar el estudio de las actividades de trabajo colaborativo son desarrollos de comunidades de software libre, como Moodle¹ y Sakai², y otros son espacios que están a cargo de alguna entidad educativa: Qoodle³ (Universidad Nacional de Quilmes), AulasWeb⁴ (Presidencia - Universidad Nacional de La Plata) y WebUNLP⁵ (Facultad de Informática - Universidad Nacional de La Plata).

Para cada uno de los entornos elegidos se analizó: cómo surgieron, cómo están organizados a nivel sistema, qué roles de usuario poseen, qué herramientas proveen para realizar la creación y diseño de las aulas, si poseen herramientas externas además de las que vienen de manera nativa.

Luego se realizó un análisis de las herramientas de trabajo colaborativo que provee cada entorno. Se puede observar en la siguiente figura el cuadro comparativo de las herramientas que ofrece cada entorno virtual.

¹<https://moodle.org/>

²<https://sakaiproject.org/>

³<https://qoodle.uvq.edu.ar>

⁴<https://aulasweb.ead.unlp.edu.ar/>

⁵<https://webunlp.ead.unlp.edu.ar/>

Actividades/Entornos	Moodle (AulasWeb/ Qoodle)	WebUNLP	Sakai
Base de Datos	✓	✗	✗
Chat	✓	✗	✓
Foro	✓	✓	✓
Glosario	✓	✓	✓
Taller	✓	✗	✗
Tarea	✓	✓	✓
Wiki	✓	✗	✓

Fig. 1. Cuadro comparativo de las herramientas de trabajo colaborativo en los entornos virtuales

Algunos ejemplos: la actividad Base de Datos es propia de Moodle, pero no está presente en Sakai, ni en WebUNLP. El foro es una actividad provista por todos los entornos pero con distintas características. En Moodle existen distintos tipos: Uso general, Preguntas y respuestas, Cada persona plantea un tema, Formato de blog y Debate sencillo. En Sakai no hay tipos, los aportes pueden ser moderados antes de que sean visibles, se necesita un tema para inaugurar el foro, de otra manera, no será visualizado. En WebUNLP el recurso se llama “Foro de discusión” y posee varios tipos Público, Privado, Bloqueado, Habilitado, Anónimo, No anónimo. La actividad Wiki está disponible en Moodle y en Sakai, en el primero permite crear Wikis de manera colaborativa o individual y soporta varios formatos (HTML, Creole, NWiki) y en el segundo no posee ninguna característica destacable, por su parte, no está como posibilidad en WebUNLP.

4 Justificación y contexto de desarrollo

La tesina se enmarca dentro el proyecto de investigación: “*Tecnologías para sistemas de software distribuidos. Calidad en sistemas y procesos. Escenarios educativos mediados por TIC*”, del Instituto de Investigación en Informática III-LIDI de la Universidad de la Plata. Se tiene en cuenta para el desarrollo la experiencia en capacitación de profesores llevadas a cabo en la Dirección de Educación a Distancia y Tecnologías de la UNLP. Se observa desde hace tiempo la falta de inclusión de actividades orientadas al trabajo colaborativo en las propuestas de los cursos a través de las herramientas provistas por el entorno AulasWeb (perteneciente a la dirección y basado en Moodle).

Para poder profundizar sobre las herramientas de trabajo colaborativo en los entornos virtuales de enseñanza y aprendizaje, se llevó a cabo una encuesta de carácter anónimo compuesta por preguntas abiertas para poder brindar mayor libertad de expresión. Estas encuestas se realizaron a los docentes y coordinadores de aulas virtuales de la Universidad Nacional de La Plata, con el fin de dar a conocer su experiencia con entornos desde una perspectiva no solo de usabilidad, sino de su experiencia como docentes (interactuando con alumnos y materiales) y acompañando diferentes propuestas educativas de distintos niveles (pregrado, grado, postgrado, extensión). De las encuestas realizadas (54 en total) se observa que no todos utilizan

herramientas de trabajo colaborativo y, generalmente, las más utilizadas son los foros y las tareas.

Las encuestas realizadas a los docentes colaboraron para el diseño y desarrollo de breves instructivos que identifiquen algunas situaciones pedagógicas y didácticas donde se puedan utilizar las actividades de trabajo colaborativo.

Para poder revisar las respuestas de los encuestados se realizó un análisis de la presencia de herramientas colaborativas en los cursos del entorno AulasWeb. De las encuestas realizadas a docentes y coordinadores de aulas y el análisis del uso de las actividades colaborativas en los cursos, se puede suponer que a los profesores se le presentan problemas al momento de utilizar las herramientas de trabajo colaborativo:

- Algunas de las actividades de trabajo colaborativo poseen cierta complejidad en su configuración lo que provoca que los docentes se abstengan de usarlas.
- Los docentes habitualmente no intentan dinamizar las actividades del curso incluyendo las actividades colaborativas. Generalmente trabajan con texto (recursos de tipo archivos en AulasWeb) convirtiendo al aula en un sitio “estático”, donde los alumnos se descargan el documento y trabajan con la actividad. Esto pasa sin que haya un real motivo, tanto en la modalidad de extensión del aula como, en las propuestas a distancia.
- Los docentes están dispuestos a trabajar con las actividades colaborativas, pero no rediseñan ni reformulan las prácticas educativas, no adecuándolas para usar tales actividades.

5 Herramienta desarrollada

Con el fin de tener una herramienta flexible y que pueda ser incorporada fácilmente en cualquier entorno Moodle, se optó por realizar una modificación al código que permita incorporar un personaje virtual que colabore en la selección de actividades y recursos de Moodle.

Se tomó como base el selector de actividades de Moodle. En la interfaz de este selector, el docente puede ver el listado de todas las actividades y recursos que posee Moodle para agregar a las aulas. Cuando selecciona una de las actividades o recursos, se muestra sobre la misma interfaz una descripción de la herramienta. A partir de este selector, se extienden tres funcionalidades:

- Organizar el selector de actividades en tres secciones:
 - **Actividades de Trabajo Colaborativo** (Base de Datos, Chat, Foro, Glosario, Taller, Tarea, Wiki, y otras herramientas externas como VPL y BigBlueButton).
 - **Otras Actividades** (Consulta, Cuestionario, Herramienta Externa, Lección, Paquete SCORM).
 - **Recursos** (Archivo, Carpeta, Etiqueta, Libro, Página, URL).
- Ofrecer para cada actividad de trabajo colaborativo una ayuda pedagógica sobre en qué situaciones educativas podrían emplear dicho recurso.

- Ofrecer un asistente que permite reproducir la ayuda a través de un audio y ver un ejemplo de uso de la actividad seleccionada.

2.1 Implementación

Para poder desarrollar esta nueva interfaz del selector de actividades se trabajó con el archivo *renderer.php*, ubicado en el directorio de Moodle, dentro de la carpeta *course*. El archivo está ubicado dentro de ese espacio ya que el Selector de Actividades es una herramienta utilizada en el contexto de curso dentro de todo el entorno.

La clase `core_course_renderer`

La clase `core_course_renderer` permite, entre otras acciones, generar y visualizar el selector de actividades a la hora de querer agregar una nueva actividad o recurso dentro del aula virtual.

Métodos modificados

Los métodos que han sido modificados y que participan en la construcción del selector de actividades son:

Public function `course_modchooser` (`$modules`, `$course`)

Este método construye el HTML para el selector de actividades. En esta función se agrega, a través de un etiqueta HTML `<p>` almacenada en la variable `$textogeneral`, el texto que explica para qué sirven los instructivos. Luego se lo inserta con una etiqueta HTML `<div>` por medio de la clase y la función `html_write::tag()`. Además se realiza la categorización de las actividades para esto se crean dos arreglos: `$activitiescolaborative` y `$otheractivities`, se incluyen los módulos de las herramientas al llamar al método `array_filter()` filtrando por nombre de la actividad. Estos arreglos se agregan a la variable PHP `$formcontent`.

Protected function `course_modchooser_module` (`$module`, `$classes = array('option')`)

Este método devuelve el HTML para la actividad/recurso recibido como parámetro con las clases necesarias. En esta función se declaran las variables globales `$CFG` y `$PAGE` que poseen parámetros de la configuración personal del sitio y así poder incluir el archivo JavaScript `dialogo.js` para que se active el funcionamiento del globo de conversación del personaje virtual. Así también en este lugar se agrega para cada actividad de trabajo colaborativo: el instructivo mediante las etiquetas HTML `<div><p>`, el asistente con la etiqueta ``, el reproductor de audio por medio de la etiqueta de HTML 5 `<audio>`, el ejemplo de la actividad a través de un link `<a>` y se captura el evento `onclick` para llamar a la función JavaScript que muestra el globo de conversación. Todo eso se almacena en la variable PHP `$speech` y se la adhiere a la variable `$output` mediante la clase y la función `html_writer::tag()`.

Protected function course_modchooser_title (\$title, \$identifier = null)

Este método setea los títulos de las categorías (Actividades de Trabajo Colaborativo, Otras actividades, Recursos) en las cuales se dividen las herramientas. Para lograr esto se modifica el atributo *title* del arreglo *\$module*.

Personaje virtual

Se creó un personaje virtual a través del sitio “Pocoyó” mediante su aplicación Pocoyize⁶, la cual permite crear avatares de caricaturas y descargarlos de manera gratuita.

La incorporación de un personaje intenta generar una estrategia que oriente a los docentes en la comprensión de las actividades de trabajo colaborativo.

En esta oportunidad se decide trabajar con un personaje que cumple el rol de tutor tecnológico, que es aquel que realiza el acompañamiento durante el aprendizaje de los alumnos y colabora en el entendimiento y uso de las tecnologías involucradas en el aula virtual. Para la creación final del personaje se trabaja con los tres aspectos mencionados por Rib Davis: las marcas de nacimiento, el aprendizaje y el personaje ahora. [10][11]



Fig. 2. Manu, el tutor tecnológico

Funciones JavaScript creadas

Se crearon un grupo de funciones para que al hacer click sobre el asistente Manu se muestre un globo de conversación con las funcionalidades de escuchar el instructivo y ver el ejemplo. Estas funciones se encuentran en el archivo *dialogo.js* ubicado en la carpeta *course* de Moodle.

Clases e Identificadores del estilo CSS creado

Los instructivos y el personaje poseen un estilo CSS propio. Se crearon clases e identificadores CSS para darle el formato adecuado a cada ayuda y al asistente. El archivo que contiene el estilo del desarrollo se llama *dialogo.css* y se ubica en la carpeta *style* del tema que esté usando el entorno Moodle, de la siguiente manera: *theme/TEMA_ACTUAL/style*.

El formato CSS creado para la herramienta se divide en tres grandes grupos:

⁶<http://www.pocoyo.com/pocoyizador>

- El estilo para generar un **Efecto UP** que sucede al pasar el mouse sobre el personaje virtual y que cambia la figura del cursor. Las clases involucradas en este estilo son: *imagen*, *ex1*, *puntero* y *reproductor*.
- El estilo para el texto de los instructivos. Los identificadores utilizados para el estilo de los textos son: *texto-asistente*, *justificar*, *centrar* y *texto*.
- El estilo para la burbuja de diálogo. La clase creada para la burbuja de diálogo es: *rectangle-speech-border*.

Para que la plantilla reconozca el archivo .css hay que modificar el archivo `config.php` ubicado en la carpeta `theme/TEMA_ACTUAL` que posee cada tema instalado en Moodle. El entorno AulasWeb utiliza el tema Essential.

6 Primera experiencia

Se instaló y utilizó la modificación desarrollada al selector de actividades; en el servidor donde reside el entorno AulasWeb (basado en Moodle) de la Dirección de Educación a Distancia y Tecnologías de la UNLP.

Se realizó una prueba piloto del asistente en el curso “Introducción al uso de los entornos virtuales de la UNLP soportados en Moodle (Aulas Web – Cavila – Cursos externos)”. El objetivo del curso busca que los profesores se apropien de conceptos básicos para poder implementar sus cursos y se acerquen a la edición básica de las herramientas de las aulas virtuales del entorno Moodle. El curso se dictó bajo la modalidad b-learning con la duración de 5 semanas con 2 encuentros presenciales, en los cuales se trabaja en forma de taller utilizando las herramientas de Moodle, en un lapso de dos horas y media cada uno. El curso se dictó durante el 2015. Asistieron 18 docentes de las distintas unidades académicas de la UNLP.

El curso propone realizar la creación y el diseño de un aula virtual tomando como referencia la propuesta educativa de alguna cátedra a la que pertenezcan los participantes. Los estudiantes tuvieron que leer la documentación y las guías de ayuda desarrolladas para las actividades de trabajo colaborativo, así como también utilizar el asistente para ver algunos ejemplos de uso de las herramientas.

Se realizó una encuesta, a través de un formulario online creado con la herramienta Google Drive, con el objetivo final de indagar el nivel de aceptación del prototipo en el entorno AulasWeb y extraer, algunas sugerencias y mejoras que puedan realizarse sobre el desarrollo realizado para este trabajo. Se pueden mencionar dos de las preguntas realizadas:

La primera pregunta *¿Utilizó los instructivos disponibles como ayuda para diseñar las prácticas educativas que involucren las actividades de Trabajo Colaborativo?*, Todos los encuestados respondieron que sí. Esto indica que recurrieron a la ayuda que ofrece la herramienta para planificar y comprender el uso desde una perspectiva más pedagógica y no sólo tecnológica.

La tercera pregunta *¿Utilizó el asistente de las actividades de Trabajo Colaborativo?*, indagaba sobre el uso de *Manu*, el asistente virtual. El 71% de los encuestados respondió que sí mientras que el 29% respondió que no, la justificación para este último porcentaje fue que no fue necesario y bastó con leer el instructivo.

La segunda parte de la tercera pregunta *¿Cuál es su opinión acerca de tener la opción de escuchar la ayuda?*, es una pregunta abierta que consultaba la opinión de que el asistente provea la posibilidad de escuchar la ayuda, las respuestas fueron diversas:

- El 43% dijo: *“Muy práctica”*, lo cual implica que es una alternativa para ciertas ocasiones donde se dificulte la lectura del instructivo.
- El 15% dijo: *“Excelente”*.
- Un 14% dijo: *“Interesante”*.
- Otro 14% dijo: *“Lo hace más real”*, lo cual es importante destacar ya que la figura del personaje es una persona que “simula” explicar el instructivo a través del audio.

7 Conclusiones

Se realizó el estudio de diferentes entornos virtuales de enseñanza y aprendizaje. Se estudiaron los distintos LMS usados tanto en la UNLP como en otras universidades, como es el caso de Sakai y Qoodle. Se realizó un estudio comparativo de las herramientas de trabajo colaborativo que poseen estos entornos virtuales delineando las potencialidades y falencias de cada una, lo cual permite decidir en determinadas situaciones y de acuerdo a las necesidades existentes, que herramienta utilizar, así como también sobre qué entorno virtual construir la propuesta educativa.

Se desarrolló una herramienta que colabora con la creación, diseño, uso e inclusión de las actividades colaborativas en las propuestas educativas que los docentes llevan adelante con sus cursos.

La experiencia piloto con la herramienta permite ver buenos resultados, tanto en la comprensión de las actividades de trabajo colaborativo desde una perspectiva pedagógica, así como también desde un punto de vista tecnológico.

Los ejemplos que provee el asistente son analizados como útiles y claros, esto aproxima a la suposición que se tenía sobre el tipo de información que ofrece el manual de ayuda de Moodle. Permiten comprender e interpretar el texto de ayuda y utilizar los ejemplos para el diseño de las propuestas educativas.

La inclusión de un asistente virtual involucrando la creación de un guión para el mismo, logra recorrer un proceso creativo que permite incorporar diversos elementos ligados a la multimedia que van desde la presentación del personaje, su historia y su forma de ser hasta los procedimientos para la asistencia en el uso de las actividades colaborativas.

Se realizó un aporte tecnológico y pedagógico a la Dirección de Educación a Distancia y Tecnologías de la UNLP, incorporando el asistente en los LMS que se utilizan actualmente: AulasWeb, CAVILA - UNLP y Cursos Externos, todos basados en Moodle.

8 Trabajo futuro

Se proponen las siguientes mejoras a futuro para aplicar sobre la herramienta desarrollada:

- Extender el correspondiente instructivo junto con el asistente a cada actividad y recurso disponible dentro del entorno AulasWeb.
- Agregar nuevas funciones para personalizar el asistente en aspectos de imagen de acuerdo a como lo prefiera el docente del curso: cambiar la ropa, el color de piel, el color y corte de pelo, los ojos, la boca, el cuerpo, la voz.
- Sumar nuevas funcionalidades al asistente Manu:
 - Proveer un espacio de preguntas frecuentes con respuestas automatizadas.
 - Sugerir un listado de herramientas externas que se puedan incluir (embeber) dentro del entorno AulasWeb. Este listado constaría de una breve descripción, un tutorial de uso y el enlace a la página web de la aplicación. Por ejemplo: construir presentaciones online (Prezi), diseñar videos interactivos (Moovly), confeccionar collages (Fotor - BeFunky), organizar líneas de tiempo (Capzles), elaborar infografías (RAW), armar nubes de etiquetas (Tagxedo - Word itOut), desarrollar murales colaborativos (Padlet), construir imágenes interactivas (Thinglink), etc.

9 Bibliografía

1. Barriga, F. D. Educación y nuevas tecnologías de la información y la comunicación: ¿Hacia un paradigma educativo innovador? (2008)
2. Bates, A. T. Technology, e-learning and distance education. Routledge. (2005)
3. Jiménez, D. Nuevos paradigmas educativos y modalidades educativas. Recuperado de: http://es.slideshare.net/auri_desi/modalidades-educativas-5551315 (2010)
4. González, A. H. Educación a Distancia y Tecnologías Digitales en la Enseñanza Universitaria. Recuperado de https://prezi.com/v-vkyghu8zsc/2015-especializacion-en-docencia-universitaria-unlp/?utm_campaign=share&utm_medium=copy (2015)
5. González, C. Cuadro comparativo: Blended learning, Distributed Learning, Online Learning. Recuperado de <http://es.calameo.com/read/001790350da562655eae> (2012)
6. García Aretio, L. La Educación a Distancia. De la teoría a la práctica. Ariel editorial S. A. (2001)
7. Moore, M. & Kearsley, G. Distance education: A systems view. Boston, MA: Wadsworth Publishing. (1996)
8. Mena, M., Diez, L. M., y Rodríguez, L. El Diseño de proyectos de Educación a distancia. Colección Itinerarios. Ediciones La Crujía y Editorial Stella. Buenos Aires. (2005)
9. Cañellas Mayor, A. LMS y LCMS: funcionalidades y beneficios. Recuperado de <http://www.centrocp.com/lms-y-lcms-funcionalidades-y-beneficios/> (2014)
10. Rib, D. Escribir guiones: desarrollo de personajes. Editorial Paidós, manuales de escritura. Barcelona. España. (2004)
11. González, A. H. TICs en el proceso de articulación entre la Escuela Media y la Universidad. Tesis de Maestría, Facultad de Informática, UNLP. (2008)

Evaluación y medición de la calidad percibida de MOOC

Marisa Cecilia Tumino y Juan Manuel Bournissen
Universidad Adventista del Plata, Libertador San Martín, Entre Ríos, Argentina
{marisatumino, juanbournissen}@doc.uap.edu.ar

Resumen. Este trabajo tiene como objetivo analizar comparativamente las principales plataformas MOOC disponibles bajo licencias abiertas, contribuyendo a la toma de decisiones de quienes pretenden desarrollar sus propios cursos masivos online, acerca de la plataforma a usar. El trabajo de investigación incluye (a) un análisis comparativo de las principales plataformas MOOC, (b) la selección de las dos plataformas que ofrecen más prestaciones y (c) la adaptación y validación de un instrumento de medición de la percepción de calidad de los usuarios. A partir del estudio se arribará a las conclusiones que permitan brindar información pertinente a quienes manifiesten interés en la construcción de MOOC. En una segunda etapa del proyecto, se construirá, implementará y evaluará un módulo de curso MOOC bajo las dos plataformas seleccionadas.

Palabras clave: MOOC, Tecnologías educativas; medición de la calidad

1. Introducción

Las siglas MOOC aluden a cuatro características de la modalidad educativa: (a) es masivo puesto que es capaz de asimilar gran número de alumnos registrados y al mismo tiempo los contenidos tienen alcance global, (b) es abierto dadas las características de gratuidad y la inexistencia de requisitos para acceder a ellos, (c) es online puesto que los cursos que se desarrollan en modalidad habitual utilizan Internet y (d) es un curso que debe contar con una serie de elementos estructurados a fin de orientar los contenidos hacia el aprendizaje.

Según el informe presentado por el New York Time de noviembre de 2012, ese fue el año del surgimiento masivo de los MOOC [1]. La velocidad de su crecimiento tiene su parangón en el fortalecimiento de las redes sociales.

El desarrollo de cursos abiertos masivos online (MOOC) es un fenómeno en pleno auge a nivel mundial. Estos cursos se asientan en plataformas tecnológicas cuyas características funcionales conviene tener en cuenta, más aún, tratándose de desarrollos de código abierto [2]. En este sentido, son importantes los detalles de carácter educativos tales como: (a) principales funcionalidades, (b) formas de analizar las prestaciones formativas para el alumnado en función del uso de la plataforma o (c) procedimientos para medir la percepción de la calidad de las herramientas utilizadas en los MOOC.

La emergencia de las iniciativas de contenido generado por el usuario, en las que el contenido no se entrega a los estudiantes sino que lo crean ellos mismos

conjuntamente, el aumento de las prácticas educativas abiertas (OEP) y los cursos abiertos en línea y masivos (MOOC), y la creación de nuevos proveedores de soluciones de autoaprendizaje como OER University, Peer2Peer University o University of the People están transformando escenarios conocidos en otros dominios de una naturaleza mucho más incierta. Esta tendencia plantea un reto para las instituciones conservadoras, especialmente las universidades. El aprendizaje es cada vez más informal a medida que las personas desarrollan redes complejas para ayudarse entre sí, de modo que intensifican la percepción de que la enseñanza y la formación ya no están proporcionadas exclusivamente por las instituciones, sino que ambas pueden también resultar de una colaboración entre individuos y sus redes especializadas [3: 110].

Tal como lo señalan Koutropoulos y Hogue [4], los MOOC proporcionan una nueva metodología y modalidad para la enseñanza y el aprendizaje. Esta novedad no solo plantea algunos problemas para los estudiantes, sino que también proporciona nuevas e interesantes posibilidades. Algunas de las habilidades de aprendizaje adquiridas por los estudiantes a través de cursos en línea y programas tradicionales son transferibles al aprendizaje MOOC. Sin embargo, la naturaleza distribuida de la MOOC y la enorme cantidad de participantes no permite el método tradicional de acudir al maestro para obtener una explicación. Los MOOC requieren de los estudiantes que sean más proactivos en su educación y en la construcción de sus planes. Todos pueden tener éxito en un MOOC si se toman ciertas medidas y diseñan estrategias antes, durante, y después de un MOOC. Asimismo, de acuerdo con estos autores, las oportunidades que pueden ofrecer los MOOC irán en paulatino y veloz aumento.

1.1. Descripción de los principales tipos de MOOC

Cabero Almenara, Llorente Cejudo y Vázquez Martínez [5], luego de un estudio pormenorizado de las tipologías de los MOOC, y sus implicaciones, señala que, independientemente de las amplias propuestas de MOOC, la tendencia se dirige a considerar dos tipos básicos de MOOC, que suelen denominarse xMOOC y cMOOC. Asimismo un último tipo de MOOC utilizado es el centrado en las tareas, tal como el tMOOC.

Los xMOOC comúnmente se corresponden con los cursos universitarios tradicionales de e-learning, mientras que los cMOOC se apoyan en la filosofía del aprendizaje conectivista de George Siemens y Stephen Downes [5].

Los cMOOC no se centran tanto en los contenidos, sino más bien en comunidades discursivas que crean el conocimiento de forma conjunta (Lugton, 2012, citado en 5). Los MOOC diseñados bajo esta perspectiva están basados en el aprendizaje distribuido en red y se fundamentan en la teoría conectivista y en su modelo de aprendizaje (Siemens, 2007; Ravenscroft, 2011, citados en 5).

El modelo tMOOC responde a un híbrido que adopta los planteamientos de las propuestas xMOOC y cMOOC, y hace especial hincapié en la resolución progresiva de tareas y actividades del estudiante. Las actividades de los tMOOC incluyen variadas propuestas tales como la resolución de casos, lectura y análisis de documentos, construcción de recursos, análisis de sitios web, elaboración de blog y wikis, entre otros, todas ellas actividades que pueden ser desarrolladas tanto

individual como colaborativamente. (Cabero y Román, 2005, citados en 5). Los xMOOC vienen marcados desde la posición objetivista, mientras que los cMOOC desde la conectivista. Por su parte, los MOOC tienen como directriz la visión constructivista.

1.2. Implicaciones de la tipología de MOOC en los pilares de la educación de Delors

Cabero Almenara, Llorente Cejudo y Vázquez Martínez [5] destacan la potencia de los xMOOC y los cMOOC en la adquisición de diferentes competencias de los estudiantes. Desde esta idea, Moya [6] ha analizado ambos tipos de diseños, a la luz de los cuatro pilares de la educación rescatados del informe de Delors [7], tal como son aprender a ser, aprender a aprender, aprender a hacer y aprender a convivir. En la Tabla 1 se muestra la significación que estos dos tipos de diseños de MOOC ofrecen al respecto.

Tabla 1: Pilares de la Educación del Informe Delors y xMooc y cMOOC. (Adaptación de Moya; 2013: 93)

Pilares de la educación	xMOOC	cMOOC
Aprender a ser	Los xMOOC proponen un aprendizaje totalmente individualizado, por lo que dependerá del propio participante su desarrollo. Carácter de formación y aprendizaje para toda la vida: “longlifelearning”.	La propuesta refleja claramente este aprendizaje, ya que implica que la conexión con el resto de los participantes, y sus interacciones, propician el crecimiento y el desarrollo de las personas. Mantiene la esencia del aprendizaje para toda la vida: “longlifelearning”.
Aprender a aprender	El aprendizaje centrado en la información que transmite el docente. Aprendizaje lineal y guiado.	Aprendizaje a partir de compartir el conocimiento con los demás. Aprendizaje activo y participativo.
Aprender a hacer	Las tareas que proponen son más de valorar si se han asumido los contenidos a partir de una autoevaluación. El aprendizaje es pasivo.	Las tareas dependen de la implicación de los participantes y de su relación con el resto. Es un aprendizaje más activo, resaltando el aprender haciendo: “learningbydoing”.
Aprender a convivir	Desde el planteamiento del modelo xMOOC no se contempla esta perspectiva de aprender a convivir, ya que el proceso de aprendizaje es totalmente individual.	La conexión que se establece en esta modalidad de cursos es un buen ejemplo del aprendizaje compartido, colaborativo, cooperativo y por tanto implica relación con el resto de la comunidad del curso.

El MOOC centrado en las tareas, el tMOOC, es un híbrido que adopta planteamientos de las dos propuestas presentadas hasta el momento. Desde esta posición, en el tMOOC se hace especial hincapié en la resolución de determinados tipos de tareas y actividades por parte del estudiante [5].

1.3. La coasociación estudiante docente

Para Prensky [8: 21], los docentes de hoy preparan a sus alumnos para un futuro en el que durante la vida laboral la tecnología se habrá vuelto más poderosa. Cada año el mundo de la información se expandirá, las herramientas se volverán más rápidas y los

centros educativos y los profesores se esforzarán por mantenerse al día. Dados todos estos cambios no resulta una cuestión fácil preparar a los estudiantes para su futuro a largo plazo mientras al mismo tiempo preservan el importante legado del pasado. Sin embargo prima un claro consenso entre los expertos. La forma de obtener éxito en estas condiciones “no consiste en centrarse únicamente en la tecnología cambiante, sino en conceptualizar el aprendizaje de una manera nueva, con adultos y jóvenes asumiendo cada uno papeles nuevos, distintos de los del pasado”.

En este sentido, los estudiantes necesitan centrarse en explorar nuevas herramientas, encontrar información, dar sentido y crear nuevos conceptos. Los docentes por su parte, deben centrarse en preguntar, guiar, proporcionar contexto, garantizar el rigor y el sentido, y asegurar resultados de calidad. La forma de trabajar juntos en el siglo XXI para producir y asegurar el aprendizaje de los alumnos es lo que Prensky [8] llama la coasociación. Para el autor la coasociación es aquella que se presenta al permitir que los alumnos se centren en la parte del proceso de aprendizaje que pueden hacer mejor, y dejar a los profesores que se centren en la parte del proceso de aprendizaje que pueden hacer mejor.

Este paradigma para Prensky [8] significa dar a los estudiantes la responsabilidad de: (a) encontrar y seguir sus pasiones, (b) usar cualesquiera tecnologías que haya disponibles, (c) investigar y recopilar información, (d) responder a preguntas y compartir sus ideas y opiniones, (e) practicar, cuando estén correctamente motivados (por ejemplo a través de juegos) y (f) crear presentaciones en texto y multimedia.

Por su parte, permitir que los profesores hagan lo que pueden hacer mejor significa otorgarles las siguientes responsabilidades: (a) elaborar y hacer las preguntas correctas, (b) asesorar a los alumnos, (c) poner el material curricular en su contexto, (d) explicar de forma individual, (e) crear rigor y (f) asegurar la calidad.

En la pedagogía de la coasociación el objetivo del profesor es no impartir teoría, puesto que ofrece, de variadas formas interesantes, preguntas que responder y sugerencias procedimentales. En la coasociación la responsabilidad se enfoca en que los estudiantes (individualmente o en equipos) busquen, supongan, encuentren respuestas y creen presentaciones sujetas a las evaluaciones posteriores del docente, de los pares o de la clase donde se valoran y examinan por su corrección, contexto, rigor y calidad. Existen diversos niveles de coasociación para adaptarse a distintos tipos de alumnos, situaciones y contextos. En síntesis el profesor no cuenta, sino que pregunta, propone temas y herramientas, aprende de los alumnos sobre tecnología, evalúa los resultados de los alumnos en cuanto a rigor y calidad y proporciona contexto, mientras que el estudiante no toma notas, sino que descubre, investiga y crea resultados y los mejora, añadiendo calidad [8].

1.4. La coasociación y sus implicaciones en la tecnología

El rol de la tecnología en este paradigma, tal como lo afirma Prensky [8] es proveer soporte a la pedagogía de la coasociación y permitir que cada estudiante personalice su proceso de aprendizaje y lo adapte a su forma de aprender. La combinación de clases con muchos alumnos y los pocos recursos han hecho dificultoso para la docencia emprender la personalización y la diferenciación. La mayor de las contribuciones de la llegada de la tecnología logra que los docentes y estudiantes se

asocien en forma más personal e individual, permitiendo a los estudiantes aprender de la forma que prefieran, mientras cumplan los objetivos del curso. Sin embargo, para usar la tecnología con éxito se debe combinar con la coasociación, puesto que este nuevo tipo de pedagogía funciona alineada con la tecnología en toda su extensión. La coasociación propicia el involucramiento de los estudiantes, desde el principio de cada clase, para descubrir solos, y compartir con los demás, materiales curriculares y ejemplos a través de múltiples medios, además de crear y comunicar sus propios aportes alrededor del mundo [8].

1.5. La coasociación y sus implicaciones en la construcción de los MOOC

El nuevo enfoque de la coasociación de Prensky se encuentra alineado con los principios de aprendizaje que se ponen en juego en los cMOOC. Es aquí donde cobra relevancia analizar las posibilidades que ofrecen los cMOOC a la luz de los roles que desempeñan tanto los docentes como los estudiantes en la coasociación de aprendizaje.

La propuesta de los cMOOC refleja claramente el aprendizaje del ser, del aprender, del hacer y el de convivir, ya que implica la conexión con el resto de los participantes, sus interacciones, el crecimiento y el desarrollo de las personas, mantiene la esencia del aprendizaje continuo, y la relación con los demás [6].

La conexión que se establece en esta modalidad de cursos responde a la coasociación en el marco del aprendizaje compartido, colaborativo y cooperativo.

2. Definición del problema

El desarrollo de cursos abiertos masivos online (MOOC) es un fenómeno en pleno auge a nivel mundial. Sin embargo, en varias instituciones educativas, la iniciativa se ha visto demorada por limitaciones que obedecen, entre otros aspectos, a la falta de recursos humanos capacitados y dedicados a este propósito.

La exploración de plataformas MOOC contribuirá con planteos de nuevos desafíos como la elaboración de propuestas en pro de una transposición didáctica más efectiva. Se espera adaptar y validar un instrumento de evaluación de las diferentes propuestas de MOOC.

2.1. Justificación científica, académica-institucional y social

En respuesta a las demandas sociales de oportunidades de capacitación o formación profesional adaptadas a sus necesidades, el estudio propone ofrecer información relevante que conduzca a la capacitación de los docentes en el desarrollo de MOOC. Las limitaciones de los cursos online masivos, demanda una herramienta de evaluación que contemple las principales dimensiones a tener en cuenta a la hora de construir un MOOC.

2.2. Objetivos

1. Impulsar el desarrollo y la implementación de MOOC en los docentes de la UAP.
2. Adaptar y validar un instrumento de evaluación de la calidad de cursos virtuales adaptado a los cursos MOOC construidos.
3. Ofrecer capacitación a los docentes en el desarrollo de MOOC.

2.3. Metodología

El estudio comprende las siguientes etapas: (a) un análisis comparativo de las principales plataformas MOOC, (b) la selección de las dos plataformas que ofrecen más prestaciones, (c) la adaptación y validación de un instrumento de evaluación de la calidad de cursos virtuales mediados por MOOC, (d) la construcción de un módulo de curso bajo las dos plataformas seleccionadas y (e) ejecución de las pruebas estadísticas correspondientes a la evaluación de los MOOC.

A partir de las etapas del estudio se arribará a las conclusiones que permitan brindar información pertinente a quienes manifiesten interés en la construcción de MOOC.

El estudio se lleva a cabo en la Universidad Adventista del Plata (UAP), con una muestra conformada por los alumnos del preuniversitario de la institución.

Cabe señalar que las dos últimas etapas del proyecto se encuentran actualmente en desarrollo, por lo que los resultados finales serán publicados oportunamente.

3. Procedimientos

A continuación se describen los procedimientos llevados a cabo para la ejecución de las etapas a, b y c mencionadas en la metodología, mientras que las etapas d y e se encuentran en desarrollo.

3.1. Análisis comparativo de plataformas

Para realizar la comparación de herramientas de MOOC se han elegido aquellas que son libres y de código abierto, lo que permite ser instaladas y configuradas sin costos para una institución educativa. Asimismo, estas características evitan la dependencia de terceras instituciones para publicar los cursos. Si bien algunas de ellas brindan las dos opciones, como lo es Edx que se encuentra disponible para subir cursos en el MIT y que también puede descargarse e instalar la plataforma en servidores propios.

De las plataformas disponibles en el mercado, las que cumplen con los requisitos mencionados son las siguientes: (a) Lernanta: P2PU (Peer to Peer University de California), (b) OpenMOOC: UNED (Universidad Nacional de Educación a Distancia), (c) edX: MIT (Massachusetts Institute of Technology) y (d) CourseBuilder: Google.

Si bien existen otras plataformas abiertas, estas cuatro, según Pereira, Sanz-Santamaría y Gutiérrez (2014) son las de mayor impacto en el mercado internacional de los MOOC.

Para este trabajo se descarta CourseBuilder debido a que Google no continúa con el proyecto de publicar nuevas versiones al haber aunado sus esfuerzos con el MIT, apoyando la plataforma edX. Por su parte, se incluye Moodle, como plataforma para desarrollo, dado que es la plataforma de educación virtual LMS (Learning Management Systems) que ya posee instalada la universidad que financia la investigación. Si bien no es una plataforma específica de MOOC, fue la plataforma que se utilizó para el primer MOOC y puede adaptarse para soportar MOOC como lo menciona el trabajo del Gobierno Vasco [9].

La Tabla 2 muestra una descripción comparativa de las características de estas herramientas basadas en distintos trabajos de investigación.

Tabla 2: descripción comparativa de herramientas MOOC libres

Características	Lernata	OpenMOOC	edX	Moodle
Videos con subtítulos e indexado de los mismos para posteriores búsquedas	x	x	x	x
ejercicios <i>online</i> de diferentes tipos (de respuesta única, respuesta múltiple, rellenar huecos, texto libre)	x	x	x	x
foros de discusión	x	x	x	x
wiki para edición colaborativa	x	x	x	x
Evaluación automática			x	x
Autoevaluación		x	x	x
Co-evaluación (evaluación de pares)		x	x	x
Sistema integrado de análisis de datos (estadísticas)			x	X (pobre)
Multidiomas		x	x	x
Reutilización de recursos			XBlocks	SCORMS
Importar contenidos de LMS			Moodle	Blackboard
Extensibilidad de herramientas LTI			x	x
Chats			Hangouts	Propio, Skype
Emblemas		x		x
Integrar experimentos A/B en el sistema.			x	
Permite a los profesores crear y probar distintos contenidos de aprendizaje entre grupos de alumnos.				
Mantenimiento y actualización	Regular	Permanente	Permanente	Permanente
Documentación y comunidad de apoyo	Poca	Poca	Poca	Mucha

Fuentes: Pereira, Sanz-Santamaría y Gutiérrez (2014) y Gobierno Vasco (2014).

3.2. Selección de las plataformas

A partir del estudio comparativo de la Tabla 2, la selección de las plataformas para construir MOOC se fundamentó en el cumplimiento de las condiciones deseadas para el trabajo. Las plataformas que reúnen las mencionadas condiciones son edX y Moodle, plataformas disponibles y utilizadas para diseñar un mismo MOOC a ser evaluado desde la calidad percibida por los usuarios.

El MOOC diseñado en ambas plataformas, apela a las directrices fundamentales que deben ser atendidas en el diseño y construcción de un MOOC. Se destaca que el principio fundamental que se contempló en los MOOC fue el de la co-construcción de los saberes entre docentes y estudiantes. Para ello resulta indispensable la incorporación de un curador de contenido en el equipo administrador de un MOOC.

Por su parte, se selecciona la tipología del cMOOC, dado que refleja claramente el aprendizaje del ser, del aprender, del hacer y el de convivir, ya que implica la

conexión con el resto de los participantes, sus interacciones, el crecimiento y el desarrollo de las personas, manteniendo la esencia del aprendizaje continuo y la relación con los demás. La conexión que se establece en esta modalidad de cursos responde a la coasociación en el marco del aprendizaje compartido, colaborativo y cooperativo.

3.3. Adaptación del instrumento de evaluación de MOOC

El instrumento aplicado para evaluar los MOOC resulta de una adaptación del instrumento validado por Mengual-Andrés, Lloret Catalá y Roig Vila [10], al que se le incorporaron dimensiones propuestas por Baldomero, Salmerón y López [11].

El instrumento así adaptado, se sometió al juicio de cuatro expertos con el propósito de obtener evidencia de validez de contenido de las dimensiones adoptadas para el estudio. Finalmente se incorporaron las recomendaciones de los expertos, quedando conformado por:

1. Dimensión 1: calidad en la comunicación y elementos multimedia con 19 ítems.
2. Dimensión 2: Coherencia curricular y adaptación al usuario con 9 ítems.
3. Dimensión 3: Planificación didáctica con 18 ítems.
4. Dimensión 4: Desempeño de los docentes con 9 ítems.

Cabe señalar que las primeras tres dimensiones responden a los aspectos a ser evaluados por los diseñadores de MOOC, mientras que la cuarta dimensión responde a las expectativas del estudiante respecto al desempeño docente, por lo que se reserva esta dimensión para ser evaluada por los estudiantes, una vez finalizado un curso de MOOC (ver Anexo).

Una vez recolectados los datos se procederá a ejecutar las pruebas estadísticas correspondientes a los efectos de identificar diferencias estadísticamente significativas de los valores obtenidos de la evaluación entre los dos grupos de estudio.

4. Referencias Bibliográficas

1. Pappano, L. Noviembre. The Year of the MOOC. The New York Times. (2012). Recuperado de: http://www.nytimes.com/2012/11/04/education/edlife/massive-open-online-courses-are-multiplying-at-a-rapid-pace.html?_r=0
2. Pereira, J., Sanz-Santamaría, S., Gutiérrez, J. Comparativa técnica y prospectiva de las principales plataformas MOOC de código abierto. RED, Revista de Educación a Distancia. n° 44, (2014). “Buenas prácticas de Innovación Educativa: Artículos seleccionados del II Congreso Internacional sobre Aprendizaje, Innovación y Competitividad, CINAIC 2013”. Recuperado de: <http://www.um.es/ead/red/44>
3. Sangrà, A y Wheeler, S. Nuevas formas de aprendizaje informales: ¿O estamos formalizando lo informal? En: La informalización de la educación. RUSC. Universities and Knowledge Society Journal, 10 n° 1, (2013): 107-115. Recuperado de <http://journals.uoc.edu/index.php/rusc/article/view/v10n1-sangra-wheeler/v10n1-sangra-wheeler-es>
4. Koutropoulos, A. y Hogue, R. J. How to succeed in a MOOC-Massive online open course. eLearning Guild, n° 8, (2012). Recuperado de <http://www.cedma->

- europe.org/newsletter%20articles/eLearning%20Guild/How%20to%20Succeed%20in%20a%20MOOC%20-%20Massive%20Online%20Open%20Course%20(Oct%202012).pdf
5. Cabero Almenara, Julio, Llorente Cejudo, M^a del Carmen y Vázquez Martínez, Ana Isabel. Las tipologías de MOOC: su diseño e implicaciones educativas, 18, n° 1, (2014): 13-26. Recuperado de <http://www.ugr.es/~recfpro/rev181ART1.pdf>
 6. Moya López, Mónica. Los MOOC/COMA: un nuevo reto educativo para el siglo XXI. Una metodología didáctica para el aprendizaje en línea. Virtualis, 4 n° 8, (2013): 84-103. Recuperado de <http://aplicaciones.ccm.itesm.mx/virtualis/index.php/virtualis/index>
 7. Delors, J. La educación encierra un tesoro. Madrid: Santillana. (1996).
 8. Prensky, Marc. Enseñar a nativos digitales. Biblioteca INNOVACIÓN EDUCATIVA. (2011). Recuperado de http://innovacioneducativa-sm.aprenderapensar.net/files/2011/09/Ensenar_nativos_digitales.pdf
 9. Gobierno Vasco (2014). Comparación de diferentes plataformas de código abierto. Recuperado de: http://edx.asmoz.org/pdf/estudio_comparativo.pdf
 10. Mengual-Andrés, S., Roig Vila, R., y Lloret Catalá, C. Validación del Cuestionario de evaluación de la calidad de cursos virtuales adaptado a MOOC.RIED. Revista Iberoamericana de Educación a Distancia, 18 n° 2, (2015): 145-169. doi:<http://dx.doi.org/10.5944/ried.18.2.13664>.
 11. Baldomero, M., Salmerón, J. L. y López, E. Comparativa entre instrumentos de evaluación de calidad de cursos MOOC: ADECUR vs Normas UNE 66181:2012. RUSC.Universities and Knowledge Society Journal, 12 n°1, (2015): 131-145.

5. Anexo: Instrumento de evaluación de MOOC

Dimensión 1: Calidad en la comunicación y elementos multimedia					
	Nunca	Casi nunca	A veces	Casi siempre	Siempre
La información que ofrece el MOOC al usuario es interactiva.					
La información que ofrece el MOOC es actualizada.					
La información que ofrece el MOOC es relevante.					
Se emplean diversos códigos comunicativos (verbal, icónico, etc.).					
Se puede recurrir a un sistema de ayudas y refuerzos constantes sobre los contenidos.					
Las actividades propuestas en el sistema resultan atractivas.					
Las imágenes contenidas tienen buena resolución.					
Las imágenes y gráficos se adecuan al texto.					
La resolución de las animaciones es buena.					
Las animaciones se adecuan al texto.					
Las presentaciones audiovisuales son atractivas.					
Las presentaciones audiovisuales se adecuan al tema.					
Los mensajes de audio son entendibles.					
Los mensajes de audio son audibles.					
La ortografía y la gramática del texto son correctas.					
Se muestran los indicadores de avance para cada estudiante (insignias).					
La sección de preguntas frecuentes y/o ayuda resulta de utilidad para el estudiante.					
La navegabilidad del MOOC es intuitiva.					
La navegabilidad del MOOC es ágil.					

(1 = nunca; 2 = casi nunca, 3 = a veces, 4 = casi siempre, 5 = siempre)

Dimensión 2: Coherencia curricular y adaptación al usuario					
	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Las actividades son coherentes con la metodología planteada.					
Los contenidos responden a los objetivos planteados.					
La evaluación es coherente con la metodología aplicada.					
La relación entre lo fundamental y lo accesorio en la información está claramente definida.					

Los contenidos se presentan de forma hipertextual.					
Los conceptos nuevos se introducen mediante algún tipo de organizador como esquemas, resúmenes, síntesis.					
El lenguaje usado está adaptado al nivel del usuario potencial.					
El sistema contempla las características y circunstancias personales y/o particulares de los usuarios (incluida alguna discapacidad).					
Los textos y los contenidos cumplen el mandato constitucional de no hacer distinción o discriminación por razón de nacionalidad, etnia, género, religión, opinión o cualquier otra condición o circunstancia personal o social.					

(1 = nunca; 2 = casi nunca, 3 = a veces, 4 = casi siempre, 5 = siempre)

Dimensión 3: Planificación didáctica					
	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Existe una guía didáctica del curso					
En la guía didáctica (si existe) se expresa claramente cómo se integran los contenidos en el proceso de enseñanza-aprendizaje.					
Los objetivos se plantean explícitamente.					
Existen actividades de refuerzo.					
Las actividades de refuerzo, cuando las hay, permiten superar las posibles deficiencias que se han detectado en la evaluación.					
Se plantean actividades abiertas que fomenten la creatividad.					
El sistema dispone de distintos itinerarios de aprendizaje.					
Existen distintos niveles de contenidos en función de los usuarios.					
Los contenidos de los mensajes de interacción con el usuario son provechosos.					
El MOOC promueve la indagación de los estudiantes en búsqueda de nuevos aprendizajes significativos.					
Los estudiantes son movilizados a buscar nueva información.					
Los tiempos disponibles para el progreso académico se adaptan a los ritmos de cada estudiante.					
El MOOC ofrece oportunidades de autoevaluación y co-evaluación.					
Se plantean actividades individuales y grupales.					
Existe variedad de técnicas didácticas para el cumplimiento de los objetivos, como por ejemplo, desarrollo de proyectos o aprendizaje basado en problemas reales.					
Se proporcionan orientaciones para la generación de nuevos materiales o recursos educativos.					
Presenta secuencias o itinerarios de repaso.					
Las actividades permiten conseguir los objetivos planteados.					

(1 = nunca; 2 = casi nunca, 3 = a veces, 4 = casi siempre, 5 = siempre)

Dimensión 4: Desempeño de los docentes (estudiantes una vez finalizado el curso)					
	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Las respuestas a las consultas sobre los contenidos del curso se reciben en el tiempo preestablecido.					
Los tutores realizan un seguimiento de los aprendizajes.					
La orientación que ofrece el docente es clara y adaptada al nivel de comprensión del estudiante.					
La realimentación recibida del docente sobre el desempeño del estudiante contribuye con el aprendizaje.					
Ante un cierto tiempo de inactividad del estudiante, el docente se comunica, mediante mensajes, con el propósito de alentar el progreso del aprendizaje.					
La modalidad de enseñanza aplicada por el docente facilita el alcance de los objetivos.					
La calendarización de las asignaciones optimiza el proceso de aprendizaje de los contenidos.					
Las respuestas del profesor resuelven las dudas del estudiante.					
Las intervenciones del profesor/tutor resuelven eficazmente los problemas que surgen durante el desarrollo del curso.					

(1 = nunca; 2 = casi nunca, 3 = a veces, 4 = casi siempre, 5 = siempre)

El discurso en entornos virtuales de aprendizaje: categorías de análisis de los desempeños escritos

Néstor Blanco¹, Nora Cuello¹, Gabriela González¹, Paula Penco¹

¹ Instituto Tecnológico para la Calidad (ITC), Facultad de Ingeniería de la Universidad Nacional de Lomas de Zamora (UNLZ), Lomas de Zamora, Buenos Aires, Argentina
Néstor Blanco, nestorhugoblanco@gmail.com; Nora Cuello, norapatriciacuello@gmail.com;
Gabriela González, gabriela.fi.unlz@gmail.com; Paula Penco, paucecipenco@gmail.com

Resumen. Las Tecnologías de la Información y la Comunicación (TIC) se han ido insertando cada vez más en el ámbito educativo, dando lugar a entornos virtuales de aprendizaje. En estos espacios el docente y el alumno interactúan mediante distintas herramientas de comunicación, las cuales se desarrollan a través de desempeños escritos, donde los alumnos tienen un papel activo y el docente debe tener una acción tutorial de guía en el proceso de aprendizaje. Así, mediante estos desempeños se espera el logro de la construcción de conocimientos significativa y colaborativamente. Una herramienta de análisis de los desempeños escritos son las categorías discursivas, las cuales nos permiten estudiar cómo se construye el conocimiento en estos entornos virtuales. En el presente trabajo se recuperan las categorías de análisis del discurso enunciadas por De Pedro Puente (2006) y se crean nuevas categorías para considerar los contenidos escritos que no son abordados por el autor de referencia.

Keywords: categorías de análisis del discurso, desempeños escritos, entornos virtuales, TIC.

1 Introducción

En la actualidad, y desde hace unos años atrás, vivimos en una sociedad atravesada por las Tecnologías de la Información y la Comunicación (TIC). De esta forma, la educación no es, ni puede ser, ajena a todos estos avances tecnológicos. Si estas nuevas herramientas llegaron a entrelazarse en la vida diaria de las personas, no debían dejar de ser incluidas en la educación, con la finalidad de adentrarla en el mundo y la sociedad en que hoy vivimos.

La llegada de estas tecnologías al mundo educativo está generando nuevos espacios de trabajo conocidos como Entornos Virtuales de Enseñanza-Aprendizaje (EVEA o EVA). La educación en estos entornos se está configurando como una nueva modalidad de educación a distancia, que se caracteriza por la integración de la tecnología en el proceso educativo. Los EVA se definen, generalmente, como un proceso o actividad de enseñanza-aprendizaje que se desarrolla fuera de un espacio físico y temporal, y a través de internet. Ofrecen

diversidad de medios y recursos para apoyar la enseñanza, y son la base tecnológica que da sustento funcional a las diversas iniciativas de teleformación.

Estos nuevos modelos de aprendizaje, basados en un alumno más activo y estrategias para aprender a aprender, implican cambios sustanciales en el papel del docente, que deja de ser la principal fuente del conocimiento para el alumno y pasa a tener un rol de tutor/orientador. Este rol apunta a lograr, en el alumno, competencias como ser la autonomía, el trabajo en equipo, la toma de decisiones, la creatividad, la reflexión y autorreflexión, la empatía. Así, se desarrolla un rol del docente como guía y planificador, y un rol activo del alumno quien debe construir su propio aprendizaje.

Sin embargo, no podemos olvidar que estos entornos virtuales no determinan por sí mismos los modelos ni las estrategias didácticas a implementar, ya que el conocimiento o acceso a estos recursos tecnológicos no exime al profesorado del conocimiento profundo de las condiciones de aprendizaje, ni del adecuado diseño y planificación docente.

Es destacable que la mayor parte de las TIC utilizadas en el ámbito educativo implican un desempeño escrito, un discurso o diálogo, tanto de docentes como alumnos. Como tipo de enunciación, el discurso es una comunicación lingüística, que necesita, indispensablemente, la existencia de un emisor y un receptor.

Así, las comunicaciones que se establecen entre el docente y el alumno es una de las características definitorias de los modelos formativos virtuales. Mientras que en situaciones presenciales predomina la comunicación verbal y gestual, en los entornos virtuales prima la comunicación textual, aunque la posibilidad de que se produzcan redes de comunicación bidireccional son mayores, más detalladas y ricas que las que permiten las situaciones presenciales.

2 Marco Teórico

Las TIC vienen a insertarse en la educación, donde la enseñanza puede ser concebida como un diálogo entre dos o más partes. Jonas F. Soltis, en el prefacio a la obra de Burbules, N. (1999), llamada *El diálogo en la enseñanza*, expone que “(...) el diálogo es una relación comunicativa simbiótica entre iguales, que exige un compromiso tanto emocional cuanto cognitivo. (...) se sustenta en una inteligencia cognitiva, pero también en los sentimientos recíprocos de interés, confianza, respeto, aprecio, afecto y esperanza de los participantes”. [1] Así, Burbules considera el diálogo como una relación comunicativa y social de la que se forma parte y en la que se entra, y no como algo instrumental, que hagamos o que empleemos, siendo atravesados por aspectos que están más allá de nosotros, que descubrimos y que nos modifican. Crear una relación dialógica supone relacionarse con otra u otras personas, construyendo lazos emocionales como el respeto, la confianza y el interés, y expresiones virtuales como la paciencia, la capacidad de escuchar y la tolerancia ante el desacuerdo.

Además, muchos estudios han demostrado que, aunque al principio parezca difícil, la experiencia confirma que existe una comunicación intensa entre el docente y los estudiantes en estos entornos virtuales, además de que ésta es una comunicación expresiva, cercana y cálida.

Retomando la idea de la enseñanza como un diálogo o intercambio discursivo entre dos o más partes, citamos a Lavandera, coincidiendo con ella: “Nosotros vamos a usar *discurso* como palabra más amplia, más general. *Texto*, lo usaremos para distinguir, en algún caso, lo producido en un determinado momento del discurso. *Discurso* será entonces simplemente habla emitida...”. [2]

De esta manera, y en relación a los principios de la concepción constructivista del conocimiento, este intercambio discursivo o diálogo permitirá distintas intervenciones donde entender o comprender “presupone incorporar información nueva a los esquemas existentes o modificar estos esquemas a la luz de la información nueva o los contenidos nuevos” [1]. Por otra parte, “el proceso de andamiaje en la enseñanza muestra que no necesariamente existe incompatibilidad entre un papel significativo para el maestro en el diálogo y una concepción activa y respetuosa del alumno” [1]. En el contexto actual, esa relación dialógica no se expone sólo desde el lenguaje oral, sino también desde el lenguaje escrito. Es más, con la llegada de las TIC, la educación se ha visto atravesada por las tecnologías puestas al servicio de ella; así, el lenguaje escrito se ha tornado el medio más usado para desarrollar los procesos de enseñanza y aprendizaje, ya que “(...) las TIC permiten que la distancia física y la no coincidencia en el tiempo (la asincronía) no sean una traba a la comunicación y el aprendizaje. (...) el EVEA permite no sólo la comunicación entre todos los participantes en la formación, sino también que exista la “vidilla” que suele haber en cualquier aula: que cada estudiante aprenda en solitario y a la vez en compañía, que existan debates, intercambios de opinión, preguntas, aclaraciones y que también se pueda dar una relación entre docentes y estudiantes, y de éstos entre sí (...). Lo importante no es el medio sino la voluntad de comunicarse y relacionarse”. [3]

Esta voluntad de comunicarse y relacionarse se puede dar de diversas maneras, mediante una variedad de distintos tipos de enunciados, como puede ser aportando una organización sobre cómo encarar y/o continuar el debate; definiendo conceptos; agregando una postura propia y/o información nueva acerca de un tema; citando de una fuente; mencionando una justificación; compartiendo un enlace a alguna fuente de interés; agregando un adjunto vinculado al tema en cuestión; reflexionando sobre el tema que se debate o algún otro vinculado a él; redactando una síntesis de lo aportado por otro/s participante/s; aportando una hipótesis nueva o formulando preguntas que puedan provocar un giro en el debate; rememorando lo expresado por otro/s participante/s para realizar una opinión/crítica positiva o negativa sobre ello; recuperando lo expresado por otro/s participante/s para rescatar una información que se considera destacable en el debate; generando respuestas sencillas a dudas o consultas de otro/s participante/s; mencionando preguntas elaboradas y/o nuevas vías para avanzar en el debate; planteando una actividad; pidiendo una aclaración sobre algo aportado por otro/s participante/s; aclarando una información que no se comprendió, aportada por uno mismo o por otro/s participante/s; mencionando mensajes de motivación; como también utilizando combinaciones de signos que dan lugar a los conocidos emoticones, con la intención de expresar estados de ánimo en forma escrita.

Estos distintos tipos de enunciados quedan reflejados en diferentes categorías, que nos serán de utilidad para analizar el discurso. Y, dentro del ámbito educativo, para evaluar cómo se desarrolló el desempeño escrito de los alumnos, con la guía del docente, y si esos desempeños lograron alcanzar la competencias que se esperan del

alumno, antes mencionadas, mediante una construcción de conocimientos en forma colaborativa y significativa.

3 Metodología

El diseño metodológico de la presente investigación es un diseño no experimental. Se busca describir, explicar y predecir la realidad, desde una aproximación a su dinámica natural.

Dentro del diseño no experimental, nos adecuamos a un estudio longitudinal, que tiene la ventaja de proporcionar información sobre cómo las variables y sus relaciones evolucionan en distintas unidades de tiempo, en múltiples mediciones, tratándose, además, de un estudio cualitativo.

El estudio se centró en distintos desempeños escritos dentro de un entorno virtual de aprendizaje, mediante herramientas de comunicación virtual, como el foro y el chat. En principio, para realizar el análisis del discurso se tomaron las categorías propuestas por De Pedro Puente, X. (2006), en su ponencia “Cómo evitar el ‘café para todos’ al evaluar trabajos en grupo, y de paso, estimular el aprendizaje reflexivo: resultados preliminares en el marco del proyecto A Wiki Forum”. [4]

Se empleó, como técnica e instrumento de recogida de datos, el grupo de discusión, formado por un grupo reducido de investigadores, mediante el cual se discutió acerca de las categorías y su definición, buscando producir un discurso social que se deriva del consenso de los integrantes del grupo sobre el tema de interés.

4 Resultados

A partir de las interacciones logradas, a continuación se enuncian dichas categorías, respetando la sigla asignada por el autor de la fuente. Claro que, para delimitarlas y apropiárnoslas, fue necesario agregar una definición propia:

Tabla 1. Categorías de análisis del discurso de De Pedro Puente, X. (2006)

CATEGORÍA / TIPO DE CONTRIBUCIÓN	SIGLA	DEFINICIÓN
* Aspectos organizativos	AO	Exposición de la organización de los contenidos que se deben presentar.
* Peticiones de ayudas	PS	Petición de comentarios, experiencias, precisiones o explicaciones a otro/s participante/s sobre los significados aportados por quien enuncia la petición.
** Ayudas a compañeros	AC	Requerimiento o solicitud para que se aporten significados sobre un tema o sobre los significados

		aportados por quien enuncia el requerimiento (AC-rq).
		Respuesta a un requerimiento o solicitud para aportar significados sobre un tema o sobre significados aportados por quien ha enunciado el requerimiento (AC-rrq).
** Nueva información	NI	Expresión de significados propios o presentados como propios (NI-sp).
		Expresión de significados atribuidos a fuentes externas (NI-sf).
** Reflexión personalizada y argumentada	RP	Exposición, mediante las propias palabras de un participante, de determinados pensamientos sobre un/os aporte/s anterior/es, argumentando y enunciando alguna/s conclusión/es.
*** Hipótesis nuevas	HN	Enunciado que se considera fiable o creíble, en el cual se identifican tópicos o temas de atención, indagación y discusión.
*** Preguntas elaboradas y nuevas vías para avanzar	PE/NV	Expresión de dudas, preguntas y/o nuevas ideas en pos de lograr avances en el tema de discusión.
*** Síntesis/ elaboración de información	SI	Síntesis, resúmenes o recapitulaciones integrando conceptos relevantes sobre el tratamiento de un tema, aportados por cualquiera de los participantes.
* Respuestas sencillas	RS	Respuestas a interrogantes planteados por otro participante.
* Definición de conceptos	DC	Definición de conceptos/términos presentados por el propio participante o requerido por otro/s (DC-c).
		Recordatorios de significados propios o de otro/s participante/s (DC-rs).
** Réplica	REP	Copia exacta (o muy parecida) del aporte de otro/s participante/s (REP-ca).

		Expresión o discurso en el que se ponen obstáculos, se refuta o se dice lo contrario sobre un argumento o respuesta expuesto por otro/s participante/s (REP-obs).
* Referencias de ampliación	REF	Referencia a fuentes de significados, como libros, artículos, entrevistas, videos, páginas web (REF-rf).
		Referencia a significados atribuidos a la autoría de un texto o contenido de interés (REF-rfa).
* Opinión de personal	OP	Valoración favorable de los significados aportados por otro/s participante/s (OP-vf).
		Valoración crítica de los significados aportados por otro/s participante/s (OP-vc).
** Refuerzo de información	RI	Aportes de significados mediante adjuntos (RI-adj).
		Aportes de significados mediante experiencias propias (RI-ep).
		Aportes de significados mediante experiencias ajenas, comentadas o sólo aludidas (RI-ea).
* Facilitación	F	Identificación y/o corrección de errores o incomprensiones en pos de facilitar los significados aportados por alguno de los participantes.
* Agradecimientos	A	Manifestación de gratitud por un bien o atención recibidos de otro/s participante/s.

Fuente: Elaboración propia

Es importante destacar, siendo de gran interés para nuestro tema de investigación, los asteriscos delante de cada categoría indican, siguiendo al autor de referencia, el nivel de contribución que aportan. Así, cuantos más asteriscos hay delante de la “Categoría / Tipo de contribución” mayor importancia se da a la contribución, en cuanto a la interacción que provoca en pos de la construcción de conocimientos.

De nuestra parte, adoptamos esta categorización con sus siglas de identificación y los asteriscos. Hemos realizado una definición de cada una de ellas, teniendo en cuenta distintas variantes que pueden presentarse dentro de una misma categoría discursiva; en estos casos, la sigla es dividida en cuantas variantes se hayan mencionado, como podemos ver en la Tabla 1.

Cada una de las categorías se diferencian claramente entre sí, salvo dos subcategorías muy parecidas en su definición, pero que procedemos a distinguir más detalladamente

para evitar confusiones: en el caso de la categoría NI-sf, se refiere al discurso que menciona contenido de un autor/fuente, siendo citado directa o indirectamente. En cambio, la categoría REF-rfa implica un discurso en el que se menciona una autoría, pero sin detallar el contenido de interés.

Además, es importante mencionar que, bajo análisis realizados en foros y chats, nos encontramos con intervenciones que aportan a la construcción del conocimiento, pero sin encuadrar en ninguna de las categorías propuestas por el autor referenciado anteriormente.

Por este motivo, luego de sucesivos análisis hemos creado distintas categorías, que detallamos a continuación:

Tabla 2. Categorías de análisis del discurso de autoría propia

CATEGORÍA / TIPO DE CONTRIBUCIÓN	SIGLA	DEFINICIÓN
* Postura propia	PP	Exposición de un punto de vista/postura/opinión personal acerca de un tema o aspecto de lo debatido, sin argumentos ni justificación.
** Propuestas de trabajo	PT	Exposición de propuestas, ideas o acciones vinculadas con el tema de discusión.
* Mención	M	Recuperación, a modo de mención (con palabras propias o citadas), de un aporte o idea de otro participante.
* Comentarios personales	CP	Comentarios acerca de cuestiones propias del participante, las cuales exceden el tema de debate (CP-off).
		Comentarios o indicaciones que, si bien refieren a formalidades del debate, no continúan la discusión del tema en cuestión (CP-ind).
* Motivación	MO	Enunciados de ánimo y/o invitación a otro/s participante/s a que participen en el debate (MO-an)
		Enunciados de apelación a otro/s participante/s para que intervengan en el debate (MO-apel).
		Enunciados resaltando aprobación en el desempeño de uno o más participantes en un momento preciso del debate (MO-aprob).
* Recordatorios	REC	Enunciado que recuerda una petición, interrogación o cuestión mencionada anteriormente, la cual ha quedado sin debatir.

* Emoticones	EM	Refuerzo emotivo de una intervención o comentario propio (EM-rp)
		Empatía o apoyo emotivo a lo dicho por otro/s participante/s (EM-emp).
		Acompañamiento emotivo del cierre de una intervención propia (EM-c)
* Estados de ánimo	EA	Enunciados que indican expresiones de estados de ánimo, como risa (“jajaja”), picardía (“jeje”), desánimo (“uy”), sorpresa (“guau”)

Fuente: Elaboración propia

Para llevar a cabo análisis del discurso mediante estas categorías, se aconseja que que cada categoría pueda aparecer una sola vez por intervención, es decir, cada intervención puede ser categorizada por más de una categoría, pero no por categorías repetidas.

5 Conclusión

Este trabajo muestra y ofrece herramientas para analizar los desempeños escritos en entornos virtuales, pudiendo determinar en qué magnitud cada categoría de análisis del discurso propuesta contribuye a la construcción de conocimiento significativa y colaborativamente, y reflexionar sobre ellos.

Como ya hemos mencionado por medio de los asteriscos, los intercambios en donde se refleje un discurso con mayor cantidad de asteriscos, significará, por lo general, un rol activo en el alumno, logrando competencias como ser la autonomía, el trabajo en equipo, la toma de decisiones, la creatividad, la reflexión y autorreflexión, la empatía. Mientras el rol del docente es de guía y planificador, el del alumno apunta a lograr construir su propio aprendizaje.

Esta comunicación textual se sustenta en una inteligencia cognitiva, pero también en los sentimientos recíprocos de interés, confianza, respeto, aprecio, afecto y esperanza de los participantes. Crear una relación dialógica supone relacionarse con otra u otras personas, construyendo lazos emocionales como el respeto, la confianza y el interés, y expresiones virtuales como la paciencia, la capacidad de escuchar y la tolerancia ante el desacuerdo.

De esta manera, y en relación a los principios de la concepción constructivista del conocimiento, el diálogo permitirá distintas intervenciones donde entender o comprender “presupone incorporar información nueva a los esquemas existentes o modificar estos esquemas a la luz de la información nueva o los contenidos nuevos”.

[1]

En fin, el desarrollo escrito puede desarrollarse de muchas maneras distintas, como podemos notar mediante las posibles categorías del discurso que manifiesten los participantes. Y, en tanto sea en forma expresiva, cercana y cálida, y con un accionar

comprometido, tanto desde el lugar del docente como del alumno, será una interrelación que apuntará a construir ese aprendizaje esperado.

6 Referencias

1. Burbules, N.: El diálogo en la enseñanza. Teoría y práctica. Amorrortu Editores S.A., Buenos Aires (1999)
2. Lavandera, B. R.: Curso de lingüística para el análisis del discurso. Centro Editor de América Latina, Buenos Aires, Argentina (1985), <https://www.scribd.com/doc/132883182/BEATRIZ-LAVANDERA-CURSO-DE-LINGUISTICA-PARA-EL-ANALISIS-DEL-DISCURSO-BEATRIZ-LAVANDERA>
3. Bautista, G., Borges, F. y Forés, A.: Didáctica universitaria en Entornos Virtuales de Enseñanza-Aprendizaje. Narcea, Madrid (2006)
4. De Pedro Puente, X.: Cómo evitar el “café para todos” al evaluar trabajos en grupo, y de paso, estimular el aprendizaje reflexivo: resultados preliminares en el marco del proyecto A Wiki Forum”. En: Jornada Espiral (2006)

Herramientas de Autor para la creación de Objetos de Aprendizaje. ESTADO DEL ARTE.

Lucía Violini¹, Cecilia Sanz¹

¹ Instituto de Investigación en Informática LIDI,
Facultad de Informática, Universidad Nacional de La Plata.
La Plata, Buenos Aires, Argentina.
{lviolini, csanz}@lidi.info.unlp.edu.ar

Resumen. En este artículo se presenta una revisión sobre Herramientas de Autor (HAs), haciendo foco particularmente en HAs para la creación de Objetos de Aprendizaje (OAs). Además se establecen criterios que permiten definirlos y analizarlos, como resultado de ello se presenta un análisis comparativo sobre una selección de herramientas. El trabajo realizado refleja aspectos centrales sobre HAs, que pueden servir de guía para las decisiones de docentes que inicien su camino en la creación de OAs. Al mismo tiempo, se observa la necesidad de contar con HAs más específicas para la creación de OAs, que consideren la carga de metadatos y el empaquetamiento respetando estándares y que acompañen tanto el diseño como la creación de OAs.

Palabras clave: Herramientas de Autor, Objetos de Aprendizaje, Materiales Educativos Digitales.

1 Introducción

El avance de la integración de las Tecnologías de la Información y la Comunicación (TICs) en Educación ha llevado a los docentes a repensar los materiales educativos existentes y a crear nuevos materiales en formatos digitales. Este proceso, sin embargo, no es sencillo; no sólo exige conocer las herramientas informáticas de edición digital sino también conocer las nuevas formas de descripción, organización, presentación y uso de los contenidos en formatos digitales, y las posibilidades que estos lenguajes ofrecen [1].

La producción de Materiales Educativos Digitales (MEDs) implica la toma de decisiones didácticas, comunicacionales y tecnológicas. Una tendencia actual es abordar el diseño pensando en la reutilización de los MEDs, y es por ello que los Objetos de Aprendizaje (OAs) han cobrado especial atención en esta línea de investigación [2].

Aún no existe consenso, por parte de los autores especializados en el tema, al momento de determinar qué es un Objeto de Aprendizaje (OA) y qué no lo es. Hay un largo recorrido en la conceptualización de OA [3]. El concepto adoptado para este

trabajo es: “*Un OA es un tipo de material educativo digital, que se caracteriza, desde el punto de vista pedagógico, por orientarse a un objetivo específico de aprendizaje, y presentar mínimamente: una serie de contenidos con el fin de abordar la temática relacionada con el objetivo, actividades que permitan al alumno poner en práctica o problematizar el contenido presentado, y una autoevaluación que posibilite conocer al alumno, si ha podido comprender esos contenidos vinculados al objetivo. Desde el punto de vista tecnológico, se caracteriza por contener un conjunto de metadatos estandarizados para su búsqueda, y recuperación, y estar integrado, utilizando un modelo de empaquetamiento que respete estándares, y de esta manera, permita su diálogo con diferentes entornos tecnológicos.*” [4].

La falta de una definición formal del concepto de OA ha dado lugar al surgimiento de una serie de Metodologías de diseño de OAs creadas bajo distintos puntos de vista. Hay metodologías que consideran mayormente aspectos tecnológicos, por ejemplo la *Metodología MESOVA* [5]. Otras metodologías destacan aspectos educativos, como la *Metodología ISDMELO* [6]. Y existen también, aunque en menor medida, metodologías que alcanzan enfoques híbridos, combinando de manera equilibrada aspectos tecnológicos y educativos, como es el caso de la *Metodología CROA* [4] [7], la *Metodología DINTEV* [8] y la *Metodología TECNOPEDAGÓGICA* [9]. Sin embargo, más allá de la perspectiva que cada metodología posea, la mayoría tiene una misma finalidad, guiar a los docentes en las tareas involucradas en el diseño de OAs. En general, las metodologías coinciden en proponer etapas que plantean la planificación y el diseño de los OAs, y algunas de ellas además consideran la implementación.

Cuando los docentes desean concretar la creación de OAs necesitan hacer uso de Herramientas de Autor (HAs) destinadas a tal cuestión. En la siguiente sección se profundiza este concepto. El trabajo que aquí se presenta se enfoca en el análisis de HAs para la creación de OAs a partir de la definición de una serie de criterios que permiten su definición y análisis.

De acá en adelante este artículo se organiza de la siguiente manera. En la sección 2 se presenta una definición del concepto de Herramienta de Autor (HA), se describen las características principales que éstas poseen y se hace foco en las particularidades que tienen las HAs para la creación de OAs. En la sección 3 se describen los criterios propuestos para definir y analizar HAs para la creación de OAs, y se presenta el análisis comparativo realizado sobre una selección de herramientas. En la sección 4 se presenta una discusión sobre las posibilidades actuales de determinadas HAs para la creación de OAs. Finalmente, en la sección 5 se detallan las conclusiones a las cuales se ha arribado y se proponen líneas de trabajo futuro.

2 Herramientas de Autor

Según Gómez Villa [10], las HAs son “*aquellas que permiten, mediante un proceso más o menos complejo de compilado, la generación de un programa que funciona independientemente del software que lo generó*”, y en un sentido más específico, se puede incluir dentro de este grupo a “*todas aquellas herramientas que permiten*

generar actividades, materiales y recursos en formato multimedia” [11]. Facilitan a diseñadores instruccionales, educadores, maestros y aprendices, el diseño de materiales educativos multimediales sin conocimiento experto en Informática [12].

Las HAs orientadas al ámbito educativo tienen como objetivo principal permitir la generación de contenidos y/o actividades educativas digitales. Se caracterizan por ser de fácil utilización, la mayoría están diseñadas de manera tal que aquellos docentes que decidan utilizarlas no requieran amplios conocimientos sobre manejo de herramientas de software ni sobre programación. Suelen ser amigables en su interfaz y de manejo intuitivo a la hora de generar los contenidos y/o actividades educativas. Proporcionan plantillas para mostrar y organizar contenidos, y/o plantillas para armar actividades. Cada docente puede personalizar las plantillas que utilice de acuerdo a los objetivos educativos que se plantea.

Existen numerosas HAs, algunas con propósitos más específicos que otras. Por un lado, se encuentran aquellas herramientas de propósito general, y por otro lado, las herramientas de propósito específico.

En el primer caso, las HAs de propósito general, proveen plantillas de contenidos y/o actividades genéricas, no apuntan exclusivamente a ningún tema. Los docentes deberán encargarse de seleccionar aquellas plantillas que sean de su interés y adecuarlas al contexto educativo en el cual las van a utilizar. Los tipos de plantillas que proporcionan estas herramientas son muy variados. En cuanto a plantillas de contenidos se pueden encontrar: libros virtuales, galerías de imágenes, reproductores de audio y video, entre otros; y con respecto a plantillas de actividades: sopas de letras, crucigramas, rompecabezas, palabras cruzadas, actividades con Realidad Aumentada, preguntas-respuestas, *test*, sólo por mencionar algunas. Por ejemplo, *Constructor* [13] y *Jcllic* [14] son HAs de propósito general. En el segundo caso, las HAs de propósito específico, se caracterizan por estar especializadas en un tema (o área) determinado. Estas herramientas no proveen plantillas genéricas sino que las plantillas de contenidos y/o actividades se enfocan en el área temática en cuestión. Hay herramientas de este tipo destinadas a: matemática, idiomas, etc. Por ejemplo, *Malted* [15] y *Reyuela* [16] son HAs de propósito específico.

2.1 Herramientas de Autor para la creación de Objetos de Aprendizaje

Cuando se quiere crear OAs es necesario considerar distintos aspectos, tales como: el objetivo instruccional, las características de interacción, interfaz, formato, descriptores, además de todos aquellos propios del diseño instruccional. Al mismo tiempo, están implicados varios procesos como: generación, búsqueda y recuperación, etiquetado, composición, entre otros. Generalmente el docente realiza estos procesos de una forma empírica y utilizando herramientas informáticas independientes, desarrolladas específicamente para tareas concretas de otra índole, lo que limita y complica su utilización [17]. Las HAs para la creación de OAs tienen un rol importante en este sentido, debieran ser un medio para que la mayoría de los docentes que lo deseen puedan crear sus OAs sin encontrar mayores dificultades.

Estas HAs se caracterizan por permitir la generación de OAs, ya sea dando la posibilidad de implementar cada uno de los elementos que compondrán al OA o integrando al OA los elementos generados con otras HAs.

Existen diversas HAs para la creación de OAs, algunas sólo admiten la implementación de OAs y otras no sólo consideran la etapa de implementación sino que contemplan las demás etapas que conforman alguna Metodología de diseño de OAs específica.

En el primer caso, las HAs para la creación de OAs sin metodología de base, delegan a los docentes las tareas de diseño de OAs, sólo se centran en permitir la implementación de los mismos. En ocasiones su uso resulta dificultoso para docentes que no acostumbran a manejar este tipo de herramientas, no sólo por su manejo, a veces poco intuitivo, sino también por la falta de guías para diseñar OAs. Al usar estas herramientas la creación de OAs puede volverse una cuestión bastante técnica (importación de recursos, carga de metadatos, generación de paquetes), lo que se convierte en un obstáculo para docentes poco familiarizados con la tecnología. No sucede lo mismo con docentes que hacen uso de la tecnología para desarrollar MEDs, que conocen las cuestiones básicas de manejo de herramientas de software o que, mejor aún, tienen conocimientos de programación. Docentes con estas características tienen mayores posibilidades de obtener muy buenos resultados en la creación de OAs al usar estas herramientas. En el segundo caso, las HAs para la creación de OAs con metodología de base, permiten a los docentes llevar a cabo todas y cada una de las etapas que involucra una Metodología de diseño de OAs en particular. Una misma herramienta permite realizar el diseño y también la implementación de OAs, y así brinda elementos que permiten construir un OA completamente. Las particularidades de cada herramienta de este tipo, sus prestaciones y su potencialidad, estarán estrechamente ligadas a la Metodología de diseño de OAs que la herramienta haya adoptado. Algunas herramientas serán más completas que otras, pero difícilmente comparables si cada una optó por seguir una metodología distinta. Existen pocas herramientas de este tipo en comparación con las herramientas que no consideran una metodología, a pesar de los importantes beneficios que traería para los docentes contar con ellas en sus procesos de creación de OAs.

En la próxima sección se presenta el núcleo de este trabajo. El mismo se vincula a una propuesta de criterios que permiten definir y analizar HAs orientadas a la creación de OAs, y su aplicación en un análisis comparativo concreto.

3 Análisis de Herramientas de Autor

Diwakar y Patwardha [18] discuten lo difícil que resulta para los profesores seleccionar alguna HA que cumpla con los criterios requeridos en su quehacer docente (diseño instruccional, calidad, personalización, entre algunos) y con un entorno que les sea amigable en su manejo [19]. Es por ello que arrojar luz en los criterios de importancia a la hora de analizar diferentes HAs constituye un aporte para los docentes que se encuentran en el camino de creación de MEDs en general.

Como parte de una investigación sobre OAs, se ha realizado el estudio de diferentes HAs y se han probado varias de ellas. Al mismo tiempo, se han establecido una serie de criterios que permiten la definición y el análisis de HAs para la creación de OAs.

3.1 Criterios de consideración

Los criterios de consideración establecidos en este trabajo tienen como finalidad definir/determinar qué HAS posibilitan la creación de OAs, ya que tienen en cuenta aspectos fundamentales y necesarios para crear OAs.

A continuación se describen los criterios de consideración establecidos:

- 1) **POSIBILIDAD DE GENERACIÓN Y/O INTEGRACIÓN DE LOS COMPONENTES DEL OA:** La herramienta debe permitir la implementación de los elementos que componen un OA, como ser: contenidos, actividades, autoevaluación; y/o debe permitir la integración de elementos generados con otras HAS (acorde a la definición de OA adoptada).
- 2) **POSIBILIDAD DE CARGA DE METADATOS:** La herramienta debe permitir la carga de metadatos respetando estándares, o en su defecto debe permitir la carga de propiedades que describan al OA y que sean consideradas (almacenadas) como los metadatos del mismo (acorde a la definición de OA adoptada, los metadatos permitirán la reutilización del OA a partir de su almacenamiento, búsqueda y recuperación).
- 3) **POSIBILIDAD DE EMPAQUETAMIENTO ESTANDARIZADO:** La herramienta debe permitir la generación de paquetes de contenidos educativos respetando modelos de empaquetamiento estandarizados (acorde a la definición de OA adoptada, un OA debe respetar estándares de empaquetamiento para su interoperabilidad).

3.2 Criterios de análisis

Los criterios de análisis establecidos en este trabajo intentan ayudar a los docentes al momento de decidir qué HA utilizar para la creación de OAs, ya que reflejan aspectos tecnológicos fundamentales de las herramientas.

A continuación se describen los criterios de análisis establecidos:

- 1) **METADATOS:** Se refiere a cuáles son los estándares que contempla la herramienta para la carga de metadatos. *Dublin Core*¹, *IMS MD*², *LOM*³, *LOM-ES*⁴, entre otros.
- 2) **EMPAQUETAMIENTO:** Se refiere a los estándares de empaquetamiento que contempla la herramienta. *IMS CP*⁵, *SCORM*⁶, entre otros.
- 3) **LICENCIA:** Se refiere a la licencia de software que posee la herramienta *GPL*⁷, *GPL-2*⁸, *MIT Open Source*⁹, entre otras.

¹ *Dublin Core*: <http://dublincore.org/>.

² *IMS MD*: <http://www.imsglobal.org/metadata/index.html>.

³ *LOM*: <https://standards.ieee.org/findstds/standard/1484.12.1-2002.html>.

⁴ *LOM-ES*: <http://www.lom-es.es/>.

⁵ *IMS-CP*: <https://www.imsglobal.org/content/packaging/index.html>.

⁶ *SCORM*: <https://www.adlnet.gov/adl-research/scorm/>.

⁷ *GPL*: <https://www.gnu.org/licenses/licenses.html#GPL>.

⁸ *GPL-2*: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.html>.

⁹ *MIT Open Source*: <https://opensource.org/licenses/MIT>.

- 4) PLATAFORMA: Se refiere a las plataformas sobre las cuales es posible ejecutar la herramienta. *Linux*¹⁰, *Mac OS X*¹¹, *Windows*¹², entre otras.

3.3 Proceso de selección

Para realizar la selección de las HAs para la creación de OAs, a ser posteriormente analizadas de manera comparativa, se partió de un conjunto de 40 HAs. Se recopiló información acerca de las mismas, se revisaron sus características principales y sus especificaciones técnicas y se probó cómo funcionan varias de ellas.

El proceso de selección fue realizado en dos etapas. La primera etapa consistió en aplicar sobre el conjunto de HAs los criterios de consideración establecidos anteriormente, lo que determinó un sub-conjunto de 7 HAs para la creación de OAs. La segunda etapa consistió en distinguir dentro de dicho sub-conjunto aquellas herramientas: (1) de mayor uso en el ámbito educativo, (2) citadas en varios trabajos de investigación y artículos científicos [3] [18] [20] [21] y (3) de las cuales se encontró información más específica. Esta segunda etapa arrojó como resultado 4 herramientas, las cuales quedaron seleccionadas para el análisis comparativo. Se aclara que ninguna de las 4 herramientas seleccionadas sigue una Metodología de diseño de OAs.

A continuación se describen brevemente las herramientas seleccionadas:

- *Ardora*: Es una aplicación informática para docentes que les permite crear sus propios contenidos web de un modo muy sencillo, sin tener conocimientos técnicos de diseño o programación web. Permite crear distintos tipos de actividades y páginas multimedia [22].
- *eXeLearning*: Es una HA que tiene como objetivo ayudar a los docentes en la creación y publicación de contenidos. Permite generar sitios web completos, insertar contenidos interactivos (preguntas y actividades de diferentes tipos) en cada página [23].
- *RELOAD Editor*: Es un empaquetador de contenidos y un editor de metadatos. Permite generar contenido electrónico (páginas web, imágenes, animaciones *flash*, *Java applets*, etc.), y empaquetarlo y describirlo para su almacenamiento en repositorios de contenido [24].
- *Xerte*: Es un entorno de desarrollo completo orientado al *E-Learning*. Va dirigido a elaboradores/productores de contenido interactivo que quieran crear contenido específico, y más elaborado haciendo uso de algunas secuencias de comandos [25].

No obstante, vale destacar que se han observado HAs que no cumplan con todos los criterios de consideración establecidos en este trabajo y sin embargo también podrían servir para la creación de OAs, si se tienen en cuenta ciertas salvedades.

A continuación se listan herramientas que podrían ser utilizadas para crear OAs:

- *GLO Maker* [26]: Esta herramienta permite generar los componentes del OA, sin embargo, no cuenta con ninguna opción para la carga de metadatos y la

¹⁰ *Linux*: <https://www.linux.com/>.

¹¹ *Mac OS X*: <http://www.apple.com/la/osx/>.

¹² *Windows*: <https://www.microsoft.com/es-ar/windows>.

exportación del OA genera un grupo de carpetas y archivos entre los cuales se encuentra un archivo HTML a partir del cual se accede al OA [3].

- *XML SCORM Studio* [27]: Esta herramienta permite generar los componentes del OA, pero no permite la carga de metadatos, aunque genera de manera automática un conjunto de ellos al exportar a SCORM [3].

La primera herramienta mencionada no cumple con los criterios de consideración referentes a los metadatos y al empaquetamiento estandarizado. La segunda herramienta no cumple con el criterio de consideración referente a los metadatos de manera estricta, ya que no permite cargar metadatos (o propiedades descriptivas) durante el proceso de creación de OAs. Esta falencia podría ser superada si se almacenan/editan los metadatos posteriormente (con otra HA que lo permita, por ejemplo: *RELOAD Editor*), ya que la herramienta en cuestión crea el archivo de metadatos correspondiente cuando se empaqueta el OA bajo el modelo SCORM.

3.4 Análisis comparativo

El análisis comparativo, sobre las HAs para la creación de OAs seleccionadas, tiene el foco puesto en los criterios de análisis establecidos anteriormente. En la figura 1 se presenta un cuadro que muestra el análisis comparativo realizado.

		CRITERIOS DE ANÁLISIS			
		METADATOS	EMPAQUETAMIENTO	LICENCIA	PLATAFORMA
HAs PARA LA CREACIÓN DE OAs	<i>Ardora</i>	Carga de propiedades, que serán metadatos.	<i>SCORM</i>	Gratuita, pero no de código abierto.	<i>Linux Windows</i>
	<i>eXeLearning</i>	<i>Dublin Core LOM LOM-ES</i>	<i>IMS CP SCORM</i>	Libre, bajo licencia: <i>GPL-2</i> .	<i>Linux Mac OS X Windows</i>
	<i>RELOAD Editor</i>	<i>IMS MD LOM</i>	<i>IMS CP SCORM</i>	Libre, bajo licencia: <i>MIT Open Source</i> .	Multiplataforma (<i>Java</i>).
	<i>Xerte</i>	Carga de propiedades, que serán metadatos.	<i>SCORM</i>	Libre, bajo licencia: <i>GPL</i> .	<i>Windows</i>

Fig. 1. Análisis comparativo de HAs para la creación de OAs.

4 Resultados preliminares

Los resultados preliminares, obtenidos a partir del análisis comparativo realizado, permiten reflexionar sobre las posibilidades actuales de determinadas HAs para la creación de OAs.

A continuación se presenta una discusión sobre cada criterio de análisis:

- Resultados sobre METADATOS: 2 de las herramientas analizadas, *eXeLearning* y *RELOAD Editor*, permiten realizar la carga de metadatos contemplando estándares como: *Dublin Core*, *IMS MD*, *LOM* y *LOM-ES*. Las otras 2 herramientas, *Ardora* y *Xerte*, permiten la carga de “propiedades” que luego serán consideradas como metadatos.
- Resultados sobre EMPAQUETAMIENTO: Las 4 herramientas analizadas coinciden en contemplar el modelo de empaquetamiento *SCORM*. Cabe aclarar que *Ardora* sólo permite empaquetamiento estandarizado de *SCORM* para actividades y paquetes de actividades. 2 de las herramientas, *eXeLearning* y *RELOAD Editor*, además contemplan el modelo de empaquetamiento estandarizado de *IMS CP*.
- Resultados sobre LICENCIA: 3 de las herramientas analizadas son libres y de código abierto, *eXeLearning*, *RELOAD Editor* y *Xerte*. Y 1 de ellas, *Ardora*, es gratuita para uso personal, sin carácter lucrativo y con fines estrictamente educativos, y no está permitida su descompilación en todo o en parte.
- Resultados sobre PLATAFORMA: Las 4 herramientas analizadas coinciden en tener a *Windows* como plataforma de ejecución posible. 3 de las herramientas, *Ardora*, *eXeLearning* y *RELOAD Editor*, también pueden ser ejecutadas en *Linux*. Y 2 de ellas, *eXeLearning* y *RELOAD Editor*, además son ejecutables en *Mac*. Cabe aclarar que *RELOAD Editor* es una herramienta multiplataforma desarrollada en *Java*.

5 Conclusiones y Trabajo futuro

Luego de la investigación realizada y habiendo analizado diversas HAs, se presentan a continuación conclusiones a las cuales se ha arribado.

Se observó que las HAs que permiten generar materiales educativos frecuentemente cuentan con una interfaz amigable y su manejo es intuitivo. Proporcionan a los docentes distintos tipos de plantillas de contenidos y/o actividades. Pueden ser de propósito general o específico dependiendo de si se enfocan o no en un tema en particular.

Con respecto a las HAs para la creación de OAs, se detectó que son aquellas que permiten la generación de OAs, ya sea posibilitando la implementación de sus componentes (contenidos, actividades, autoevaluación) o su importación, además permiten la carga de metadatos y respetan estándares de empaquetamiento. A su vez, entre estas herramientas, se distinguieron dos tipos: las herramientas que no contemplan una Metodología de diseño de OAs y las herramientas que si lo hacen.

Se notó que aquellas HAS para la creación de OAs que siguen una Metodología de diseño de OAs son más completas y útiles para los docentes, ya que consideran todas las etapas que requiere crear OAs y acompañan en cada una de las tareas mediante guías de diseño e implementación. Este tipo de herramientas no abundan en el mercado y, por ende, son las más difíciles de encontrar. Las herramientas disponibles, en su mayoría, se centran en la implementación de OAs y asisten al docente con más o menos detalle en dicha etapa, pero no contemplan las etapas previas que requiere la creación de OAs.

El proceso de selección de HAS para la creación de OAs que se hizo, con el objetivo de realizar luego un análisis comparativo sobre determinadas herramientas, dejó ver que actualmente son pocas las HAS que realmente permiten crear OAs. Existen herramientas que sí lo permiten, pero la mayoría no suele considerar todos los aspectos fundamentales que hacen a la creación de un OA. Se encuentran, por ejemplo, herramientas que contemplan la exportación de paquetes respetando estándares pero no permiten la carga de metadatos, y viceversa. Los docentes, en particular los que no están acostumbrados al uso de la tecnología para generar materiales educativos, deben lidiar con estas cuestiones al momento de elegir una HA que les permita la creación de OAs. Es por ello que los criterios de consideración y análisis establecidos permiten guiar a los docentes en la selección de HAS para crear OAs, teniendo en cuenta los aspectos fundamentales para este tipo de MEDs.

Debido a la falta que se vislumbra de HAS para la creación de OAs, en particular basadas en Metodologías de diseño de OAs, se espera como trabajo futuro desarrollar un *framework* para la creación de OAs que además de permitir la implementación de OAs guíe a los docentes en el diseño de los mismos.

Referencias

1. Fernandez Pampillón, A. M.: Desarrollo de una Norma Española de Calidad de Materiales Educativos Digitales. Revista IEEE-RITA, vol. 2 (1), pp. 49--56, ISSN 2255-5706 (2011)
2. Maldonado, J., Astudillo, G.: Los Objetos de Aprendizaje: Un estado del arte en Iberoamérica. VI Conferencia Iberoamericana sobre Tecnologías y Aprendizaje, Miami, Estados Unidos (2014)
3. Astudillo, G. J.: Análisis del estado del arte de los objetos de aprendizaje. Revisión de su definición y sus posibilidades. Trabajo final de Especialización, Facultad de Informática, Universidad Nacional de La Plata (2011)
4. Metodología CROA de la Universidad Nacional de La Plata, <http://croa.info.unlp.edu.ar/>
5. Parra Castrillón, E.: Propuesta de metodología de desarrollo de software para objetos virtuales de aprendizaje-MESOVA. Revista Virtual Universidad Católica del Norte, vol. 1 (34), pp. 113--137 (2011)
6. Baruque, L. B., Melo, R. N.: Learning theory and instructional design using learning objects. Journal of Educational Multimedia and Hypermedia, vol. 13 (4), pp. 343 (2004)
7. Sanz, C., Barranquero, F., Moralejo, L.: CROA: A learning object design and creation methodology to bridge the gap between educators and reusable educational material creation. EDULEARN16 Proceedings, pp. 4583--4592 (2016)
8. Borrero M., Cruz E., Mayorga S., Ramírez K.: Una metodología para el diseño de objetos de aprendizaje. La experiencia de la Dirección de Nuevas Tecnologías y Educación Virtual, DINTEV. Universidad del Valle, Dirección de Nuevas Tecnologías y Educación Virtual, Cali, Colombia (2010)

9. Hernández Y., Silva A.: Una Experiencia Tecnopedagógica en la Construcción de Objetos de Aprendizaje Web para la Enseñanza de la Matemática Básica. *Revista de Tecnología de Información y Comunicación en Educación Eduweb*, vol. 5 (1), ISSN 1856-7576 (2011)
10. Villa, M. G., Morales, A. M. F., Martínez, J., Valenzuela, P. P. M., Saorín, S. M., Marín, A. R. C., Del Baño, J. V.: Herramientas de Autor e integración curricular: “Las Aventuras de Topy”, una aplicación multimedia para el desarrollo de la Comunicación Alternativa y Aumentativa en el aula. *Actas del II Congreso Nacional de Nuevas Tecnologías y Necesidades Educativas Especiales*, Murcia, España (2002)
11. Moralejo, M. L.: Análisis comparativo de Herramientas de Autor para la creación de actividades de Realidad Aumentada. Estudio de sus características específicas para el escenario educativo. Trabajo final de Especialización, Facultad de Informática, Universidad Nacional de La Plata (2014)
12. Dabbagh, N.: Authoring Tools and Learning Systems: A Historical Perspective. *Annual Proceedings-Atlanta*, vol. 87 (2001)
13. Constructor, <https://constructor.educarex.es/>
14. Jelic, <http://clic.xtec.cat/es/jclic/index.htm>
15. Malted, <http://malted.cnice.mec.es/presentacion/QMalted.htm>
16. Rayuela, <http://cvc.cervantes.es/aula/pasatiempos/>
17. Menéndez, V. H., Prieto, M. E., Zapata, A.: Sistemas de Gestión Integral de Objetos de Aprendizaje. *Revista IEEE-RITA*, vol. 5 (2), pp. 56--62 (2010)
18. Diwakar, A., Patwardhan, M., Murthy, S.: Pedagogical analysis of content authoring tools for engineering curriculum. In *Technology for Education (T4E)*, 2012 IEEE Fourth International Conference on, pp. 83--89 (2012)
19. Nieves-Guerrero, C. G., Menéndez-Domínguez, V. H., Gómez, O. S.: Estudio Comparativo de Herramientas de Apoyo a la Creación de Objetos de Aprendizaje. *Revista IEEE RITA*, vol. 2 (3), pp. 101--108 (2014)
20. Ojea, A. M.: Free software tools for the authoring of SCORM learning objects. In *Int. Conf. Technol. Train. Commun.*, vol. 361 (2007)
21. Astudillo, G., Sanz, C. & Willging, P.: Análisis de Compatibilidad entre Objetos de Aprendizaje basados en SCORM y LMS de Código Abierto. *Conferencias LACLO*, vol. 3 (1) (2012)
22. Ardora, <http://webardora.net/>
23. eXeLearning, <http://exelearning.net/>
24. RELOAD Editor, <http://www.reload.ac.uk/editor.html>
25. Xerte, <http://www.nottingham.ac.uk/xerte/>
26. GLO Maker, <http://www.glomaker.org/>
27. Xml SCORM Studio, <http://xmlscormstudio.sourceforge.net/>

Modeling Students through Analysis of Social Networks Topics

María Emilia Charnelli¹, Laura Lanzarini², Javier Díaz¹

¹LINTI - Laboratory of Research in New Information Technologies

²III LIDI - Institute of Research in Computer Science LIDI
Computer Science School, National University of La Plata

mcharnelli@linti.unlp.edu.ar, laural@lidi.info.unlp.edu.ar,
jdiaz@unlp.edu.ar

Abstract. Educational Data Mining gathers the multiple methods that allow new and useful information extraction from great volumes of data coming from the educational context. The goal of this article is to obtain a model of the students of the Computer Science School of the UNLP from their participation in Facebook. The work describes the process of extraction of latent topics in posts made in public groups related to the School, and the modeling of the students from the topics discovered. Additionally, it includes the preprocessing done to the collected data, which constitutes a fundamental stage since it strongly conditions the performance of the models to be obtained. Finally, obtained results are presented together with conclusions and future lines of work.

Keywords: Educational Data Mining, Learning Analytics, Topic Modeling, User Modeling, Recommender Systems.

1 Introduction

In the last few years, educational institutions have embarked on their own exploration of big data sets to increase retention rates and provide students with a customized and higher quality experience. Applying data mining techniques in education has allowed to characterize the actors involved in teaching and learning processes. Generally speaking, it is very difficult to exploit available information to create models that describe students objectively. In particular, extracting information from the web constitutes a significant challenge due to the unstructured nature of the data it contains. Social networks make up an environment external to the educational institution, however containing valuable information regarding the interests of the students. Current research focuses on methods for extracting implicit information on student behaviors from social platforms in order to obtain dynamic models that are capable of easily adapting to changes in information and contributing to decision-making processes in education.

Analyzing latent topics has emerged as one of the most efficient methods to classify, group and retrieve textual data, such as those found in social network

posts. Many latent topic modeling methods have been developed and studied extensively, such as PLSA [1] and LDA [2]. In these models, documents are modeled as mixtures of topics, where a topic is a probability distribution of all the possible words in the documents. Statistical techniques are used to learn the topics and the mixing coefficients for each document. Conventional topic models reveal latent topics by discovering word co-occurrence patterns in all documents [3] [4]. Applying traditional topic modeling techniques on short texts such as tweets, Facebook status updates and instant messaging may not yield optimal results since they lack rich contexts. The main reason is that topic modeling implicitly captures word co-occurrence patterns by document in order to discover topics, therefore there is a severe data dispersion in shorter documents. More specifically, word occurrence in short texts plays a less discriminatory role in comparison to longer documents where the model has a number of words that is sufficient to know how they are related [5]. Discovering topics in short texts is crucial for a wide range of topic analyzing tasks, such as characterizing content [6] [7], modeling user interest profiles [8][9], and detecting latent or emerging topics [10]. BTM (Biterm Topic Model) [11] is an effective way to learn latent topics in short texts. BTM extracts underlying topics in a set of documents and a global distribution of each topic in each of them, through an analysis of the generation of biterms. Likewise, BTM has extensions to treat data flows and an incremental scheme for updating the model and giving more importance to the latest data collected.

This paper presents a method for obtaining a model of the students of the Computer Science School of the National University of La Plata (UNLP) through an analysis of the posts they make in public student groups in the Facebook social network.

This work is organized as follows: the second section describes the preprocessing effected on the collected data, the third section shows the extraction and modeling of latent topics, the fourth section shows the construction of student models through discovered topics, and the fifth section shows results obtained. Finally, the sixth section presents conclusions and future work.

2 Data preparation

The information used in this work comes from Facebook posts in groups created by Computer Science School students. The data were collected through the Facebook Graph API and involve over 3000 posts and 1500 students that write, comment, share and “like” posts. The Graph API is the main way of consulting and collecting data in the Facebook platform. Posts, group information and public user information were gathered. The following information was obtained from each of the posts: id of the post, creation date, author, likes, comments, shares and unstructured text content. When operating with textual information, it is necessary to use Text Mining techniques, in order to represent each post in a vector of terms. This was achieved through a process comprising many stages. The first stage consisted of the application of a stopwords filter, which

filters the words that match any indicated stopword. Stopwords were filtered in Spanish and English, using words from the social networks context such as smileys, greetings, etc; words from the context of the group such as university, school, informatics, etc; words that refer to web page addresses; words that are between symbols, etc. Likewise, students post code fragments that constitute solutions to exercises included in courses. In general, the most exercises are from the first year and written in Pascal. Therefore, each sentence or signature of the language used was reduced to a single word in order to better identify them. Following, each word in the text was reduced to its root applying the Snowball [12] stemming algorithm. The importance of this process lies in that it eliminates syntactic variations related to gender, number and tense. The algorithm was applied for both the English and Spanish languages. Once the roots of each of the words were obtained, the frequency of occurrence was calculated for each of them, and words that appeared more than once were chosen.

3 Extraction of latent topics

BTM was used for extracting topics in Facebook groups, which is an unsupervised learning technique that discovers topics characterizing a set of brief documents. In this context, each post and comment is considered a document.

Let a set of N_D documents be called corpus and let W be the set of all the words in the corpus, a topic is defined as a probability distribution of W . Therefore, a topic may be characterized by its T most likely words. Given a number K of topics, the goal of BTM is to obtain the K distributions of each of the words.

A “biterm” is a pair of words that co-occur without a set order in a short document. In this case, two different words in a document constitute a biterm. Given a corpus of N_D documents and a W vocabulary of unique words, it is assumed to contain N_B $\mathbf{B} = \{b_i\}_{i=1}^{N_B}$ biterms with $b_i = (w_{i,1} \in W, w_{i,2} \in W)$, and K topics expressed of W . Let $z \in [1, K]$ be a variable to indicate a topic. The $P(z)$ probability that a document in the corpus is of a z topic is defined as a $\boldsymbol{\theta} = \{\theta_k\}_{k=1}^K$ K -dimensional multinomial distribution with $\theta_k = P(z = k)$ and $\sum_{k=1}^K \theta_k = 1$. The $P(w|z)$ distribution of words per topic can be represented as a $\Phi \in R^{K \times W}$ matrix where the ϕ_k k -th row is a W -dimensional multinomial distribution with $\phi_{k,w} = P(w|z = k)$ entry and $\sum_{w=1}^W \phi_{k,w} = 1$. Given the α and β parameters, the main assumption of the model is that each of the documents of the corpus was generated thus:

1. A $\boldsymbol{\theta} \sim \text{Dirichlet}(\alpha)$ topic distribution is chosen for all the corpus
2. For each $k \in [1, K]$ topic
 - A $\phi_k \sim \text{Dirichlet}(\beta)$ distribution of words is extracted for the topic
3. For each $b_i \in \mathbf{B}$ biterm
 - A $z_i \sim \text{Multinomial}(\boldsymbol{\theta})$ topic assignment is extracted

– Two $w_{i,1}, w_{i,2} \sim \text{Multinomial}(\phi_{z_i})$ words are extracted

Taking into account the generation mechanism assumed by BTM, likelihood for all the corpus can be obtained given parameters α and β from the probability of each of the biterms:

$$P(\mathbf{B}|\alpha, \beta) = \prod_{i=1}^{N_B} \int \int \sum_{k=1}^K P(w_{i,1}, w_{i,2}, z_i = k | \boldsymbol{\theta}, \boldsymbol{\Phi}) d\boldsymbol{\theta} d\boldsymbol{\Phi} \quad (1)$$

$$= \prod_{i=1}^{N_B} \int \int \sum_{k=1}^K \theta_k \phi_{k,w_{i,1}} \phi_{k,w_{i,2}} d\boldsymbol{\theta} d\boldsymbol{\Phi} \quad (2)$$

Obtaining exactly the $\boldsymbol{\theta}$ and $\boldsymbol{\Phi}$ parameters that maximize the likelihood of equation 2 is an untreatable problem. Following the proposals in [13], parameters $\boldsymbol{\theta}$ and $\boldsymbol{\Phi}$ can be approximated using Gibbs sampling [14].

3.1 Co-occurrence Matrix

In order to evaluate the quality of the topics obtained, the coherence metric proposed by Mimno et al. [15] is used. Given a z and its T most likely words $V^{(z)} = (v_1^{(z)}, \dots, v_T^{(z)})$ where $v_i^{(z)} \in W$ for $i = 1 \dots T$, the coherence score is defined as:

$$C(z; V^{(z)}) = \sum_{t=2}^T \sum_{l=1}^t \log \frac{D(v_t^{(z)}, v_l^{(z)}) + 1}{D(v_l^{(z)})}$$

where $D(v)$ is the frequency of word v in all documents, $D(v, v')$ is the number of documents where words v and v' co-occur. The Coherence metric is based on the idea that words that belong to one concept will tend to co-occur in the same documents. This is empirically demonstrable since the coherence score is highly correlated with human criteria. In order to evaluate the general quality of a set of topics, the $\frac{1}{k} \sum_k C(z_k; V^{(z_k)})$ average of the coherence metric is calculated for each of the topics obtained. These results allow us to determine the amount of topics that best represent the entire corpus.

4 Modeling the Students

Once the K topics representing Facebook group posts collected were obtained, the students were modeled as vectors in a K -dimensional space. Each position of the vector represents the level of participation of the student in each of the topics. The level of participation is associated with possible actions performed by a user on the contents. Possible actions are creating content, commenting or liking a post. Given n users, $X \in R^{n,K}$ is defined as the matrix that contains in its rows the representation of each user in the new characteristics space. Let $A_{l,m} = \{a_1, a_2, \dots, a_t\}$ be the set of actions of user l on posts classified in topic m . Then the l, m -th component of matrix X is defined as:

$$X[l, m] = \max \left(\sum_{i=1}^t w(a_i), 1 \right), \text{ with } l = 1 \dots n \text{ and } m = 1 \dots K$$

where $w(a_i)$ is the weight associated with action a_i . $w(a_i)$ is defined as:

$$w(a_i) = \begin{cases} \frac{1}{20} & \text{if } a_i \text{ is a new post} \\ \frac{1}{40} & \text{if } a_i \text{ is a comment} \\ \frac{1}{50} & \text{if } a_i \text{ is a "like"} \end{cases}$$

Cosine similarity is used in order to evaluate the similarity between two users in the new space of characteristics. Given $v_1, v_2 \in R^K$, the similarity function is defined as:

$$d(v_1, v_2) = \frac{v_1 v_2}{\|v_1 v_2\|} = \cos(\theta)$$

5 Results

The model obtained with BTM was evaluated in the set of Facebook group posts. For each number of topics between 2 and 45, the average of the obtained coherence was calculated sampling the test and training set randomly in 1000 iterations. Figure 1 shows the average of the coherence of the topics model according to the number of topics extracted. What is interesting is the number of topics in which there is a breaking point in the growth of the average coherence function. In this case, the optimal value is between 10 and 12 latent topics. Once the optimal number of topics was obtained, each of the Facebook group posts was classified into a topic according to the latent topics model that was obtained.

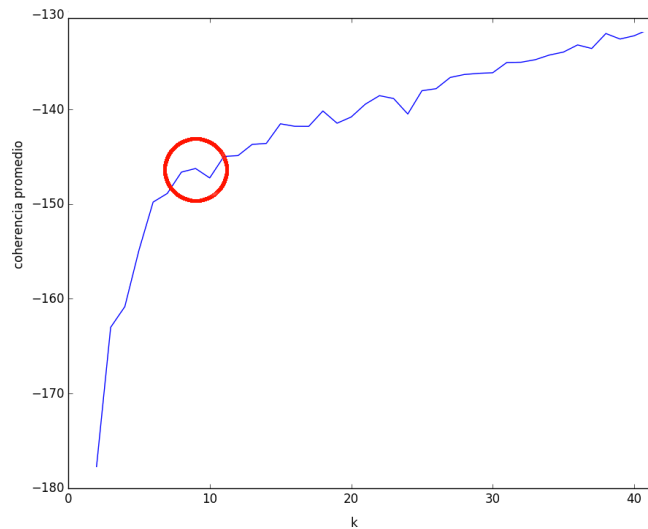


Fig. 1. Average coherence for different Ks

Table 1 shows topics obtained with $K=10$. For each of the topics, the six most important words are shown, i.e., those with the most likelihood of belonging to the topic. For example, topic 1 is about scholarships, topic 9 is about the first year programming subject where Pascal is used, and topic 10 contains posts on student accommodation search and offer.

Topic	Most important words of the topic				
1	enrollment	students	scholarships	university	national
2	support	workshops	courses	student	matters
3	community	sharing	tools	hacking	security
4	classroom	midterm	final	date	adp
5	courses	enrollment	information	page	systems
6	work	experience	knowledge	development	java
7	file	commands	linux	ubuntu	text
8	tutoring	classes	study	question	mathematics
9	pascal	close	file	enter	reset
10	students	double	simple	rooms	foreigners

Table 1. Model of the topics obtained with BTM

Afterwards, the actions performed by the users on each of the contents classified in the 10 topics were obtained which allowed for the generation of the users model described in section 4. Unsupervised learning techniques were applied to determine the underlying structure of student groups. In order to obtain the optimal number of user clusters automatically, the Silhouette [16] index was used, finding the value that optimized the criterion. Figure 2 shows the result of the evaluation of the model, generated by a hierarchical clustering algorithm with the average linkage criterion. It can be observed that the optimal value is implicitly imposed by the amount of topics obtained with BTM. In this context, it is interesting to model with greater detail the interests of the users. The following criterion-optimizing value is found in $k = 42$.

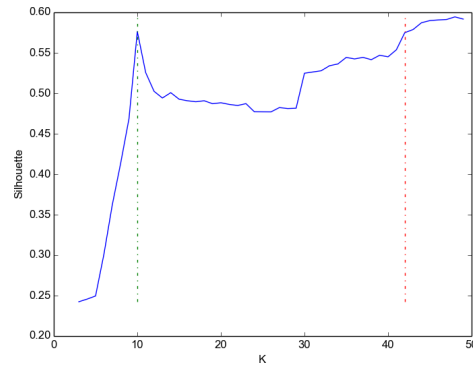


Fig. 2. Silhouette index on hierarchical clustering with average linkage

Figure 3 shows the dendrogram obtained by means of a hierarchical algorithm using the average linkage criterion with the cosine similarity metric, showing the obtained groupings and the correspondence with the cluster validity criterion.

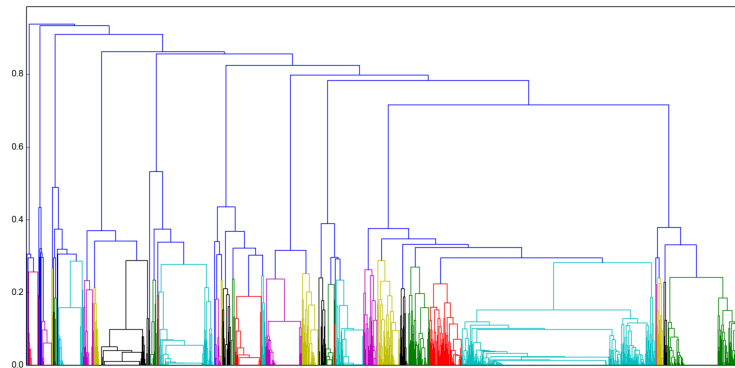


Fig. 3. Dendrogram obtained using average linkage and cosine distance

6 Conclusions and Future Work

This article presents a method for modeling the students of the Computer Science School of the UNLP through the detection of latent topics in their posts in public Facebook groups. This allows characterizing the students from a different context, knowing their topics of interest and how they relate to one another.

The methodology and the validation metrics used to obtain the user model presents satisfactory preliminary results. Topic coherence allows for automatic detection of the optimal number of topics and the user model presents compact groupings that characterize the behavior of the students in Facebook groups.

One of the future work points contemplates the implementation of an incremental model that allows updating the latent topics and therefore, the user modeling, from new posts in social networks.

The results of this work join those in [17] [18], which identifies features better characterizing students regarding their academic level through personal and academic information provided by the student management system. Results obtained will allow for the creation of an initial recommender system for the educational environment.

References

1. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM (1999) 50–57
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *Journal of machine Learning research* **3** (2003) 993–1022
3. Boyd-Graber, J.L., Blei, D.M.: Syntactic topic models. In: *Advances in neural information processing systems*. (2009) 185–192
4. Wang, X., McCallum, A.: Topics over time: a non-markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2006) 424–433
5. Hong, L., Davison, B.D.: Empirical study of topic modeling in twitter. In: Proceedings of the first workshop on social media analytics, ACM (2010) 80–88
6. Zhao, W.X., Jiang, J., Weng, J., He, J., Lim, E.P., Yan, H., Li, X.: Comparing twitter and traditional media using topic models. In: *European Conference on Information Retrieval*, Springer (2011) 338–349
7. Guo, J., Xu, G., Cheng, X., Li, H.: Named entity recognition in query. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, ACM (2009) 267–274
8. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the third ACM international conference on Web search and data mining, ACM (2010) 261–270
9. Ramage, D., Dumais, S.T., Liebling, D.J.: Characterizing microblogs with topic models. *International Conference on Weblogs and Social Media* **5** (2010) 130–137
10. Lin, C.X., Zhao, B., Mei, Q., Han, J.: Pet: a statistical model for popular events tracking in social communities. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2010) 929–938
11. Cheng, X., Yan, X., Lan, Y., Guo, J.: Btm: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering* **26** (2014) 2928–2941
12. Gupta, V., Lehal, G.S.: A survey of common stemming techniques and existing stemmers for indian languages. *Journal of Emerging Technologies in Web Intelligence* **5** (2013) 157–161
13. Griffiths, T.L., Steyvers, M.: Finding scientific topics. *Proceedings of the National academy of Sciences* **101** (2004) 5228–5235

14. Geman, S., Geman, D.: Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence* (1984) 721–741
15. Mimno, D., Wallach, H.M., Talley, E., Leenders, M., McCallum, A.: Optimizing semantic coherence in topic models. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics (2011) 262–272
16. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* **20** (1987) 53–65
17. Lanzarini, L., Charnelli, M.E., Baldino, G., Diaz, J.: Seleccin de atributos representativos del avance academico de los alumnos universitarios usando tecnicas de visualizacion: Un caso de estudio. *Revista TE&ET* (2015) 42–50
18. Lanzarini, L., Charnelli, M.E., Diaz, J.: Academic performance of university students and its relation with employment. In: *Computing Conference CLEI, 2015 Latin American*. (2015) 1–6

Metodologías de Ensamblaje de Objetos de Aprendizaje. Análisis bajo la lupa del concepto de Objetos de Aprendizaje subyacente.

Astudillo Gustavo J.¹, Sanz Cecilia V.² y Santacruz-Valencia Liliana P.³,
¹GrIDIE. Dpto. de Matemática, FCEyN, UNLPam, Av. Uruguay 151, La Pampa, Argentina, astudillo@exactas.unlpam.edu.ar,
²III LIDI, Facultad de Informática, UNLP, Calle 50 y 120, La Plata, Argentina, csanz@lidi.info.unlp.edu.ar,
³Dpto. de Lenguajes y Sistemas Informáticos I, Universidad Rey Juan Carlos, c/Tulipán s/n, Móstoles, Madrid, España, liliana.santacruz@urjc.es

Resumen. En este trabajo se presenta un análisis sobre Metodologías de Ensamblaje de Objetos de Aprendizaje (OA), con foco en la definición adoptada por cada una en relación a lo que es un OA. La investigación forma parte de un estudio más abarcativo sobre este tipo de metodologías, sin embargo se discute aquí la incidencia que tiene el no contar con una definición acordada de OA para llevar adelante el proceso de ensamblaje. Para ello se realiza un análisis detallado de 33 metodologías seleccionadas a partir de una revisión bibliográfica. Los resultados señalan la necesidad de explicitar una definición de OA previo al trabajo con una de tales metodologías y atender a las principales características de los OA acordadas por la comunidad académica.

Palabras clave: Metodologías de ensamblaje, Objetos de aprendizaje, Definición.

1 Introducción

El aprendizaje es un proceso que necesita de la secuenciación del contenido –y de las estrategias que lo vehiculizan–, y es precisamente, en este aspecto del proceso educativo en el que se enfoca el ensamblaje.

Las Metodologías de Ensamblaje (ME) tienen como objetivo principal la definición de un itinerario de aprendizaje con base en un conjunto de Materiales Educativos Digitales (MED) alojados en repositorios. El proceso involucra tres etapas: definición de la temática, definición de la secuencia de aprendizaje y agregación de los MED apropiados.

La presente investigación está enmarcada en un trabajo de tesis de maestría [1]. En el mismo se llevó adelante un proceso de búsqueda y selección de ME de Objetos de Aprendizaje (MEOA). Se analizaron 33 MEOA, lo que permitió la definición de criterios de evaluación.

El análisis realizado, entre otros resultados [1], permitió corroborar que los autores hablan sobre la utilización de OA por parte de las ME para crear los itinerarios de

aprendizaje. Sin embargo, la conceptualización y caracterización de los OA difiere entre las metodologías. Esto obliga a realizar un análisis centrado en la definición y caracterización que cada ME, hace previo a la elección de la misma y/o del Sistema Ensamblador (SE) que la implementa, con vistas a su utilización. En este sentido, el presente trabajo realiza un aporte que facilita la toma de decisiones, ya que expone y clasifica las definiciones utilizadas, hace uso de criterios de evaluación que permiten caracterizar el tipo de material educativo con el que trabajan las MEOA y discute la importancia de la adopción de una definición apropiada.

En adelante este trabajo se organiza de la siguiente manera. En la sección 2 se presentan algunos antecedentes en relación al concepto de Metodología de Ensamblaje, Sistemas Ensambladores y MED utilizados para el ensamblaje. En la sección 3, se analizan las diferentes definiciones de OA abordadas en las ME bajo estudio; en la sección 4 se presentan los criterios con los que se han analizados las MEOA, y en particular se pone el foco en el criterio referido a la conceptualización de OA, que se lo aplica para realizar un análisis sobre este aspecto. Finalmente, en la sección 5 y en la 6, se discute el impacto que tiene no contar con una definición única de OA en las Metodologías de Ensamblaje y se presentan conclusiones al respecto.

2 Objetos de Aprendizaje y Metodologías de Ensamblaje

Una metodología de ensamblaje busca establecer una correspondencia sintáctica, estructural y semántica entre materiales educativos [2]. Así es posible generar un itinerario de aprendizaje en base a MED alojados en repositorios. Esta acción puede ser llevada a cabo con distintos niveles de automatización y puede tener (o no) en cuenta el perfil de usuario [1].

Los SE deben hacer uso de los MED disponibles en almacenes locales o remotos. El material debe cumplir con un conjunto de requisitos que permita utilizarlo en el contexto de estos sistemas. Básicamente, deben:

- contar con información que permita relacionarlo con otro
- contar con información que lo describa (técnica y pedagógicamente)
- estar alojado en repositorios

Como se afirma en [3] “los [OA] son vistos como la tecnología del futuro [...] debido a la capacidad de adaptación, reutilización y escalabilidad potencial que ofrecen”. Estas ventajas que ofrece el paradigma de OA están sustentadas en un tipo de material educativo que desde sus inicios propone el ensamblaje, así como el etiquetado por medio de metadatos y el diseño con vistas a la reutilización [4, 5].

En pos de maximizar la reutilización del material (característica distintiva de los OA): los metadatos deben apegarse a un estándar y los OA deben ser alojados en repositorios. Por tanto, los OA cumplen, al menos en teoría, con los requerimientos impuestos por los SE al material educativo con el que se crean los itinerarios de aprendizaje.

3 Definiciones de OA Utilizadas por las MEOA

Para un análisis preliminar, es posible focalizarse en las definiciones adoptadas por las MEOA. Cabe aclarar que sólo 20 de las 33 metodologías analizadas definen el concepto de OA, el resto lo utilizan sin proponer una definición.

El conjunto de publicaciones analizadas permite dar cuenta de la variedad y disparidad entre las definiciones de OA adoptadas por los distintos autores.

Del total de publicaciones que proponen una definición de OA, algunas (8 de 20) optan por una conceptualización genérica en consonancia con la propuesta por David Wiley [6]. Así un OA es definido como:

- “cualquier documento digital que se puede utilizar para el aprendizaje [...] descrito por metadatos” [7].
- “cualquier pedazo de información (texto, imagen, sonido, etc) que se puede identificar unívocamente” [8].
- “contenido crudo (raw content) u objetos de aprendizaje atómicos (texto, figuras, resumen de texto, palabras clave de texto)” [9].
- “un elemento con documentos digitales reciclables de contenido multimedia, que tienen un propósito y algún uso en la enseñanza y el aprendizaje, a la vez que cumplen con cierta especificación tecnológica.” [10].
- “pequeños componentes curriculares que puedan ser reutilizados varias veces en diferentes contextos de aprendizaje.” [11].
- “un recurso educativo atómico y auto-contenido, que está identificado unívocamente y es direccionable por un URI.” [12].
- “elemento instruccional basado en el paradigma [OOP...], orientado al soporte del aprendizaje en línea, que se crea una sola vez y se puede utilizar muchas veces más y en distintos contextos.” [13].
- “una entidad, digital o no digital, que puede ser usada para aprendizaje, educación o entrenamiento.” [14].

Las siguientes definiciones (7 de 20) se caracterizan por atribuir a los OA, entre otras propiedades, la capacidad de poder combinarse o ensamblarse. Así, los OA son:

- “una unidad de contenido de aprendizaje con un objetivo específico, a menudo utilizada como bloque de construcción en diseño y ensamblaje de cursos. Los OA forman colecciones, son etiquetados con metadatos y almacenados en repositorios.” [15].
- “la unidad de ensamblaje en entornos de enseñanza y aprendizaje más grandes construidos a partir de unidades más pequeñas.” [16].
- “recursos educativos descritos por meta-datos y organizados en una estructura multicapa, de tal forma que los elementos más avanzados de esta estructura incluyen conocimiento asociado (CA) con vistas al ensamblaje y la reutilización.” [2].
- “recursos educativos pequeños y fácilmente reutilizables que pueden componerse para permitir una instrucción personalizada y la creación de cursos.” [17].
- “materiales de aprendizaje auto-contenido que una vez desarrollados, puede ser intercambiados, compuestos, y pueden volver a utilizarse.” [18].

- “...los OA son estructurados, es decir, compuestos por una o más secuencias alternativas de otros componentes, hasta llegar a OA atómicos.” [19].
- “unidades reutilizables de contenido educativo que pueden ser secuenciados en unidades más grandes para permitir el aprendizaje personalizado.” [20].

En las siguientes cinco definiciones, se puede destacar que los autores retoman algunas de las características básicas de los OA acordadas por autores de referencia:

- “unidades mínima de información reutilizables [...que combinan] tres elementos: contenidos, descripciones del comportamiento del objeto, y un conjunto de metadatos que se refieren a los objetos.” [21].
- “elementos de granularidad fina para la transferencia de conocimiento [...] que] permiten a los ingenieros de contenido diseñar unidades didácticas modulares auto-contenidas, re-compuestas en nuevos cursos” [22].
- “material pedagógico, accesibles desde la web, que pueden ser diseñado para su uso y reutilización en una variedad de contextos, desde la remediación hasta la preparación de una lección” [23].
- “fragmentos o partes de un curso que puede variar en tamaño y complejidad desde un gráfico sencillo de todo el curso”. Además deben ser: auto-contenido, independiente, reutilizable, de baja granularidad, que pueda ser combinado, personalizado y etiquetado con metadatos [3].
- "Un OA se entenderá como todo el material estructurado de una manera significativa, y que debe estar relacionado con un objetivo de aprendizaje que debe corresponder a un recurso digital que puede ser distribuido y consultado en línea. Un OA también debe tener [...] metadatos que incluya una lista de atributos que no sólo describen los posibles atributos de un OA, sino que también permiten catalogarlo y distribuirlo" [24].

4 Criterios de Evaluación para el Análisis de MEOA

Con base en la revisión de las publicaciones seleccionadas, se pudieron identificar distintos focos de análisis de las MEOA y se generaron 17 criterios (ver Fig. 1). Los mismo permiten tanto, clasificar las diferentes MEOA, como seleccionar un SE en función de cómo se desea utilizar y del material con el que se cuenta [1, 25].

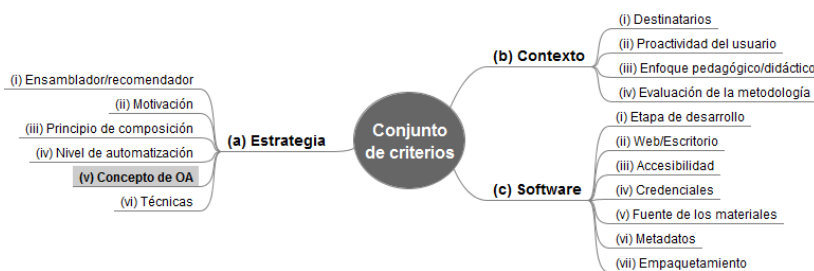


Fig. 1. Lista completa de criterios

Si bien las publicaciones seleccionadas usan OA, no los caracterizan y/o definen de la misma forma. Esto se evidencia, en esta sección, con base en los criterios definidos.

4.1 Descripción de los Criterios

De los criterios (Fig. 1), este trabajo se centra en el *a.v Concepto de OA*. Este criterio permite evaluar cuáles de las características, que mayor acuerdo tienen en la comunidad especializada [26], están presentes (o no) en el concepto de OA que adopta cada ME. Para ello, se revisó tanto la definición de OA utilizada, como la caracterización que hacen del material con que trabajan.

Cabe aclarar que se agrega la característica “Ensamblable”, que es un dato de valor para el contexto de este trabajo. También se registra cuando “No definen” el concepto de OA (esto no invalida la identificación de algunas de las demás características).

Todas las MEOA trabajan con material digital y, por ende, esta característica está presente en todas las publicaciones analizadas y no se incluye como etiqueta.

Las etiquetas utilizadas para este criterio son¹:

- Intencionalidad pedagógica
- Estructura interna
- Metadatos
- Reutilizable
- Autocontenido
- Interoperable
- Accesible
- Granularidad
- Ensamblable
- No lo definen

4.2 Aplicación del Criterio

Al aplicar *a.v Concepto de OA* (Fig. 2) puede afirmarse que no hay unanimidad en la conceptualización de OA. Además, aunque no hay una definición aceptada mayoritariamente por los expertos, 13 de 33 ME usan el concepto, pero no lo definen.

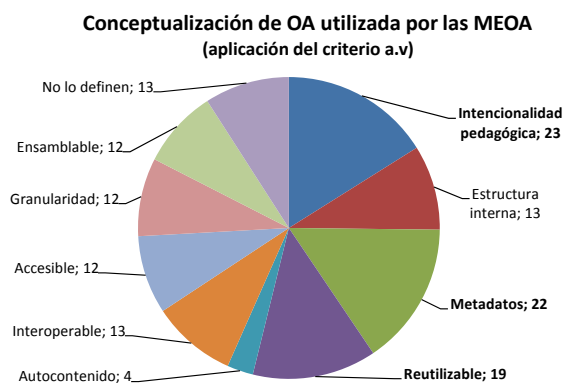


Fig. 2. Aplicación del criterios *a.v Concepto de OA*

¹ Por cuestiones de espacio no se describen las etiquetas, se sugiere revisar [1].

Es posible identificar tres características que son utilizadas mayoritariamente (Tabla 1): intencionalidad pedagógica (23 de 33), metadatos (22 de 33) y reutilizable (19 de 33). En un segundo grupo se adoptan las características interoperable (13 de 33), ensamblable (12 de 33), accesible (12 de 33), también, definen la estructura del OA (13 de 33) y abordan la granularidad (12 de 33). Sólo 4 de 27 exigen que se material auto-contenido.

Tabla 1. Caracterización de los OA propuestas en las ME analizadas.

ID*	Intencionalidad pedagógica	Estructura interna	Metadatos	Reutilizable	Autocontenido	Interoperable	Accesible	Granularidad	Ensamblable	No lo definen
S01	✓	✓	✓	✓				✓		
S03	✓		✓	✓					✓	
S04	✓			✓						
S05	✓		✓	✓		✓			✓	
S06	✓	✓	✓	✓		✓		✓	✓	
S07	✓		✓	✓			✓		✓	
S08	✓	✓	✓	✓	✓	✓		✓	✓	
S13	✓	✓	✓			✓			✓	
S14			✓					✓	✓	
S15	✓		✓	✓			✓			✓
S16	✓	✓		✓				✓		✓
S17	✓			✓					✓	
S18	✓					✓				✓
S20	✓	✓	✓	✓		✓				✓
S21	✓	✓	✓			✓		✓	✓	
S22			✓	✓		✓	✓	✓		
S23	✓	✓	✓	✓	✓	✓	✓	✓	✓	
S24	✓	✓		✓		✓				
S25							✓			✓
S27	✓	✓	✓				✓			
S28			✓	✓						✓
S30		✓	✓				✓	✓		✓
S31			✓			✓	✓			✓
S32	✓		✓	✓	✓		✓	✓		
S35	✓		✓	✓		✓	✓	✓		
S38	✓	✓	✓	✓	✓			✓	✓	
S39										✓
S40	✓									
S41						✓	✓			✓
S42	✓									✓
S43			✓							✓
S44				✓					✓	✓
S45	✓	✓	✓	✓						

* Pueden identificarse los autores en la Tabla 4.1, Cap. 4 en [1].

Por otra parte, el análisis permitió identificar qué metodologías le exigen al material con el que trabajan que tenga la capacidad de ensamblarse. Como puede verse en la Tabla 1, menos de la mitad (12 de 33) de las metodologías exigen esta característica en los OA. Tres de ellas (S03, S07 y S14) incluyen en la definición la idea de bloque de construcción, mientras que el resto (S05, S06, S08, S13, S17, S21, S23, S38, S44) le asocian la capacidad de ensamblarse con otros OA y formar MED de granularidad mayor.

5 Discusión

La definición de OA, después de más de 20 años², sigue siendo tema de debate. Si bien se han propuesto una gran cantidad de definiciones [26–28], la pregunta ¿Qué es un OA? sigue sin tener una respuesta única. Esto obliga, o debería hacerlo, a que quienes trabajan con OA hagan explícita una definición. Además, la misma debería reflejar, al menos, los principales acuerdos alcanzados por la comunidad especializada en el paradigma. Caso contrario, es muy posible que al momento de hacer uso de los OA en un contexto real, se utilicen recursos u otro tipo de materiales educativos que no debieran ser considerados OA.

En el contexto de las MEOA aquí analizadas, puede observarse de manera concreta, lo expuesto en el párrafo anterior. Al analizar las publicaciones pudieron identificarse tres grupos en función de la forma en que cada MEOA representa los OA: (i) los que utilizan una representación de los OA a través de metadatos, ontologías o servicios web (12 de 33); (ii) los que optan por estándares de empaquetamiento como SCORM³ o IMSCP⁴ (10 de 33); y (iii) aquellos que optan por otros tipos de materiales como son: slide o dispositivas (3 de 33), WebQuest⁵ (1 de 33), GLO⁶ (1 de 33) o recursos y actividades disponibles en el entorno LAMS⁷ (1 de 33). Para el resto (5 de 33) no se encontró información sobre el tipo de material con el que representan/asocian los OA.

Al enfocarse en la definición de OA es posible realizar un análisis de la caracterización de los OA hecha por las distintas MEOA. Se podría afirmar, observando la Fig. 2, que las MEOA definen los OA de forma general como un MED reutilizable etiquetado con metadatos.

Asimismo, otro análisis que resulta de interés es contrastar las definiciones presentadas por las MEOA, con una definición del concepto que atienda a las principales características en las que acuerdan los autores de referencia en la temática. En este trabajo se elige la definición de [30]:

“Un tipo de MED, que se caracteriza, desde el punto de vista pedagógico, por orientarse a un objetivo específico de aprendizaje, y por presentar: una serie de contenidos con el fin de abordar la temática relacionada con el objetivo, actividades

2 Wayne Hodgins introduce por primera vez el concepto en 1994.

3 Sharable Content Object Referente Model (<http://www.adlnet.org/scorm/>).

4 IMS Content Packaging (<https://www.imsglobal.org/content/packaging/index.html>).

5 Lección orientada a investigar utilizando información de la web (<http://www.webquest.org/>).

6 Generative Learning Object (Objetos de Aprendizaje Generativos) [29].

7 Learning Activity Management System (<http://lamsfoundation.org/>).

que permitan al alumno poner en práctica o problematizar el contenido presentado, y una autoevaluación que posibilite conocer al alumno, si ha podido comprender esos contenidos vinculados al objetivo. Desde el punto de vista tecnológico, se caracteriza por contener un conjunto de metadatos estandarizados para su búsqueda, y recuperación, y estar integrado, utilizando un modelo de empaquetamiento que respete estándares, y de esta manera, permita su diálogo con diferentes entornos tecnológicos.”.

En la Tabla 2 se propone dicha comparativa –se incluyen, por cuestiones de espacio, sólo las MEOA con mayor coincidencias con la caracterización elegida–. Puede observarse que menos de la mitad de las MEOA hacen una conceptualización apropiada de OA y sólo dos explicitan la estructura interna de objeto, aunque para las que usan SCORM/IMSCP se podría asumir que siguen la estructura propuesta por el estándar.

Tabla 2. Comparativa entre la definición de Sanz y las conceptualizaciones las MEOA.

Sanz	Punto de vista pedagógico			Punto de vista tecnológico		
	Dig.	Intenc. pedagógica	Estructura interna	Metadatos	Interoperable	SCORM IMSCP
S05	✓	✓		✓	✓	✓
S06	✓	✓	✓	✓	✓	
S20	✓	✓		✓	✓	✓
S23	✓	✓		✓	✓	✓
S35	✓	✓		✓	✓	✓
S01	✓	✓		✓		✓
S07	✓	✓		✓		✓
S08	✓	✓		✓	✓	
S13	✓	✓		✓	✓	
S18	✓	✓		✓	✓	✓
S21	✓	✓		✓	✓	
S22	✓			✓	✓	✓
S24	✓	✓			✓	✓
S31	✓			✓	✓	✓
S45	✓	✓	✓	✓		

6 Conclusiones

Los criterios definidos para llevar adelante el análisis de las MEOA han permitido identificar fortalezas y debilidades de las metodologías aquí revisadas. Una de las principales debilidades se vincula con los aspectos abordados en este trabajo, donde se observa que los tipos de materiales que se utilizan si bien son denominados OA refieren a diferentes caracterizaciones y formatos.

Al intentar llevar adelante un estudio comparativo de las MEOA, la falta de una definición acordada y de consenso generalizado sobre el concepto de OA, dificulta la comparación de las mismas. Máxime cuando éste no se define de forma explícita en la publicación donde se describen algunas de las metodologías.

Asimismo, la mayoría de las definiciones utilizadas por los autores de MEOA no tienen en cuenta las principales características, que cuentan con el acuerdo de la comunidad científica especializada en la temática.

Todo esto lleva a que para la adopción de este tipo de metodologías, y de los SE que las implementan, deba analizarse en detalle el tipo de material con el que se cuenta o bien diseñar material *ad-hoc* que se adapte al propuesto por la MEOA.

El diseño de MED, y de OA en particular, es un proceso costoso, como también lo es la adaptación del mismo a un nuevo formato. Por lo que este aspecto limitaría la adopción de MEOA.

Se cree que es necesario continuar con estas investigaciones y aportar respuestas claras y consensuadas en relación a qué es un OA. Esto favorecería tanto el diseño y producción de OA, así como también la reutilización de los mismos y la adopción/creación de nuevos enfoques metodológicos/tecnológicos que deseen hacer uso y reuso de este tipo de MED, particularmente para las MEOA.

Referencias

1. Astudillo, G.J., Sanz, C.V., Santacruz Valencia, L.P.: Estrategias de diseño y ensamblaje de Objetos de Aprendizaje, <http://sedici.unlp.edu.ar/handle/10915/53442>, (2016).
2. Santacruz-Valencia, L.P., Delgado Kloos, C., Cuevas Aedo, I.: Automatización de los procesos para la generación ensamblaje y reutilización de Objetos de Aprendizaje, www.lite.etsii.urjc.es/liliana/Defensa_Tesis_LPSV.pdf, (2005).
3. Stănică, J.L., Crișan, D.A.: Dynamic Development And Assembly Of Learning Objects In A Math Learning Environment. *J. Inf. Syst. Oper. Manag.* 6, 29–40 (2012).
4. Wiley, D.: The post-LEGO learning object, <http://opencontent.org/docs/post-lego.pdf>, (1999).
5. Wiley, D.: The learning objects literature. In: *Handbook of research on educational communications and technology*. pp. 345–353. Taylor & Francis, New York/London (2007).
6. Wiley, D.: Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In: *The Instructional Use of Learning Objects* (2000).
7. Bouzeghoub, A., Selmi, M.: Authoring Tool for Structural and Semantic Coherence Validation of Composed Learning Objects. In: *Advanced Learning Technologies, 2009. ICALT 2009*. pp. 175–177. , Riga, Letonia (2009).
8. Rigaux, P., Spyrtos, N.: Selene report: Metadata management and learning object composition in a self elearning network. Last Accessed Sept. (2007).
9. Schreurs, J., Dalle, R., Sammour, G.N.: Authoring Systems Delivering Reusable Learning Objects | Schreurs | International Journal of Emerging Technologies in Learning (iJET). *Int. J. Emerg. Technol. Learn. IJET.* 4, 37–42 (2009).
10. Menéndez Domínguez, V.H., Castellanos Bolaños, M.E., Zapata González, A., Prieto Méndez, M.E.: Generación de objetos de aprendizaje empleando un enfoque asistido. *Pixel-Bit Rev. Medios Educ.* 141–153 (2010).
11. Roig Vila, R.: Diseño de materiales curriculares electrónicos a través de Objetos de Aprendizaje. *RED Rev. Educ. Distancia.* 1–9 (2005).
12. Ullrich, C., Melis, E.: Pedagogically founded courseware generation based on HTN-planning. *Expert Syst. Appl.* 36, 9319–9332 (2009).
13. López, M.G., Miguel, V., Montaña, N.E.: Sistema Generador de AMBientes de Enseñanza-ApRendizaje Constructivistas basados en Objetos de Aprendizaje (AMBAR): la

- Interdisciplinariedad en los ambientes de aprendizaje en línea. *Rev. Educ. Distancia*. 1–14 (2008).
14. Becerra, C., Astudillo, H., Mendoza, M.: Improving learning objects recommendation processes by using domain description models. *Conf. LACLO*. 3, (2012).
 15. Farrell, R.: Dynamic Assembly of Learning Materials in a Corporate Context. *Educ. Technol.* 46, 70–73 (2006).
 16. Pahl, C., Barrett, R.: A web services architecture for learning object discovery and assembly. In: *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*. pp. 446–447. ACM, New York, NY, USA (2004).
 17. Colucci, S., Di Noia, T., Di Sciascio, E., Donini, F.M., Ragone, A.: Semantic-based automated composition of distributed learning objects for personalized e-learning. In: *The Semantic Web: Research and Applications*. pp. 633–648. Springer (2005).
 18. Li, Y., Huang, R.: Dynamic composition of curriculum for personalized e-learning. IOS Press. 151, 569–576 (2006).
 19. Lopes Gançarski, A., Bouzeghoub, A., Defude, B., Lecocq, C.: Iterative search of composite learning objects. In: *IADIS International Conference WWW/Internet*. pp. 8–12. , Vila Real, Portugal (2007).
 20. Karam, N., Linckels, S., Meinel, C.: Semantic Composition of Lecture Subparts for a Personalized e-Learning. In: *Franconi, E., Kifer, M., and May, W. (eds.) The Semantic Web: Research and Applications*. pp. 716–728. Springer Berlin Heidelberg (2007).
 21. Sarasa, A., Piquer, J., Arriola, R., Iglesia, S.: LOMEditor: Composition and Classification of Learning Objects. In: *Mendes, A., Pereira, I., and Costa, R. (eds.) Computers and Education*. pp. 241–249. Springer London (2008).
 22. Wetzlinger, W., Auinger, A., Sary, C.: Ad-hoc Composition of Distributed Learning Objects using Active XML. *Int. J. Emerg. Technol. Learn. IJET*. 3, 33–39 (2008).
 23. Kellar, M., Stern, H., Watters, C., Shepherd, M.: An Information architecture to support dynamic composition of interactive lessons and reuse of learning objects. In: *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on System Sciences*. p. 10 pp. IEEE, Hawaii (2004).
 24. Torres, I.-D., Guzmán-Luna, J.A.: Composition of Learning Routes Using Automatic Planning and Web Semantics. In: *Sobh, T. and Elleithy, K. (eds.) Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*. pp. 321–328. Springer International Publishing, Online (2015).
 25. Astudillo, G.J., Sanz, C.V., Santacruz Valencia, L.P.: Criterios para evaluar metodologías de ensamblaje de objetos de aprendizaje. In: *2015 International Symposium on Computers in Education (SIIE)*. en imprenta, España (2016).
 26. Astudillo, G., Sanz, C., Willging, P.: Análisis del estado del arte de los objetos de aprendizaje. Revisión de su definición y sus posibilidades, <http://sedici.unlp.edu.ar/handle/10915/4212>, (2011).
 27. Callejas Cuervo, M., Hernández Niño, E.J., Pinzón Villamil, J.N.: Objetos de aprendizaje, un estado del arte. *Entramado*. 7, 176–189 (2011).
 28. Santacruz Valencia, L.P.: Objetos de aprendizaje: estado de la cuestión. In: *Actas del II Seminario de Investigación en Tecnologías de la Información: SITIAE 2008*. pp. 67–79. Universidad Rey Juan Carlos (2009).
 29. Damaševičius, R., Štuikys, V.: On the Technological Aspects of Generative Learning Object Development. In: *Mittermeir, R.T. and Sysło, M.M. (eds.) Informatics Education - Supporting Computational Thinking: Third International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2008 Torun Poland, July 1-4, 2008 Proceedings*. pp. 337–348. Springer Berlin Heidelberg, Berlin, Heidelberg (2008).
 30. Sanz, C.V.: Los objetos de aprendizaje, un debate abierto y necesario. *Bit Byte*. 1, 33–35 (2015).

Bot guía para un ambiente virtual inmersivo.

Alejandro Altube¹, Iris Sattolo¹, Liliana Lipera¹,

¹ Facultad de Informática Ciencias de la Comunicación y Técnicas Especiales, Universidad de Morón- Cabildo 183 Morón

alealtube@hotmail.com, iris.sattolo@gmail.com, llipera@unimoron.edu.ar,

Abstract. Los Ambientes Virtuales Inmersivos (AVIs) presentan características que son aprovechables para la simulación ya que los objetos que lo componen pueden contener script resultando ser objetos inteligentes. La posibilidad de incorporar Agentes Virtuales Autónomos (AVAs), para poder guiar a usuarios novatos dentro del metaverso construido en la Universidad de Morón, dio origen a este trabajo. En el mismo se presentan las características de un agente reactivo simple y su implementación en un entorno virtual inmersivo, utilizando las herramientas que brinda la plataforma Opensim. El bot resultante utiliza funciones propias de la plataforma NPC y funciones del lenguaje LSL.

Keywords: Inteligencia Artificial, Ambientes Virtuales Inmersivos, Agentes Reactivos Simples.

1 Contexto

Este trabajo es parte del proyecto de investigación que se está llevando a cabo en la FICCTE (Facultad de Informática, Ciencias de la Comunicación y Técnicas Especiales) perteneciente a la Universidad de Morón titulado: “Ambientes virtuales inmersivos, elementos interactivos”. Es la continuación del proyecto realizado en los años 2014-2016: “Ambientes virtuales inmersivos de aprendizaje orientados a la capacitación y al desarrollo de planes de evacuación”.

Para la construcción del metaverso se utilizó el software Opensim (aplicación servidora multiplataforma y multiusuario). Durante el período de investigación se presentaron los avances en los congresos CACIC 2013, WICC 2014 y 2015, TE&ET 2014 y 2015.

Actualmente, éste espacio virtual está montado en una isla definida en un servidor de prueba en la Universidad de Morón al cual se accede en forma local a través de una red LAN.

Una de las líneas de investigación del presente proyecto plantea la posibilidad de dotar al entorno y a los diferentes tipos de objetos de autonomía o inteligencia propia.

Dado que el usuario novel no está preparado para vivir situaciones desconocidas, al entrar las primeras veces al mundo, se propuso la incorporación de agentes inteligentes que habiten el metaverso aun cuando no existan usuarios reales y le brinden alguna ayuda.

2 Introducción

Se presenta en esta sección una síntesis de los ambientes virtuales inmersivos, la plataforma utilizada para el desarrollo del trabajo y una introducción a los agentes racionales.

2.1 Ambientes virtuales inmersivos

Los mundos virtuales inmersivos, (ambientes virtuales inmersivos, AVIs o metaversos) son entornos tridimensionales generados por computadora que recrean escenarios reales o imaginarios. En ellos el usuario puede interactuar a través de un avatar, su representación virtual, con el mundo y con otros usuarios.

Actualmente los ambientes virtuales muestran distintas actividades, como ejemplo de ellas podemos citar en el área de educación y pedagogía, el espacio desarrollado en Second Life (Harvard Law School) [1], donde los estudiantes interactúan entre sí como expositores y jueces de un tema. En turismo, según Cruz Ruiz en su tesis de grado [2], Grecia lidera un proyecto de turismo virtual de sus Islas (Epsilon, s.f.).

En cuanto al desarrollo de agentes inteligentes en los AVIs el trabajo propuesto por Fabiana Santiago Sgobbi et al [3] propone un asistente para el cuidado de la salud de los ancianos. En Gonzales Martínez [4] se propone un agente conversacional utilizando herramientas de procesamiento natural. En la aplicación desarrollada por Johnson et al [5], utiliza el entorno virtual para mostrar el uso de agentes virtuales y su interacción con los objetos inteligentes del entorno.

Estos espacios, en educación, permiten a los usuarios relacionarse en forma remota admitiendo mayor interactividad en contraparte con las plataformas bidimensionales utilizadas actualmente. El desarrollo de mundos virtuales, como innovación en la práctica educativa, ha creado nuevas posibilidades en la adquisición de conocimientos y la formación de procesos de aprendizaje en línea.

Según el informe de Gartner (2007) *“Las nuevas arquitecturas, los nuevos espacios interconectados, los nuevos estándares abiertos posibilitan que los usuarios de Internet tengan algún tipo de presencia dentro de los mundos virtuales basados en la imagen del usuario a través de su avatar o de entornos multiusuario y, aparentemente esta presencia ha aumentado”* [6]. A estos entornos Digitales Inmersivos se los puede considerar partes vitales del Web3D emergente.

En el año 2001 Castronova [7] definió las tres características más importantes de los mundos virtuales: Corporeidad, Persistencia, Interactividad.

“Corporeidad : las personas acceden al programa a través de una interfaz, la cual simula un entorno físico donde el usuario observa y se mueve en primera persona; el entorno generalmente está sometido a las leyes de la física y se caracteriza por la escasez de recursos”.

“Persistencia: el programa sigue funcionando independientemente, aun cuando los usuarios estén conectados o no, recuerda la localización de personas y cosas como también quién es propietario de los objetos”.

“Interactividad: es la cantidad posibles de interacciones que el usuario puede realizar en el mundo virtual, siendo la interacción la capacidad que el usuario tiene de modificar el medio”.

2.2 Opensim

Opensim brinda un espacio vacío, posibilita la creación de un ambiente con objetos editables que pueden simular entes reales al adquirir comportamiento mediante scripts. Estos elementos pueden interactuar entre sí o con los avatares/usuarios. Los espacios consisten en una o más regiones que componen el metaverso. Para acceder al servidor Opensim desde los usuarios, se utilizan visores (programas clientes) que permiten controlar el avatar, editar objetos y crear scripts. Una gran variedad de ellos existen en el mercado, creados por grupos independientes y de uso libre.

A los objetos que demuestran la posibilidad de interacción con el entorno, se los suele llamar objetos inteligentes.

Otra posibilidad que permiten estas plataformas es la incorporación de Agentes Virtuales Autónomos (AVAs) los cuales se definen como: “entidades de software que representan personajes virtuales inmersos en escenarios 3D, teniendo la capacidad de reaccionar al entorno que los rodea, simulando tener vida propia”. Estos agentes dentro de Opensim son llamados bots.

2.3 Agentes

Se presentan las características de los agentes y se intenta catalogar a los AVAs (Agentes virtuales autónomos) dentro de las diferentes clasificaciones.

Existen distintas definiciones sobre agentes dependiendo el punto de vista utilizado, según Franklin y Graesser: “*un agente autónomo es un sistema que ubicado dentro de un entorno, percibe dicho entorno y actúa sobre él, a lo largo del tiempo, en busca de cumplir sus objetivos, y por tanto para llevar a cabo lo que percibe en el futuro*” [8].

Según Wooldridge y Jennings [9] existen dos nociones de agente, una débil y una fuerte. Un agente débil es aquel que goza de las siguientes propiedades:

- Autonomía: Operan sin una intervención directa de humanos, tienen un cierto grado de control de sus acciones y estado interno.
- Habilidad social: Se comunican con humanos u otros agentes a través de sus interfaces de comunicación.
- Reactividad: Actúa como resultado de cambios en su entorno. En este caso, un agente percibe el entorno y esos cambios dirigen su comportamiento.
- Proactividad: No solo actúan en respuesta a su ambiente, sino que son capaces de presentar un comportamiento oportunista, dirigido por objetivos, tomando iniciativas cuando sea apropiado.

Un agente fuerte, además de las características anteriores, posee una o más de las siguientes características:

- Nociones mentales: Posee creencias, deseos e intenciones.
- Racionalidad: Realizan acciones con el propósito de lograr objetivos.
- Adaptabilidad: Son capaces de aprender y cambiar su comportamiento en base a sus propias experiencias.
- Veracidad: No son capaces de comunicar conscientemente información falsa.

No es el fin de este trabajo nombrar las distintas características propuestas por los diversos autores, por tal motivo, se resolvió adoptar el esquema propuesto por Osorio et al[10] el cual sugiere que el agente puede ser catalogado según las características observada en él (Tipo de entidad, la similitud con el ser humano, según la arquitectura de control, las tareas que realiza, grado de autonomía, localización, tipo de entorno, tipo de actividad, tipo de interacción). En la tabla 1 se resumen las características observadas (columna 1), su parametrización (columna 2) y la categorización que nuestro bot tendría dentro de los AVAs (columna 3).

Tabla 1. Clasificación de Agentes de acuerdo a sus características.

Características observadas según	Clasificación	Agentes virtuales autónomos.
Tipo de entidad	Real (Humano, biológico, robot físico) Computacional (software, vida artificial)	Computacional (vida artificial)
Similitud con el humano	En cuanto a la estructura o Comportamiento	Estructura y comportamiento
Arquitectura de control	Reactivos, cognitivos, híbridos. Basados en estados, o en emociones	Reactivo Basado en Estados
Tareas que realiza	Transaccional, informativo, negocios, agentes de interfaz	Informativo
Grado de autonomía	Guiados, interactivos, autónomos, perceptivos.	Interactivos
Localización	Móvil, estacionario, distribuido	Móvil
Tipo de entorno	Escritorio, Red, Pedagógico, Virtual	Virtual
Tipo de actividad	Solitarios Social (grupos o colaborativos)	Colaborativo
Tipo de interacción	Con el usuario, otros agentes, con el entorno, o múltiples.	Interacciones múltiples.

2.4 Esquema de un agente reactivo simple

Según Nwana, los agentes reactivos se caracterizan porque pueden operar rápida y efectivamente, sin la necesidad de procesar una representación simbólica del entorno en que se encuentran, ya que constituyen una categoría de agentes que no posee modelos simbólicos externos [11]. Los agentes reactivos toman decisiones basadas en el presente, actúan siguiendo un esquema estímulo-respuesta según el estado actual del entorno en el que están inmersos.

Según Iglesia [12] un agente reactivo está formado por los siguientes tipos de módulos:

Módulos de interacción con el entorno (sensores y actuadores en la Figura 1).

Módulos de competencia (cada una de la capas en la Figura 1).

El funcionamiento es el siguiente: los sensores recogen información que es directamente enviada a los módulos de competencia. La llegada de información a uno

de estos módulos provoca en él una reacción que se transfiere al entorno a través de los actuadores.

Según el informe de la escuela Politécnica de Madrid, cuando los módulos de competencia trabajan en paralelo, lo normal es que aparezcan dependencias, por este motivo, se deben poder comunicar entre ellos. La comunicación normalmente utiliza tanto mecanismos como lenguajes de comunicación simples, y puede ser uno-a-uno o se puede realizar a través del entorno [13].

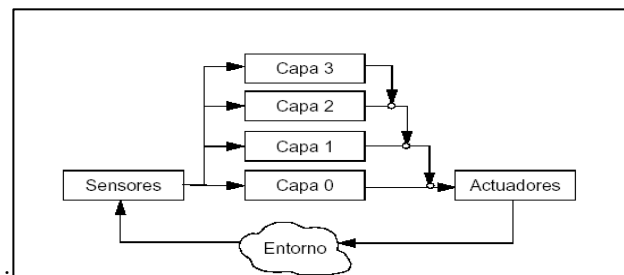


Figura 1 Arquitectura de un agente reactivo [12]

2.4 Bots

Existen dos alternativas que ofrece Opensim para la implementación de bots. La primera opción es crearlos mediante un script en LSL (Linden Scripting Language), y la segunda opción es mediante un cliente controlado por programa.

Los scripts son programados en LSL, lenguaje de programación basado en eventos, estados y funciones; creado para ser utilizado en ambientes de virtuales de Second Life.

LSL brinda una gran cantidad de funciones pero se han desarrollado en Opensim funciones propias llamadas OS. Dentro de las mismas, las NPC (Non Player Characters) que se utilizan para crear y manipular bots. Se describen aquí las más importantes para este trabajo:

Creación del bot: la función *osNpcCreate* permite su creación indicando nombre, apellido, posición inicial (dentro del ambiente) y apariencia. Esta apariencia se define dentro de una *notecard*.

Mover al bot: desde la posición actual hasta una fija. Siempre en línea recta. Para obtener un trayecto no recto se debe llamar varias veces a estas funciones. Se puede indicar que camine, corra o vuele.

Obtener su posición, rotación y dueño (número identificador del usuario que lo creó).

Guardar/cambiar de apariencia.

Comunicarse: Se pueden crear canales de comunicación para que los objetos interactúen entre sí o con usuarios y, en este caso, los bots.

Sentarse y pararse tanto en objetos como en el suelo.

Tocar objetos.

Ejecutar animaciones: Animar al bot, se pueden generar movimientos o usar los genéricos del Opensim y aplicarlos al bot. De esta manera se logra que el bot baile, salude.

En la segunda opción, implementación de bot a través de un cliente controlado por programa, para realizar la conexión se debe crear un programa en C# .NET, utilizando la librería OpenMetaverse. Esta librería, creada por *OpenMetaverse Foundation* y de acceso libre, es una interfaz que permite conectar usuarios a Opensim y ejecutar las acciones del avatar mediante funciones. Lógicamente, se necesita un usuario de Opensim disponible para la conexión.

3 Desarrollo

Se propuso en este trabajo la utilización de las herramientas que brinda la plataforma Opensim, sin la introducción de herramientas de Inteligencia Artificial.

Se planteó adoptar la arquitectura de un agente reactivo simple, utilizando para la construcción del mismo la alternativa que ofrece el entorno, creando el bot con funciones NPC.

Adoptando la definición que Russell y Norvig ofrecen de agentes en forma genérica y conceptual reflejada en la figura 2, “Un agente es cualquier cosa capaz de percibir su medioambiente con la ayuda de sensores y actuar en ese medio utilizando actuadores” [14], se analiza cómo, el bot, puede realizar esas acciones.

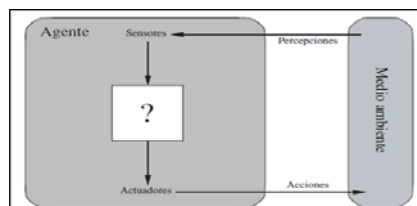


Figura 2 Arquitectura de un agente simple

De acuerdo con Aylett y Luck [15], la percepción es la interacción del agente con su entorno, el agente debe percibir diferentes objetos, pudiendo ser estos estáticos (como las paredes, muebles y otros), con movimiento (puertas que se abren), avatares en movimiento como también otros agentes.

El mecanismo empleado por el bot para esta interacción será la utilización de sensores para obtener información del entorno como disparador. Las funciones `llsensor`, `llSensorRepeat`, son provistas por el lenguaje pero no son propias del bot.

En cuanto al comportamiento que se espera del bot, será la acción que debe realizar de acuerdo a cada percepción. Podrá comunicarse, moverse y tocar objetos para interactuar con el mismo.

Existe la posibilidad de que el bot de respuesta a preguntas del usuario mediante el chat del Opensim, generando así un chatbot. En general existen movimientos predefinidos que se pueden utilizar para dar mayor sensación de realismo. Los actuadores serán los responsables de realizar estas acciones. La figura 3 muestra la arquitectura propuesta para implementar el bot.

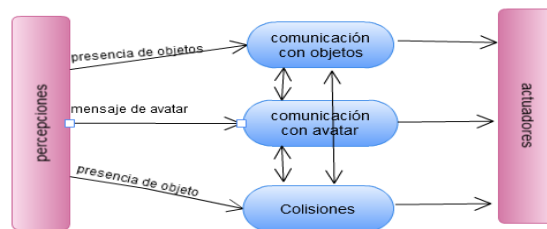


Figura 3. Arquitectura del bot.

3.1 Ambiente

El bot se moverá dentro del ambiente ya creado durante el proyecto “Ambientes virtuales inmersivos de aprendizaje orientados a la capacitación y al desarrollo de planes de evacuación” (figura 4) En este caso se propone un “bot guía” para dar soporte a posibles usuarios novatos que deseen conocer las distintas ubicaciones del entorno. El bot se encargará de guiar a los usuarios a locaciones específicas, enseñando las tareas que se pueden realizar en las mismas.



Figura 4 Ambiente creado en la Universidad de Morón

El bot se situará en el punto de inicio marcado con una flecha roja. Cuando el usuario ingrese al edificio el bot se acercará y preguntará, mediante el chat, si necesita dirigirse a alguna sala o locación en especial. Éstas se referencian en el mundo por las coordenadas (x,y,z). Tales como: Puerta de entrada (28,212,24); Sala de cascos (136,216,24); Sala de máquinas (18,214,24).

3.2 bot guía

Se muestra la implementación del módulo comunicación con el avatar, referenciando a cada percepción el comportamiento y el actuador correspondiente, a su vez, las acciones que realiza el bot. El paso entre ellas dependerá de las percepciones recibidas del usuario tanto en el sensor como en el chat y el menú (Figura 5).

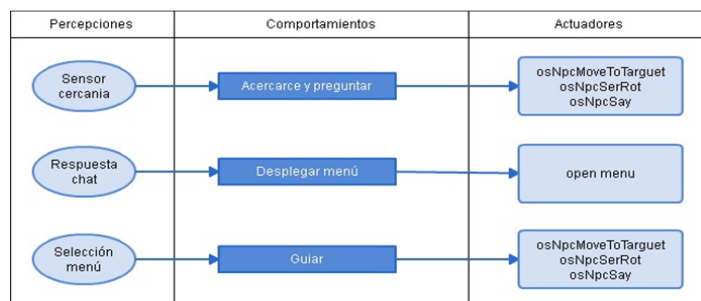


Figura 5 Percepciones y actuadores del bot.

Cuando el *Sensor cercanía* detecta a un avatar, el bot preguntará al usuario si necesita sus servicios esperando una respuesta por sí o por no. El accionar en cada alternativa será la siguiente:

1. Respuesta “sí”:
 - a. Se despliega un menú con las locaciones disponibles.
 - b. El usuario selecciona la locación.
 - c. El bot guía al usuario hasta la locación seleccionada y le brinda información útil sobre la misma.
2. Respuesta “no”:
 - a. El bot se despide y vuelve a su ubicación inicial.

3.4 Implementación de prueba

Las funciones NPC que brinda Opensim no son suficientes por sí solas para llevar a cabo el desarrollo del bot y sus funciones, por lo que se resolvió combinarlas con el resto de las funciones LSL. Ya se describieron las funciones NPC, en cuanto a las LSL usadas se necesitaron las siguientes: sensor de cercanía (del avatar con el bot), el menú de locaciones, y la comunicación entre bot y avatar vía chat, entre otras.

Para la implementación del bot se requirió la construcción de diferentes objetos (la figura 6 muestra una captura de pantalla de los mismos):

Objeto Bot guía:

Script 'botguia': contiene el `OsNpcCreate`, se ejecutará al hacer click sobre el objeto que lo contiene, se comunica con el objeto 'Comandos del bot', enviándole el número identificador del bot creado, a través de un canal establecido.

Apariencia: Notecard con la apariencia que tomara el `OsNpcCrate`.

Objeto Comandos del bot:

Script 'comandosbot': se ejecutará cuando reciba el “mensaje” con el número identificador del bot npc por el canal establecido. Contendrá el código del bot.

Objeto Copiar apariencia:

Script 'copiarapariencia': simplemente copia la apariencia del avatar usuario. Lo importante es que al momento de copiar la apariencia se tenga adjuntado (lleve consigo) el objeto 'Comandos del bot', ya que esta apariencia será la que tome el objeto 'Bot guía' al momento de crear el bot.

AparienciaBot: *Notecard* que se creara al hacer click sobre este objeto.



Figura 6 Objetos para implementar el bot

Al no existir funciones NPC que permitan adjuntar un objeto al bot, una vez creados los objetos que contienen los scripts responsables de la comunicación, se deben colocar en el inventario del avatar que será copiado. Si bien resultará más laborioso a la hora de modificar el código, ya que se deberá repetir el proceso: *copiar apariencia y adjuntar objeto* (acoplamiento), se logra de este modo la comunicación entre el avatar y el bot mediante la instancia *comandos del bot*. Se muestra en la figura 7 un diagrama de clases para la implementación del mismo.

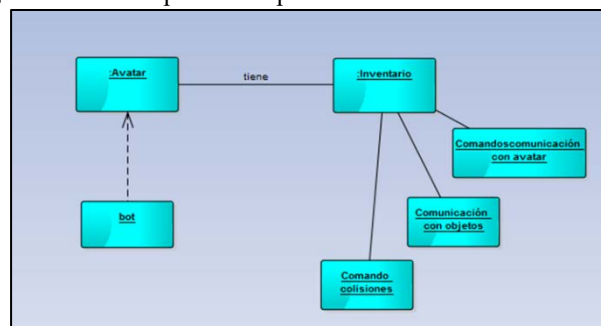


Figura. 7 Diagrama de clases para implementar el bot

Conclusiones y futuras líneas

Se logró que el bot guía realice tareas con las características de agentes débiles: autonomía, habilidad social y reactividad.

Resultó ser un agente reactivo simple, ya que actúa sobre el entorno en base a las percepciones recibidas.

Mayor interacción entre el usuario y el ambiente al crear un bot que simule un personaje más real.

En cuanto a las futuras líneas de investigación se nombran:

- Posibilidad de implementación de un bot a través de un cliente/usuario controlado por programa.
- Creación de agentes conversacionales, llamados chatbots, mediante AIML (Artificial Intelligence Mark-up Language).

- Implementación de multiagentes en ambientes virtuales.
- Búsqueda de posibilidad de creación de un agente virtual que cumpla con definiciones de agente fuerte, tales como: Agregar emociones a los agentes virtuales, Aprendizaje en agentes virtuales, Adaptar comportamiento de agentes en base a estado anímico y experiencia.

Referencias

1. <http://maps.secondlife.com/secondlife/Berkman/123/219/24?title=Harvard+Law+School&msg=The+Berkman+Center+for+Internet+%26+Society+at+Harvard+Law+School+has+held+multiple+classes+within+Second+Life>.
2. <http://eprints.ucm.es/13048/1/memoria.pdf>
3. Fabiana Santiago Sgobbi, Felipe Becker Nunes, Liane Taroucco A UTILIZAÇÃO DE AGENTES INTELIGENTES NO APOIO AO AUTOCUIDADO DE IDOSOS- <http://seer.ufrgs.br/index.php/renote/article/view/53520/33032>
4. Gonzalez Martínez. Agente conversacional con habilidades para la venta de productos en un mundo 3D - <http://www.informaticahabana.cu/sites/default/files/ponencias/EDU058.pdf>
5. W. L. Johnson, and J. Rickel, "Steve: An Animated Pedagogical Agent for Procedural Training in Virtual Environments", Sigart Bulletin, ACM Press, vol. 8, number 1-4, 16-21, 1997.
6. Gartner Inc. (2007). Gartner Says 80 Percent of Active Internet Users Will Have A "Second Life" in the Virtual World by the End of 2011. Retrieved May 10, 2008 from <http://www.gartner.com/it/page.jsp?id=503861>
7. Castronova, E. (2001). Virtual Worlds: A first-hand account of market and society on the cyberian frontier .
8. Intelligent Agents III Agent Theories, Architectures, and Languages is i tan agent, or just a program? A taxonomy for autonomous agents. http://www.springer.com/la/book/9783540625070?wt_mc=ThirdParty.SpringerLink.3.EPR653>About_eBook
9. Wooldridge M., Jennings N. R. (1995). Intelligent Agents: Theory and Practice. The Knowledge Engineering Review, Vol. 10:2, pp. 115-152
10. Osorio, Musse, Santos, Heinen, Braun, Silva. Intelligent Virtual Reality Environments .Principles, Implementation, Interaction, Examples, and Practical Applications.
11. Nwana, H. (1996). Software agents: an overview. The Knowledge Engineering Review, 11(3):205-244. <http://www.btexact.com/projects/agents/publish/papers/review3.htm>.
12. Iglesias, C.A (1997) Fundamentos de los Agentes Inteligentes. Informe Técnico UPM/DIT/GSI 16/97.
13. Universidad politécnica de Madrid. Informe técnico. Arquitecturas de Agentes
14. 12 Rusell S, Norvig P. (2004). Inteligencia artificial: Un enfoque moderno 2da edición. Editorial Pearson.
15. Aylett, R. and Luck, M. (2000) "Applying Artificial Intelligence to Virtual Reality: Intelligent Virtual Environments". Applied Artificial Intelligence.

Transformación de Exámenes de Ambientes E-Learning en Videojuegos Web

Curras, David ¹, Dabove, Mariano ¹,
Sartorio, Alejandro ², Vaquero, Marcelo ², Diamand, Luciano ²

¹ Grupo de desarrolladores e investigadores del proyecto ² Grupo de docentes
Universidad Abierta Interamericana (UAI)
Centro de altos estudios en tecnología informática (CAETI)

Resumen. El objetivo de este trabajo es presentar un modelo para la creación de juegos educativos como metáforas representativas de las interfaces de la herramienta de examen utilizada en un eco-sistema de e-learning. Las plataformas de e-learning se componen de Objetos de Aprendizaje que los educadores pueden crear utilizando herramientas de software. Controlando estos objetos es posible establecer métodos de evaluación lúdicos para los estudiantes.

Palabras Clave: videojuego, educación, examen, web, objetos de aprendizaje, e-learning.

1 Introducción

Un juego educativo es un medio a través del cual los usuarios adquieren cierto conocimiento a partir de sus interacciones con el mismo. Los juegos educativos combinan la motivación necesaria para captar la atención de los usuarios, junto con los objetivos de aprendizaje que subyacen a las actividades que se realizan en el juego. Teniendo en cuenta estos aspectos, es posible facilitar este proceso para cada individuo mediante la adaptación de los juegos con los que pueda interactuar ya que ayudan a un desarrollo más amigable y sustentable en su educación y formación cognitiva puesto que a veces resulta tedioso que un estudiante aprenda con un libro [1]. Gracias a esto, nos planteamos los siguientes interrogantes: 1- ¿Podemos ayudar a los jóvenes mediante un videojuego, a que logren sostener en el tiempo y forma un proceso de aprendizaje? 2- ¿Podemos lograr que dichos jóvenes se “apropien” de las nuevas tecnologías? 3- ¿Podemos utilizar una plataforma educativa como mecanismo para que estos aprendizajes puedan ser compartidos con sus pares, y así lograr una mayor integración sin verse afectada la estructura y plan de estudios? [2].

Una de las principales actividades de los jóvenes es jugar. Desde que son pequeños, aprenden desde colorear, contar, sumar o leer ya sea con libros o juegos didácticos. Estas herramientas junto a la informática educativa surgen como estrategia para utilizar correctamente las nuevas tecnologías en forma de aprendizaje. Desde entonces han evolucionado tanto los juegos, como su metodología. De hecho, los juegos virtuales educativos generan un medio didáctico en el que se pueden crear actividades orientadas a la construcción del conocimiento en todas las disciplinas. Se

favorece un aprendizaje lúdico, alternativo y autónomo en el que el estudiante deja de utilizar un cuaderno y lápiz, para interactuar con la PC [3].

Las nuevas tecnologías comenzaron a ser utilizadas por los docentes como nuevos métodos pedagógicos que suponen un soporte interactivo a las “clases magistrales” [4]. Uno de ellos puede ser aprender con actividades lúdicas. La idea no es nueva ya que cuando jugamos construimos un espacio imaginario en el que se ensayan destrezas y capacidades, que quizás no están dentro de nuestras posibilidades inmediatas, pero que podemos utilizarlas en un futuro [3].

Los juegos educativos resultan difíciles de adaptar para los educadores que desean utilizarlos en sus cátedras. Esto se debe a que las actividades que realiza el usuario para lograr los objetivos formativos son seleccionadas previamente por el desarrollador del juego.

El contenido educativo digitalizado, puede organizarse en Objetos de Aprendizaje [12] y contener Metadatos de Objetos de Aprendizaje [13] para describir la estructura de los datos que el juego debe consumir, permitiendo la flexibilidad necesaria para adaptarlos a las cátedras. Las plataformas de e-learning se componen de Objetos de Aprendizaje, que los educadores pueden crear utilizando herramientas de software desarrolladas para tal fin [14].

Las plataformas de e-learning son una herramienta importante para la organización y distribución de contenido educativo, así como para la creación de cursos virtuales de enseñanza a distancia. Muchas de estas plataformas proveen herramientas para la evaluación de los conocimientos del alumno mediante la creación de formularios con preguntas y problemáticas redactadas por el educador. En este contexto, las plataformas de e-learning pueden convertirse en una fuente de contenido para los videojuegos educativos, aportando imágenes, documentos y textos que formen parte del juego, así como preguntas y actividades para la evaluación del aprendizaje. Las bases de datos de las plataformas de e-learning pueden ser utilizadas para parametrizar el videojuego, y así facilitar el desarrollo de un juego capaz de motivar al alumno y permitir la adaptación del mismo a los contenidos que se desea enseñar.

Estos juegos utilizan elementos lúdicos como fuente de conocimiento y motivación para el proceso de aprendizaje, y junto con la flexibilidad de las plataformas de e-learning, las cuales proveen un mecanismo de configuración, evaluación y modificación del contenido, permiten guiar a los usuarios de forma personalizada durante dicho proceso.

El objetivo de este trabajo es presentar un modelo para la creación de juegos educativos como metáforas representativas de las interfaces de la herramienta examen utilizada en un eco-sistema e-learning. Siguiendo los requerimientos descritos en la sección 3, expondremos el diseño e implementación de una interfaz que conecte el contenido generado por una plataforma de e-learning para parametrizar un juego educativo.

2 Estado del Arte

A continuación se describen partes de los trabajos utilizados como punto de partida e inspiración para este trabajo. Estas son las referencias de los seis trabajos procesados:

El trabajo 1 [1] propone un modelo para la presentación y generación de juegos adaptativos, utilizándolos como fuentes de conocimiento y motivación para el proceso de aprendizaje. Toma varios capítulos donde cuenta la necesidad de los juegos educativos adaptativos, desarrolla el modelo teórico de juego, el proceso de adaptación y la parametrización de valores y finaliza con un ejemplo de implementación del modelo.

Los trabajos 2, 3, 4 y 5 [2,4,5,6,10] muestran el diseño y desarrollo de juegos educativos, incluyendo soluciones con herramientas como Invocación Remota de Métodos (IRC), protocolo HTTP, preguntas basadas en la Especificación para la Interoperabilidad de Preguntas y Exámenes [4], y desarrollo para plataformas móviles. También exponen los resultados de implementar los juegos en torneos de matemática y cursos universitarios.

El trabajo 6 expone la evolución del proceso de aprendizaje con los cambios tecnológicos y la forma en que se fue adaptando a los diferentes avances en las TIC. También describe los distintos tipos de plataformas educativas de e-learning, sus características y herramienta.

Tomando como referencia los trabajos anteriormente expuestos, a continuación se presenta una tabla comparativa buscando los ítems más significativos y referenciales para llevar a cabo el proyecto.

Tabla 1. Comparación de proyectos anteriores con el proyecto actual.

TRABAJOS	C1	C2	C3	C4	C5	C6	C7	C8
Generación de Juegos Educativos Adaptativos	x	x	x	x	x			
Diseño y desarrollo de un juego educativo para ordenador sobre enfermedades tropicales y salud internacional	x	x	x	x				
Trivia Mathematica	x	x	x	x				
ProBot: Juego para el aprendizaje de lógica de programación	x	x	x	x	x			
Diseño del juego educativo UNA-Pregunta	x	x	x	x	x	x	x	
Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos						x		x
Trabajo actual	x	x	x	x	x	x	x	x

C1: Videojuego. C2: Juego Educativo. C3: Diseño. C4: Desarrollo.
C5: Juego Parametrizable. C6: Web. C7: Movil. C8: Plataforma e-learning.

3 Análisis

Definimos los requerimientos para la interfaz de conexión entre la herramienta examen y el videojuego educativo.

Requerimientos Funcionales. 1. La interfaz deberá conocer la estructura de datos de la plataforma de e-learning. 2. El juego deberá identificarse para acceder a los métodos de la interfaz. 3. El juego podrá obtener los datos de parametrización desde la interfaz en cualquier momento que lo requiera. 4. La interfaz deberá reconocer los parámetros que hayan cambiado en la plataforma de e-learning y tenerlos disponibles para el juego de forma instantánea. **Requerimientos No Funcionales.** 1. La interfaz

será web, para poder acceder de forma remota desde diferentes instancias del juego. 2. El juego se identificará, utilizando una clave privada generada por la interfaz, utilizando el estándar JWT (RFC 7519) [11].

4 Statechart como transformador de un examen en un videojuego

Utilizamos el lenguaje de especificación gráfico Statechart creado para modelar el comportamiento de los aspectos dinámicos que permitirán interpretar los enunciados de un examen en una instancia de un juego. Statecharts, fue introducido originalmente con D. Harel [8] como la extensión de máquinas finitas con conceptos de jerarquías y paralelismo. Aprovecharemos su representación visual para implementar una formalización del modelado de los aspectos visuales y convencionales de un examen, sus fundamentos y utilización, para usarlo como traductor de escritura de un videojuego en lenguaje de scripting de fácil utilización en los ambientes Web donde son implementados en instancias de herramientas de exámenes en ambientes e-learning modernos. Luego transformaremos los Statecharts en módulos funcionales para implementar aplicaciones de juegos Web. Usaremos JavaScript [7], un lenguaje liviano y potente, ideal para el tipo de representación que se necesita implementar. Por ejemplo, supongamos que se tiene un sitio e-learning [9] configurado con una herramienta examen en las que hay diferentes preguntas y se deben seleccionar distintas opciones. Simplificando la notación, se puede representar los elementos relacionados como: Pregunta \rightarrow Opciones \leftarrow Respuestas \rightarrow Caminos. Las flechas indican dependencias entre elementos. Por ejemplo, una Pregunta tiene opciones, las Respuestas son una instancia de las Opciones seleccionadas y al mismo tiempo definen un Camino de navegación hipertextual dentro del sitio. Un camino puede ser el link a la siguiente pregunta, la navegación a otra herramienta, o contenido hipertextual relacionado con el tipo de respuesta seleccionado.

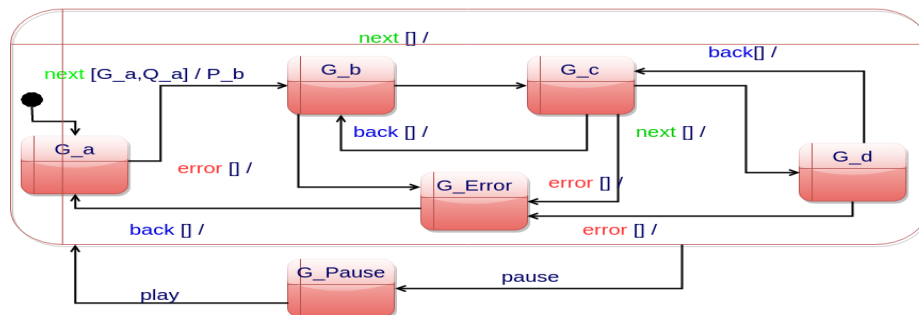


Fig. 2 (a). Statechart denominada Game para especificar los escenarios del juego.

Las estrategias de especificación están basadas en el diseño de tres máquinas principales. La máquina *Game*, está compuesta por estados que representan diferentes escenarios visuales del juego. La figura 1(a) representa estados conceptuales que identifican el inicio de un escenario de juego con el estado G_a . Cuando se produce un evento *next* ocurre una transición a otra instancia del escenario del juego. Por

ejemplo, se muestra por pantalla un nuevo obstáculo. De igual manera, con el evento *back* se vuelve al estado anterior. Las pausas en el juego se encuentran especificadas por medio de un super-estado que abarca estados internos ($G_a .. G_d$). Con el evento *pause* se puede transicionar desde cualquiera de estos estados al estado G_{Pause} para representar la acción de pausa en el juego. De la misma manera se especifican las distintas posibilidades de error por medio de la transición de los estados ($G_a .. G_d$) hacia G_{Error} . En este caso, para cada G_x (x genérico) sale una flecha para G_{Error} , permitiendo identificar diferentes tipos de errores que se pueden producir dependiendo del estado anterior.

Ahora se explican algunos aspectos conceptuales que se tuvieron en cuenta para la interpretación funcional de esta máquina implementada en módulos JavaScript que se muestra en el fragmento de código 2 de la clase *Game*. En este caso los eventos *next* y *back* son interpretados a través del método *updateGame*. Los fenómenos *winGame* y *endGame* tienen el comportamiento similar a los anteriores, denotan el comienzo del juego y la instancia de finalización en el marco de un escenario específico. La conexión de la máquina con el entorno se produce a través de las interfaces del módulo, representadas por las operaciones que lo definen. Por ejemplo, las operaciones *pauseCondition* y *pauseReceiver* implementan la transición de los estados G_x (x: *a..c*) hacia el estado G_{Pause} . Estas operaciones acceden al estado de un *buffer* interno que se utiliza como soporte para implementar las funcionalidades del entorno de las máquinas.

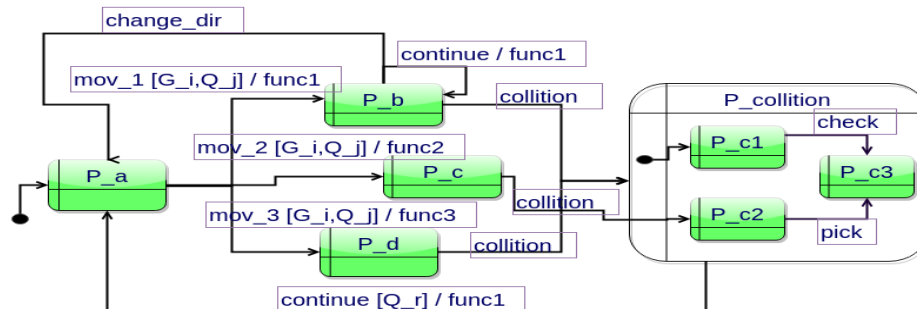


Fig. 1 (b). Statechart denominada Player para especificar las funcionalidades del juego.

La máquina *Player* se encarga de la representación funcional del juego. La figura 1 representa los movimientos que puede hacer el personaje principal del juego. En este caso se especifican cada uno de esos movimientos a través del evento *mov_i* (i: 1...N), para N casos posibles. Por ejemplo: giros, avances, saltos, cambios de velocidad, selección de caminos en lugares bifurcados, etc. De la misma forma se resuelven las funcionalidades de las colisiones. Cuando ocurre una colisión, la máquina *Player* adquiere conocimiento de este fenómeno si puede capturar el evento *collision*. En ese caso, se accede a una nueva máquina ($P_{collision}$) que se encargará de especificar el comportamiento del sistema cuando ocurra una colisión. Hay accesos desde tres tipos de estados diferentes. En la figura 1(b) hay tres estados representativos de estas posibilidades. El estado P_b captura una colisión y pasa la máquina de colisiones que automáticamente pasa al estado P_{c1} . Lo mismo ocurre si

desde P_d se captura el evento *collition*. El estado inicial de la super-máquina $P_{collition}$ se encuentra referenciado con el círculo negro. Luego, esta máquina se encargará de especificar todas las funcionalidades necesarias para el manejo de la lógica de colisión. En este caso, a través de los eventos *check* y *pick* que permitirán sincronizar el comportamiento con la máquina *Question* encargada de especificar las estructuras y propósitos de las preguntas del examen.

Siguiendo con la misma estrategia de la máquina anterior, analizaremos la implementación realizada en JavaScript a través del módulo *Player* de la figura. Este módulo configura tres tipos de eventos: *movePlayer*, para el control de los movimientos e implementación de los eventos *mov_i* de la maquina *Player*; *checkCollision*, que implementa el evento *collition*; y *talk* que se utilizará para representar los eventos *check* y *pick* dentro de la máquina *Player*.

Por último, en la Figura 1(c) muestra las principales componentes de nuestra propuesta de Statechart para la especificación de la estructura del examen, respetando los conceptos anteriormente mencionados. Partiendo del estado inicial Q de la máquina *Question* hay dos posibles caminos de transición, uno para el control de una respuesta a través del evento *check* y otro para el evento *pick*. El evento *check* espera la aprobación de la validación con el evento *valid* y luego puede derivar en tres posibles estados. Hacia el estado *Predicate* con el evento *pred*, al estado *Condition* con el estado *cond* y el estado *Receive* con el evento *rec*. El otro camino comienza con el evento *pick*, iniciando la selección de una consigna para que resuelva el usuario. Pasa por el estado intermedio *Check* a través de *one* y luego hacia *Condition* y *Receive* con los eventos *cond* y *rec*, respectivamente.

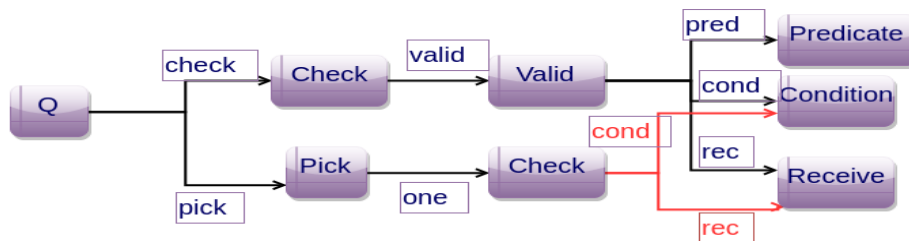


Fig. 1 (c). Statechart denominada *Question* para especificar el cuestionario del examen.

En cuanto a la implementación en módulo JavaScript de este Statechart, el método *checkValidPredicate*, que representa los eventos *check*, *valid*, *pred* y *pickOnePredicate*, se encarga de representar los eventos *pick* y *one*. Al igual que en los módulos anteriores, se encuentran operaciones como *pickOneCondition* encargada de conectar con el entorno, en este caso el *buffer* interno, con el propósito de verificar condiciones de posibles transiciones de estados. Por otro lado, la operación *pickOneReceiver* tendrá la responsabilidad de referenciar la pregunta adecuada y actualizar el *score* del usuario participante.

Este caso se utiliza para modelar la representación de un examen como un videojuego. La figura 1 representa tres máquinas en paralelo. Esta representación determina unos de los principales conceptos que se pretende exponer en este trabajo. Una vez entendido el propósito, se conformará una estructura eficaz para aplicar

cualquier tipo de examen (selecciones múltiples, asociar, textos libres, mensajes de aviso según respuestas, etc.) para cualquier videojuego que se pueda expresar en el grupo de máquinas de la figura 1.

Por cuestiones de espacio, concentramos en esta sección el principal concepto que se quiere transmitir en este trabajo como una propuesta de transformación del formato de formulario web como herramienta de examen, hacia un videojuego que contenga el espíritu y los objetivos de esa instancia de evaluación.

5 Diseño

Arquitectura de la aplicación. El juego educativo obtiene los datos necesarios para adaptar su contenido a través de la interfaz de conexión web, la cual posee métodos públicos conocidos por el juego. La interfaz tiene acceso a la base de datos de la plataforma de e-learning con permisos de solo lectura, y consulta los datos cada vez que un juego se conecta para verificar si hubo modificaciones. La plataforma de e-learning se encuentra conectada a la base de datos de forma local y puede realizar modificaciones en cualquier momento. La figura 2(a) muestra de forma gráfica la arquitectura de la aplicación.

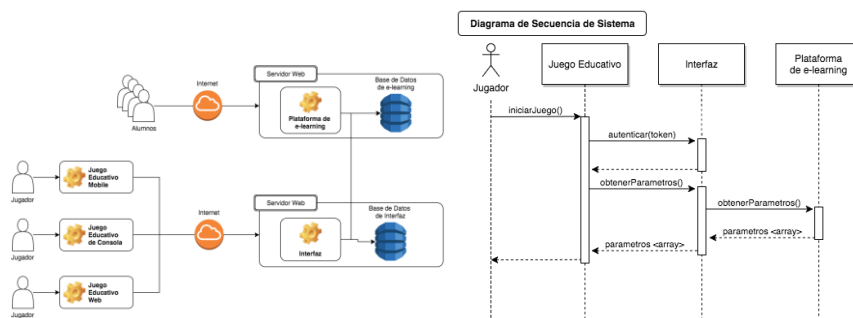


Fig. 2 (a). Arquitectura de la aplicación.

Fig. 2 (b). Diag. Secuencia del Sistema (DSS).

El Diagrama de Secuencias de Sistema expuesto en la figura 2(b) muestra las interacciones entre el jugador, el juego educativo, la interfaz de conexión y la plataforma de e-learning. También se pueden ver la firma de los métodos expuestos por la interfaz.

6 Implementación

Se ha creado un juego educativo web, el cual utiliza una interfaz de conexión para acceder a la base de datos de una plataforma de e-learning para adaptar su contenido a diferentes cursos.

Utilizando el lenguaje JavaScript [7], desarrollamos una abstracción de Statecharts [8] para implementar la interfaz, y un juego gráfico de plataformas, con preguntas y respuestas que se seleccionan de forma dinámica dependiendo del nivel y la

experiencia del jugador. De esta manera concretamos los conceptos desarrollados en la sección 4.

```
game$ ls js/abstract/
condition.js          machine.js           receiver.js         state.js
emitter.js           predicate.js        state-buffer.js    super-machine.js
game$ ls js/game
game.js              player.js           questions.js
```

Fig. 3. Estructura de archivos.

La figura 3 muestra la estructura de archivos creada, donde la carpeta *abstract* contiene una representación abstracta del modelo Statecharts, y la carpeta *game* posee las clases necesarias para crear el juego.

Cada una de las clases dentro de *game* extiende de la clase *Machine* (Máquina), la cual posee un estado de clase *State* y una lista de predicados que son instancias de la clase *Predicate*.

Fragmento de código 1. JavaScript de las clases *Machine* y *Condition*.

```
var Machine = (function() {
  'use strict';
  function Machine(state, pred) {
    var t = this;
    if (!(t instanceof Machine)) {
      return new Machine(state);
    }
    this.state = state;
    this.pred = pred;
  }
  return Machine;
})();

var Condition = (function() {
  'use strict';
  function Condition(cFn) {
    var t = this;
    if (!(t instanceof Condition)) {
      return new Condition(cFn);
    }
    Condition.prototype.evaluate = cFn;
    return Condition;
  }
})();
```

El predicado es responsable de realizar las transiciones entre estados de la máquina a la que pertenece, utilizando las instancias de *Emitter* y *Receiver*. El receptor de eventos ejecutará las acciones necesarias para transicionar al siguiente estado, solo cuando la condición (*Condition*) sea verdadera.

En el Fragmento de código 2, se puede ver que la clase *Game* extiende de *Machine* (línea 50), y en su constructor se crean las instancias de *State* (línea 10) y *Predicate* (líneas 43 y 44) necesarias para iniciar y finalizar el juego. El código final del constructor de la clase *Game* es más extenso y posee 8 predicados para poder transicionar entre los diferentes estados de la máquina. De manera similar, las clases *Player* y *Question* extienden de *Machine*, y transicionan sus estados mediante predicados instanciados de la clase *Predicate*.

Cada máquina puede emitir eventos que serán recibidos por ellas mismas, o por otras máquinas. Para ello se utilizó la clase *Event* de JavaScript como un *EventManager*, donde todos los eventos emitidos son capturados por *Event*, y enviados al receptor que esté esperando escucharlo.

Cuando el jugador se mueve, la máquina *Player* envía el evento *movePlayer*, el cual es capturado por la instancia *Receiver* del predicado *movePlayerPredicate*. En este caso, la máquina *Player* posee los predicados para resolver la transición del

estado por sí misma, es decir, que emite y recibe el evento utilizando sus propios predicados.

Fragmento de código 2. JavaScript de la clase Game.

```
var Game = (function() {
  'use strict';
  function Game(args) {
    if (!(this instanceof Game)) { return new Game(args); }
    this.gameState = new State({ state: {}, context: {} });
    var startC = function() {
      return StateBuffer.gameState === 'idle';
    };
    var startR = function() {
      StateBuffer.gameState = 'playing';
      var rows = 10;
      var columns = 5;
      var blockWidth = (this.gameState.width / rows);
      var blockHeight = 30;
      for (i = 0; i < columns; i++) {
        for (j = 0; j < rows; j++) {
          id = 'block'+j+i;
          block.create(id);
          this.gameState.dynamicList.push(block.list[id]);
        }
      }
      window.onkeydown = keyboard.press;
      window.onkeyup = keyboard.release;
      setInterval(game.predicates.updateGame, 1000/60);
    };
    var updateC = function() {
      return StateBuffer.gameState === 'playing';
    };
    var updateR = function() {
      for (i = 0; i < this.gameState.staticList.length; i++) {
        this.gameState.staticList[i].render();
      }
      for (i = 0; i < this.gameState.dynamicList.length; i++) {
        this.gameState.dynamicList[i].update();
        this.gameState.dynamicList[i].render();
      }
      if(!Object.keys(block.list).length) {
        game.predicates.winGame();
      }
    };
    var gStart = new Predicate('startGame', startR, startC);
    var gUpdate = new Predicate('updateGame', updateR, updateC);
    var gMachine = new Machine(gameState, {
      startGame: gStart,
      updateGame: gUpdate,
    });
    this = _.extend(this, gMachine);
  }
  return Game;
})();
```

Fragmento de código 3. JavaScript de la clase Predicate.

```
var Predicate = (function() {
  'use strict';
  function Predicate(evt, evtFn, cFn) {
    if (!(this instanceof Predicate)) {
      return new Predicate(evt, evtFn);
    }
    this.receiver = new Receiver(evt, evtFn, cFn);
    this.emitter = new Emitter(new Event(evt));
  }
  return Predicate;
})();
```

La máquina *Player* también posee predicados que emiten eventos para detectar si el jugador está en un sector del juego donde es necesario mostrar una nueva pregunta, en cuyo caso se disparará el evento *pickQuestion*, que será tomado por la máquina *Question*, la cual, luego de hacer las comprobaciones necesarias, enviará el evento *showQuestion* pasando como parámetro el texto de la pregunta. Aquí la máquina *Game* lo recibe y utiliza los predicados necesarios para cambiar su estado y así mostrar la pregunta correspondiente en pantalla.



Fig. 4. Juego educativo mostrando una pregunta con sus posibles respuestas.

7 Adaptaciones

El modelo presentado es fácilmente adaptable a diferentes tipos de juegos, con nuevos desafíos para permitir evaluar al estudiante desde enfoques más dinámicos que un simple cuestionario.

Adaptación del tipo de juego. Si se pretende realizar un nuevo juego basándose en el modelo propuesto, podemos optar por modificar la apariencia gráfica y parte de la lógica, pero manteniendo la modalidad de evaluación mediante preguntas y respuestas. La mecánica del juego se encuentra definida dentro de las máquinas *Game* y *Player*. Modificando el código dentro de los predicados *startReceiver* y *updateReceiver* de la clase *Game*, y dentro del predicado *moveReceiver* de la clase

Player, se puede lograr un juego con apariencia y jugabilidad diferentes pero con el mismo sistema de preguntas y respuestas de opciones múltiples.

Adaptación del método de evaluación. También se pueden realizar modificaciones al tipo de examen. Las plataformas educativas permiten crear Objetos de Aprendizaje para evaluar al estudiante con diferentes métodos, como pueden ser las preguntas con opciones múltiples, verdadero o falso, y de redacción.



Fig. 5. Juego educativo con modificaciones en las clases Game y Player.

Para crear un juego que responda a otro tipo de método de evaluación, es necesario modificar la implementación de la máquina *Questions*, haciendo que el predicado *pickQuestion* provea el desafío acorde al nivel del jugador; y dentro del predicado *evaluateAnswer* se deben establecer las reglas que determinen si la respuesta es correcta.

8 Conclusiones y trabajo futuro

El modelo creado permite describir y generar juegos educativos para evaluar los conocimientos adquiridos por el estudiante, dependiendo del método de evaluación y los contenidos generados por el docente dentro de una plataforma de e-learning.

Este modelo ha sido aplicado a un juego de preguntas y respuestas con múltiples opciones, y en un futuro próximo se utilizará en un contexto real para obtener datos que permitan determinar bajo qué condiciones se obtienen los mejores resultados.

Se está considerando ampliar el modelo para incluir nuevos métodos de evaluación y enseñanza, aprovechando las herramientas que proveen las plataformas de e-learning mediante Objetos de Aprendizaje.

Entre los objetivos a largo plazo, se encuentra la creación de un formalismo para la descripción de los componentes de los juegos educativos y su relación con los contenidos y métodos de evaluación.

Referencias

1. Carro, R.M., Breda, A.M., Castillo, G., Bajuelos, A.L.: Generación de Juegos Educativos Adaptativos. (2002)
2. Giménez, P.C.; Arévalo, C.P.; Martínez Herráiz, J.J: Diseño y desarrollo de un juego educativo para ordenador sobre enfermedades tropicales y salud internacional: una herramienta docente más de apoyo al profesor universitario. (2011)
3. León, M.: Los métodos de enseñanza aprendizaje y la informática. Facultad de ciencias médicas, Las Tunas, Cuba. (2004)
4. Morales, R., Gallegos, E.P.: Trivia Mathematica: Una Experiencia de Desarrollo con Software Libre. (2012)
5. Moreno, J., Montaña E.A.: ProBot: Juego para el aprendizaje de lógica de programación. (2009)
6. Ramírez, E.R, Sandí, H.R.: Diseño del juego educativo UNA-Pregunta: promoviendo la participación y el aprendizaje en línea. (2014)
7. What is JavaScript? https://developer.mozilla.org/docs/Web/JavaScript/About_JavaScript
8. Harel, D. (1987). Statecharts: A Visual Formalism for Complex Systems. *Science of Computer Programming* 8 231-274. North-Holland. (1987)
9. Sartorio, A., & Cristiá, M.: Primera aproximación al diseño e implementación de los DHD. In XXXIV Congreso Latinoamericano de Informática. (2008)
10. Boneu, J.M.: Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos. (2007)
11. JSON Web Token (JWT). RFC 7519. <https://tools.ietf.org/html/rfc7519>
12. IEEE 1484.12.1 – 2002 Standard for Learning Object Metadata.
http://129.115.100.158/txor/docs/IEEE_LOM_1484_12_1_v1_Final_Draft.pdf
13. Metadatos de Objetos de Aprendizaje.
<http://www.asociacionelearning.com/contenidos/lom-learning-object-metadata.asp>
14. Cohen, E. B., Nycz, M.: Learning Objects and E-Learning: an Informing Science Perspective. (2006)

Integrando modelo de aprendizaje supervisado al análisis del desempeño de alumnos en cursos virtuales sobre plataformas Moodle

Rodrigo Jacznik¹, Mariano Tassara¹, Ivan D'Uva¹, Guillermo Baldino¹, Roxana Giandini¹ y Leopoldo Nahuel¹

¹Grupo de Investigación y desarrollo Aplicado a Sistemas Informáticos - GIDAS
UTN Facultad Regional La Plata
{rjacznik ,mtassara, iduva, gbaldino, rgiandini, lnahuel} @linsi.edu.ar

Abstract: Los cursos virtuales, al ser entornos asíncronos y no presenciales, pueden dificultar el seguimiento de los alumnos. Los Entornos Virtuales de Educación y Aprendizaje (EVEA) suelen proveer herramientas para dicho cometido. En Moodle, algunos plugins toman información relativa a las actividades de los alumnos, entregando estadísticas al profesor. Esta información puede no ser precisa con respecto a la capacidad de liderazgo o riesgo de abandono. El uso de Redes Neuronales Artificiales (RNA) puede ayudar a predecir el comportamiento de los alumnos y obtener conclusiones en etapas tempranas del proceso de aprendizaje en un EVEA. En trabajos anteriores se ha desarrollado un plugin para Moodle que analiza métricas sociales mediante la teoría de grafos. Este artículo esboza las ventajas de integrar a este desarrollo una RNA que complemente el uso del grafo para brindar conclusiones ricas sobre el desempeño de los alumnos en un curso virtual de Moodle.

Keywords: Entornos virtuales de Educación, Aprendizaje Supervisado, Moodle, Redes Neuronales

1 Introducción

La educación virtual es un modelo que trata de ser flexible. Está basado en la filosofía de la asincronía y no concurrencia, es decir, no es necesario que coincidan ni en el tiempo, ni en el espacio, tampoco los agentes que participan en el proceso de aprendizaje.

Uno de los problemas que tiene la Educación Virtual y la Educación a Distancia es el poco contacto personalizado con el alumno y, por lo tanto, el poco conocimiento de las situaciones personal que atraviesa con respecto al aprendizaje en el curso. Además también existen los problemas que comparte con el modelo de Educación Presencial en general, como el temor de que muchas personas comiencen los cursos y luego los abandonen. Es por eso, que el desarrollo de herramientas que permitan medir las relaciones dentro del aula virtual y sus implicancias en el rendimiento de los alumnos se tornan valiosas en la medida que permiten anticipar acciones a tomar. En trabajos anteriores se ha desarrollado un prototipo del plugin IARSE para un curso virtual de Moodle [1] para medir las interacciones en un foro del curso, aplicando los criterios de la métrica centralidad. De esta forma el docente tiene una herramienta que le ayuda a reconocer, dentro de ese curso, aquellos alumnos que puedan tener ascendencia o mayor participación que sus compañeros.

El creciente uso de plataformas como los EVEA, requiere establecer más herramientas para los profesores, como mecanismos de predicción del comportamiento que reemplacen la percepción que pueden tener en clases presenciales [2].

En el presente trabajo buscamos, a partir de la implementación del plugin IARSE, profundizar las herramientas de soporte al docente ampliando el prototipo de forma tal que incluya la aplicación de técnicas de machine learning que permitan al modelo dar soporte a decisiones del docente.

2 Actividades de los alumnos en un EVEA: caso de Moodle

El trabajo se desarrolló tomando el EVEA Moodle, el cual es utilizado como soporte en la enseñanza en distintas cátedras de la Facultad Regional La Plata de la Universidad Tecnológica Nacional. Particularmente el plugin IARSE se utilizó en un entorno de prueba simulando foros y participantes, y luego se hizo extensivo a un ejercicio real de un foro, de una cátedra actual de la carrera de Ingeniería en Sistemas de Información.

Entre las ventajas que existen por las cuales hemos decidido utilizar el EVEA Moodle, además de ser el utilizado en nuestra Facultad, podemos mencionar su extensibilidad y su extendido uso, que alcanza 9 millones de usuarios en 229 países, lo que permite la existencia de una comunidad fuerte que desarrolla una gran cantidad de plugins.

Moodle está basado en conceptos de constructivismo social, por lo que las actividades que se dan en un curso virtual de la plataforma hacen posible un esquema de enseñanza colaborativo, en el que los alumnos contribuyen a su propia formación. El profesor, además de ofrecer material, debe crear un ambiente que permita a los alumnos construir su propio conocimiento [3].

En cada curso virtual, el alumno puede contribuir realizando un conjunto variado de actividades. Las actividades más relevantes son:

Tareas: los alumnos pueden subir su trabajo para que sea calificado y comentado por el profesor.

Elección/Consulta: el profesor puede realizar una pregunta con opciones múltiples para que los alumnos la respondan. Es similar a la encuesta, pero contiene sólo una pregunta.

Feedback: es un tipo de encuesta que puede crear el profesor con sus propias preguntas, a diferencia de las encuestas predefinidas.

Foro: puede contener discusiones asincrónicas entre todos los participantes del curso.

Lección: permite crear una serie de páginas de contenido que los alumnos recorren de manera flexible.

Examen/Cuestionario: permite construir evaluaciones con múltiples características.

Encuesta predefinida: encuesta con preguntas estáticas sobre el curso.

Taller: es un espacio en el cual los alumnos evalúan trabajos de otros alumnos, guiados por el profesor [4].

En el caso de consultas, feedback y encuestas predefinidas, el análisis del comportamiento radica en saber si el alumno las respondió o no, al igual que para las lecciones es útil saber si las tomó. Las tareas y los exámenes tienen un nivel más de complejidad porque, además de la necesidad de saber si fueron realizados, hay que saber la nota obtenida. En cuanto a los foros y los talleres, proporcionan interacciones entre los participantes del curso, permitiendo detectar en los alumnos características como la centralidad, capacidad de liderazgo o aislamiento.

Además, existen otras actividades que no serán de utilidad para el desarrollo que se plantea, tales como: chat, base de datos, acceso a herramientas de aprendizaje externas, glosario, SCORM (Modelo de Referencia de Objeto de Contenido Compartible) y Wiki.

3 Plugin IARSE: análisis del curso mediante métricas sociales y grafos

Con el objetivo de ayudar al docente a aplicar mejoras en la calidad de aprendizaje de un curso virtual en Moodle, se ha desarrollado en trabajos anteriores un plugin que utiliza la teoría de grafos para aplicar métricas sociales para determinar el comportamiento de los alumnos e identificar problemas. El plugin crea un grafo con los alumnos como nodos y las interacciones entre ellos como arcos y calcula la métrica “centralidad” para saber cuál es el alumno más central en el grafo. Como primer desarrollo, se permitía armar un grafo por cada discusión dentro de un foro. Las últimas mejoras incluyen la realización del grafo del foro entero.

El objetivo final del plugin es el análisis del curso completo, teniendo en cuenta foros, actividades personales y actividades grupales de los alumnos para obtener

conclusiones que el docente a cargo del curso pueda interpretar, para proponer mejoras [1].



Figura 1. Plugin IARSE aplicado sobre una discusión

El grafo es creado completamente con código JavaScript en la vista, mediante la librería D3.js [5]. Los scripts PHP se limitan a obtener información de la base de datos de Moodle y presentarla de manera ordenada.

4 Análisis del desempeño de alumnos utilizando redes neuronales

Las Redes Neuronales Artificiales (RNA) son modelos de Inteligencia Artificial que emulan el funcionamiento del cerebro en cuanto a la conexión de sus células (neuronas). Este tipo de estructuras tienen las propiedades de realizar actividades de computación distribuida, tolerar entradas con mucho ruido y aprender. Las RNA tienen la capacidad de entender la significación de datos complicados, imprecisos o desestructurados y pueden usarse para extraer patrones, detectar tendencias o hacer predicciones, tareas que pueden ser demasiado complejas de realizar de otra manera.

Cada RNA es un conjunto de perceptrones conectados. Un perceptrón es una unidad con muchos canales de entrada y un canal de salida. Cada conexión tiene asignado un peso. Durante la fase de aprendizaje, la red aprende ajustando esos pesos para poder predecir la correcta salida para las tuplas de entrada. La conectividad entre los nodos de una red neuronal está relacionada con la forma en que las salidas de las neuronas están canalizadas para convertirse en entradas de otras neuronas. La señal de salida de un nodo puede ser una entrada de otro elemento de proceso, o incluso ser una entrada de sí mismo

(conexión autorrecurrente). Las conexiones pueden ser hacia adelante o hacia atrás con respecto a los niveles y una RNA puede tener uno o múltiples de ellos [6].

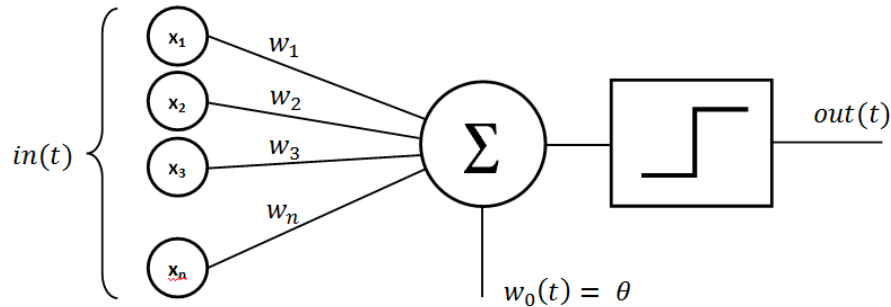


Figura 2. Perceptrón

Cada perceptrón de la RNA analiza un conjunto de entradas, aplicando el peso correspondiente de la conexión. La suma de las entradas multiplicadas por el peso es obtenida por una función que produce la salida del perceptrón.

$$y = f\left(\sum_{i=1}^n x_i w_i\right)$$

Cuando la red es multicapa, cada función es multiplicada por un peso, pero ahora la salida se usa para elegir entre muchas posibles salidas:

$$y = \operatorname{argmax}_y f(x, y)w$$

$$y = \operatorname{argmax}_y f(x, y)w$$

Durante el período de aprendizaje, se itera entre muchos ejemplos, dejando el peso cuando el resultado es correcto comparado con el deseado y cambiándolo si no lo es. Si la salida es binaria, el problema puede resolverse con un sólo perceptrón [7].

Como se explicará más adelante, en nuestro caso el resultado será discreto, limitando las posibles salidas a un vector de números enteros positivos. El tener una cantidad de salidas posibles mayor a dos hace que el modelo necesariamente sea multicapa.

4.1 Aprendizaje supervisado

Aprendizaje supervisado es un sub-categoría de machine learning. La elección de este tipo de aprendizaje radica en la naturaleza de los datos con los que contamos ya que utilizamos un set de entrenamientos etiquetados, esto es, se sabe el origen de los datos.

El objetivo principal de este estilo de aprendizaje es construir un modelo que realiza predicciones a partir de ejemplos.

Específicamente, un algoritmo de aprendizaje supervisado toma como entrada un conjunto de ejemplos (input) y su respuesta (output), y entrena al modelo para realizar predicciones aceptables para la respuesta a nuevos datos.

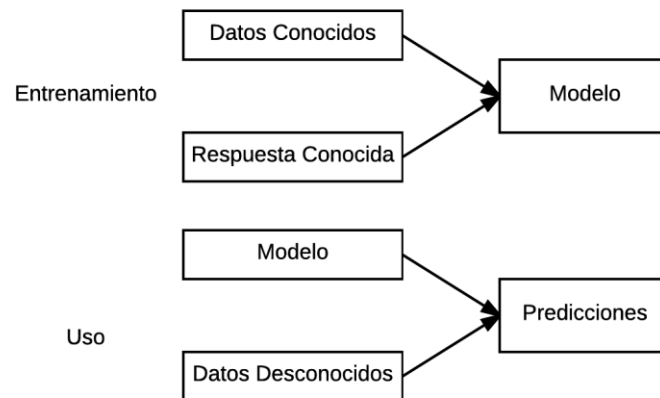


Figura 3. Proceso de aprendizaje supervisado

El aprendizaje supervisado se divide en 2 categorías:

- Clasificación: el objetivo es asignar una clase o etiqueta de una serie de clases a la predicción.
- Regresión: el objetivo es asignar un valor continuo a la predicción [6].

4.2 RNA aplicadas en educación

Usar RNA para problemas relativo al comportamiento humano, más específicamente en ambientes educativos, puede resultar adecuado por varias razones:

- Pueden encontrar patrones en conjuntos de datos no estructurados.
- Pueden acoplarse a las plataformas de e-learning, cuyo uso está en constante crecimiento.
- Su uso es adecuado para pronósticos o predicciones de problemas de categorización.

Estudios realizados en la Universidad Nacional Técnica de Atenas, Grecia sugieren que se pueden obtener predicciones precisas en cursos virtuales de 10 semanas a partir de la tercer semana. Las predicciones realizadas por RNA fueron comparadas con otras predicciones estadísticas de regresión lineal. La comparación arrojó resultados

favorables para las RNA, concluyendo que éstas son más eficientes en cualquier etapa de pronóstico [8].

En cuanto a la precisión de las RNA en el análisis del comportamiento de alumnos en ambientes educativos, un estudio realizado en el Departamento de Ingeniería de la Universidad de Ibadan, Nigeria, demuestra que una RNA basada en un modelo de perceptrón multicapa es capaz de predecir correctamente el rendimiento de más del 70% de los estudiantes analizados. El objetivo del estudio era predecir el rendimiento futuro de un alumno que estuviera siendo considerado para su admisión en la Universidad. Se analizaron datos personales de cada futuro alumno, como edad, tipo y localización de su escuela secundaria, notas en materias, entre otros. La red utilizada fue entrenada con los datos de ex alumnos de la Universidad [9].

5 Integración de una red neuronal al plugin IARSE

El objetivo principal de este trabajo es expandir el plugin IARSE con la capacidad de predecir el la nota de los alumnos de cursos virtuales a partir de su comportamiento dentro del EVEA.

El objetivo principal de este trabajo es expandir el plugin IARSE con la capacidad de predecir el comportamiento de los alumnos de cursos virtuales dentro de Moodle, tomando un set de datos de entrada compuesto de información personal y de información relativa a las actividades de cada alumno dentro del EVEA.

Más concretamente el resultado se obtendrá en forma numérica, representando la nota final del alumno. Esto, sumado a la información proveída por el grafo sobre las interacciones en el curso, puede permitir al profesor identificar tempranamente a los alumnos que tendrán problemas para afrontar las actividades, así como a los que presenten características de liderazgo.

Para lograr esto se hará uso de técnicas de aprendizaje supervisado.

5.1 Set de datos

Para generar el modelo utilizaremos como set de entrenamiento cursos ya finalizados.

Como datos de entrada, utilizaremos para cada alumno:

- Actividad de logueo: cantidad de conexiones por semana.
- Actividad en el foro (métricas): el plugin actualmente está preparado para calcular las métricas indegree y outdegree, que se refieren a las interacciones de entrada y de salida de los alumnos. Además otro dato de entrada sería la cantidad de discusiones generadas por cada alumno.
- Desempeño en consultas y encuestas: cantidad de consultas, feedback y encuestas predefinidas sobre el total de las realizadas en el curso.

- Desempeño en tareas y exámenes: cantidad de realizados sobre el total en el curso y nota promedio. Se representará con una sola entrada, ya que la nota también refleja la ausencia o no del alumno.
- Datos personales: edad, sexo, cantidad de cursos en los que participa.

Y como dato de salida la nota final del curso. En aprendizaje supervisado esto se traduce en 9 etiquetas para la predicción.

Además de predecir la nota final del curso, nos interesa intentar determinar el riesgo que tiene el alumno de abandonar el curso. Para realizar esta predicción, agregamos una etiqueta.

Como los datos de respuesta (output) son discretos, el problema a resolver es de categorización.

5.2 Estructura de la red neuronal

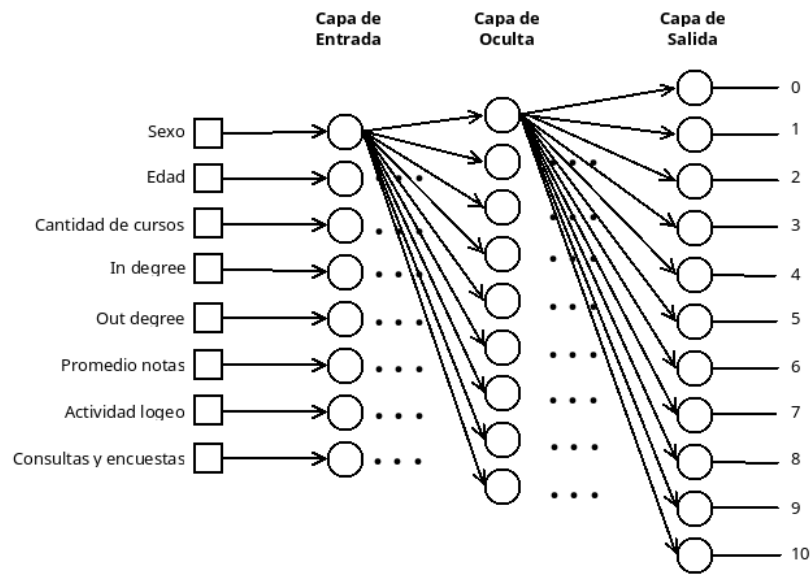


Figura 4. Estructura de la RNA. Cada neurona se conecta con todas las neuronas de la capa siguiente. En la imagen se ven las conexiones de la neurona superior de cada capa a modo de ejemplo.

La red neuronal que se utilizara para realizar la predicción es completamente conectada y tiene 3 capas.

- La capa de entrada, la cual tiene tantas neuronas como datos de entradas

- La capa de salida, con tantas neuronas como etiquetas de respuesta. La etiqueta 0 indica que el alumno abandonó el curso y de la etiqueta 1 al 10 indican la nota.
- La capa oculta. Inicialmente tiene 9 neuronas (el promedio entre las neuronas de entrada y salida). La cantidad de neuronas en esta capa se puede modificar con el fin de mejorar la performance de la red.

5.3 Entrenamiento

El objetivo del entrenamiento de la red neuronal es ajustar los pesos que cada neurona tiene con respecto a su entrada.

Para realizar el entrenamiento se realiza una validación doble, la cual consiste en dividir el set de datos en dos partes. La primera parte, que consiste en el 70% del set de entrenamiento, se utiliza para entrenar la red neuronal. La segunda parte, el restante 30%, se usa para validar la red entrenada.

Si el error que se produce en la respuesta es menor al error aceptable, consideramos la red como entrenada. En caso contrario, se deberán ajustar los valores de la red neuronal (los pesos iniciales, la función de paso de cada red, la cantidad de capas ocultas y la cantidad de neuronas en cada una de estas capas).

6 Conclusiones y trabajos futuros

El uso de técnicas de inteligencia artificial en ambientes virtuales de enseñanza, específicamente redes neuronales permite diseñar estrategias docentes sobre el abordaje de diferentes problemáticas dentro del aula. Permite también mejorar la elaboración de contenidos y orientar la clase de acuerdo a interés y problemáticas particulares de cada curso. Esto redundará en una mejora de las prestaciones de docentes y alumnos en los procesos de aprendizaje.

La ventaja más visible es aprovechar las predicciones para tener una mejor visión de la particularidad de cada alumno. Al tratarse de cursos virtuales, el profesor no tiene el contacto individual que es tan necesario en cursos presenciales. A partir de las puntuaciones que resultan del análisis, se pueden tomar acciones paliativas para los alumnos que abandonarían o no llegarían a aprobar, además se puede identificar a los alumnos con más incidencia en el curso teniendo en cuenta las notas altas y complementando este análisis con el uso del grafo que también es parte del plugin.

Como trabajo futuro planteamos la posibilidad de ampliar la arquitectura del plugin a otras plataformas educativas además de Moodle, y hacerlo extensivo a redes sociales en las cuales interactúan los grupos de alumnos.

Asimismo, exploramos la posibilidad de compartir datos anónimos entre distintos cursos con el fin de evolucionar la red neuronal y generar mejores resultados.

Referencias

1. Rodrigo Jacznik, Mariano Tassara, Ivan D'Uva, Guillermo Baldino, Herramienta de software pedagógica para identificar relaciones y comportamientos en entornos de educación virtual, CyTal (2016)
2. Petar Halachev, Prediction of e-Learning Efficiency by Neural Networks, Cybernetics and Information Technologies Journal (2012)
3. Reynaldo Pérez Casales, José Rojas Castro, Grismilda Pauli Hechavarría, Algunas experiencias didácticas en el entorno de la plataforma Moodle, Revista de Informática Educativa y Medios Audiovisuales (2008)
4. <https://docs.moodle.org/>
5. <https://d3js.org/>
6. Stuart Russell, Peter Norvig, Artificial Intelligence - A Modern Approach, Tercera Edición (2010)
7. Michael Collins, Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms, Conference on Empirical Methods in Natural Language Processing (2002)
8. Ioanna Lykourantzou, Ioannis Giannoukos, George Mpardis, Vassilis Nikolopoulos, Vassili Loumos, Early and Dynamic Student Achievement Prediction in E-Learning Courses Using Neural Networks, Journal of the American Society for Information Science and Technology (2009)
9. V.O. Oladokun, A.T. Adebajo, O.E. Charles-Owaba, Predicting Students' Academic Performance using Artificial Neural Network: A Case Study of an Engineering Course, The Pacific Journal of Science and Technology (2008)

Modelo de Proceso para Uso Didáctico en la Enseñanza de Ingeniería del Conocimiento

Ezequiel Scordamaglia, Dario Rodriguez, Hernan Amatriain, Ramón García-Martínez

Laboratorio de Investigación y Desarrollo en Ingeniería de Software
Grupo de Investigación en Sistemas de Información.
Departamento de Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús
rgm1960@yahoo.com

Resumen. El concepto de Modelo de Proceso esta arraigado en los nuevos enfoques de la Ingeniería del Software. La construcción de Sistemas Basados en Conocimiento (SBC) es un tema central en la enseñanza de Ingeniería de Conocimiento. Sin embargo, los modelos de proceso para el desarrollo de SBC propios de la Ingeniería de Conocimiento son complejos para un primer curso en la materia. En este trabajo se propone modelo de proceso de uso didáctico en construcción de sistemas basados en conocimiento.

Palabras Clave. Ingeniería del Conocimiento. Sistemas Basados en Conocimiento. Modelo de proceso para uso didáctico.

1. Introducción

La enseñanza de la Ingeniería de Conocimiento en grado tiene una larga tradición en Argentina. El plan de Licenciatura en Sistemas de Información de la Universidad Nacional de Lujan vigente en el año 1990 contaba con la asignatura “Sistemas Expertos”; y el Plan 1996 de Ingeniería Informática de la UBA preveía la asignatura “Sistemas Automáticos de Diagnóstico y Detección de Fallas” que en sus versiones tempranas se ocupaba de la utilización de Sistemas Expertos en el área de los sistemas de producción.

Desde el año académico 2015, la Licenciatura en Sistemas de la Universidad Nacional de Lanús cuenta con la asignatura electiva “Sistemas Basados en Conocimiento”. Como se describe en [García-Martínez et al., 2016] tiene los siguientes objetivos: “...[i] *Que el estudiante se familiarice con el abordaje metodológico del desarrollo de los sistemas basados en conocimiento, [ii] Que el estudiante desarrolle la capacidad de construir un sistema basado en conocimiento...*”. Entre los contenidos se destacan “...*Modelización de los Conocimientos Fácticos. Modelización de los Conocimientos Estratégicos. Modelización de los Conocimientos Tácticos. Generación del Modelo Dinámico...*”. Para el modelado de conocimiento en la práctica de la asignatura se presentan los siguientes formalismos: Tablas CAV, Tablas PER, Grafos Causales, Diagramas Jerárquicos de Tareas y Mapas de Conocimiento [García-Martínez y Britos, 2004].

Se enseña la Metodología Ideal [Carrillo Verdun, 1987; Gomez et al., 1997] que no propone técnicas para desarrollar las tareas que componen las fases, debiendo ser considerada por esta razón un modelo de proceso [Rodríguez, 2015].

La experiencia de dictado de la asignatura en 2015 ha puesto en evidencia que la Metodología Ideal es un modelo de proceso de una complejidad inapropiada para un primer curso Ingeniería del Conocimiento. Como alternativa fueron considerados los procesos propuestos por Grover [1983], por Brulé y Blount [1989] y por Schreiber [2000]. Aun difiriendo en su complejidad, también resultaban no utilizables en el curso de grado.

En este contexto, en esta comunicación se delimita el problema (sección 2), se propone como solución un modelo de proceso para uso didáctico en la enseñanza de ingeniería del conocimiento (sección 3), se da una prueba de concepto (sección 4), y se formulan conclusiones preliminares (sección 5).

2. Delimitación del Problema

La articulación de la enseñanza de Ingeniería del Conocimiento dentro del área Ingeniería de Software, requiere presentar un abordaje metodológico que le permita a los estudiantes tener un primer acercamiento con base ingenieril del proceso de construcción de un sistema basado en conocimiento. Sin embargo la complejidad de las metodologías usuales de INCO exceden las necesidades de un primer curso en la materia.

En este contexto, se plantea la necesidad de disponer de un modelo de proceso de uso didáctico para el desarrollo de sistemas basados en conocimiento que articule con los formalismos de modelado de conocimiento que se ven en la asignatura.

3. Solución Propuesta: Modelo de Proceso para Uso Didáctico en la Enseñanza de Ingeniería del Conocimiento

En la Figura 1 se presenta el Modelo de Proceso para Uso Didáctico en la Enseñanza de Ingeniería del Conocimiento propuesto como solución al problema identificado. El proceso se estructura en tres fases: segmentación, conceptualización y formalización e implementación.

La fase “segmentación” se compone de cuatro tareas: “identificación de estados”, “análisis de dependencias entre estados”, y “obtención de criterios para la clasificación de un estado”. La fase “conceptualización” se compone de cinco tareas: “identificación de conceptos atributos y valores”, “codificación de conceptos atributos y valores”, “identificación de reglas”, “modelado conceptual de reglas”, e “identificación de encadenamiento de tareas”. La fase “formalización e implementación” se compone de tres tareas: “modelado de estructuras en CLIPS”, “transformación de reglas conceptuales en reglas CLIPS”, y “armado de contexto CLIPS”. En la Tabla 1 se relacionan los artefactos insumo que requiere cada tarea y los artefactos producto que genera cada tarea.

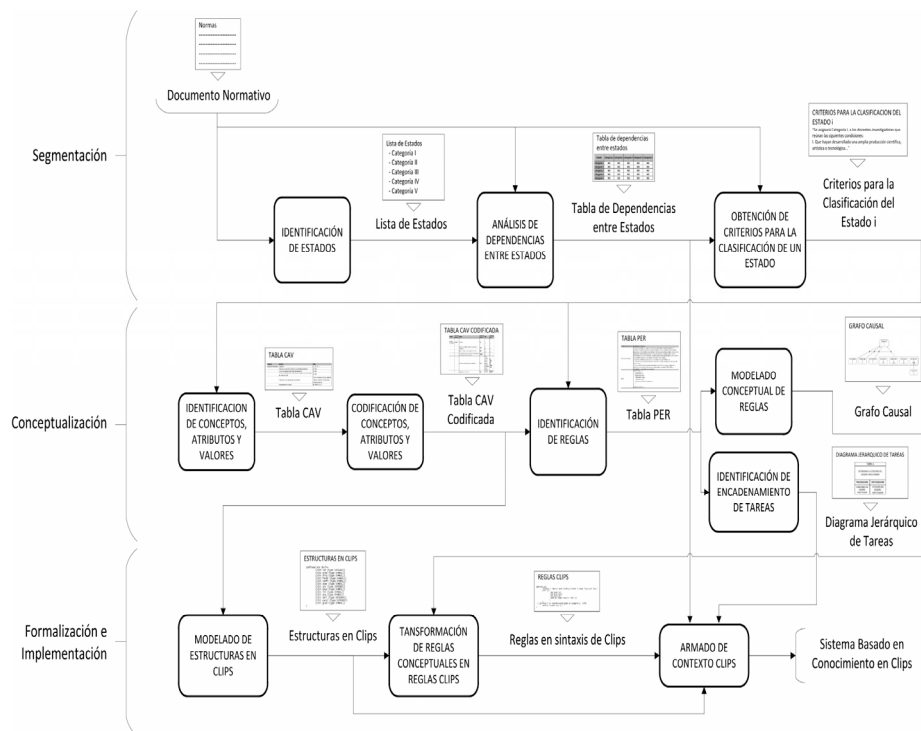


Fig. 1. Modelo de Proceso para Uso Didáctico en la Enseñanza de Ingeniería del Conocimiento

Tabla 1. Relación de artefactos insumo y producto asociado a cada tarea

FASE	TAREA	ARTEFACTOS INSUMO	ARTEFACTOS PRODUCTO
SEGMENTACIÓN	identificación de estados	documento normativo	listado de estados
	análisis de dependencias entre estados	<ul style="list-style-type: none"> documento normativo listado de estados 	tabla de dependencia entre estados
	obtención de criterios para la clasificación de un estado	<ul style="list-style-type: none"> documento normativo tabla de dependencia entre estados 	criterios para la clasificación del estado i
CONCEPTUALIZACIÓN	identificación de conceptos atributos y valores	criterios para la clasificación del estado i	tabla CAV
	codificación de conceptos atributos y valores	tabla CAV	tabla CAV codificada
	identificación de reglas	<ul style="list-style-type: none"> criterios para la clasificación del estado i tabla CAV codificada 	Tabla PER
	modelado conceptual de reglas	tabla PER	grafo causal
	identificación de encadenamiento de tareas	tabla PER	diagrama jerárquico de tareas
FORMALIZACIÓN E IMPLEMENTACIÓN	modelado de estructuras en CLIPS	tabla CAV codificada	estructuras de CLIPS
	transformación de reglas conceptuales en reglas CLIPS	<ul style="list-style-type: none"> grafo causal estructuras de CLIPS 	reglas en sintaxis de CLIPS
	armado de contexto CLIPS	<ul style="list-style-type: none"> tabla de dependencia entre estados estructuras de CLIPS reglas en sintaxis de CLIPS diagrama jerárquico de tareas 	Sistemas basados en conocimiento en CLIPS

4. Prueba de Concepto

Para ilustrar el “Modelo de Proceso para Uso Didáctico en la Enseñanza de Ingeniería del Conocimiento” propuesto se presenta en esta sección una prueba de concepto sobre el dominio de categorización en el programa de incentivos docentes que se describe en el siguiente texto (“documento normativo”):

“..Se asignará Categoría I, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan desarrollado una amplia producción científica, artística o tecnológica, de originalidad y jerarquía reconocida, acreditada a través del desarrollo de nuevas tecnologías, patentes, libros, artículos publicados en revistas de amplio reconocimiento, preferentemente indexadas, invitaciones como conferencistas a reuniones científicas de nivel internacional, participación con obras de arte en eventos internacionales reconocidos y otras distinciones de magnitud equivalente, y II. Que hayan acreditado capacidad de dirección de grupos de trabajo de relevancia, y III. Que hayan formado becarios y/o tesis de doctorado o maestría, investigadores o tecnólogos del más alto nivel, y IV. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado o Adjunto en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.

Se asignará Categoría II, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan demostrado capacidad de ejecutar, dirigir y planificar en forma exitosa proyectos de investigación científica o de desarrollo tecnológico, acreditada a través de publicaciones o desarrollos de tecnología. En el caso de proyectos artísticos, tal capacidad se acreditará mediante obras de arte originales presentadas en ámbitos nacionales o internacionales reconocidos, y II. Que hayan contribuido a la formación de becarios y/o tesis de doctorado o maestría, investigadores o tecnólogos del más alto nivel, y III. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado o Adjunto en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.

Se asignará Categoría III a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan realizado una labor de investigación científica, artística o de desarrollo tecnológico, debidamente documentada y que acrediten haber dirigido o coordinado exitosamente proyectos de investigación científica, artística o de desarrollo tecnológico, evaluados por entidades de prestigio científico o académico reconocido o que presenten destacados antecedentes en el área disciplinar en la que pretenden categorizar, y II. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado, Adjunto o Jefe de Trabajos Prácticos en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.

Se asignará Categoría IV, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan realizado una destacada labor de investigación científica, artística o de desarrollo tecnológico, bajo la guía o supervisión de un docente investigador I, II, o III o equivalente, durante TRES (3) años como mínimo, y II. Que como docentes hayan alcanzado el cargo de Jefe de Trabajos Prácticos, Ayudante de Primera o equivalente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.

Se asignará Categoría V, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan participado, al menos UN (1) año, en un proyecto de investigación acreditado por la universidad u otro organismo de investigación reconocido a nivel nacional o internacional, y II. Que sean graduados universitarios, y III. Que como docentes hubieran alcanzado la categoría de Ayudante de Primera o equivalente...”

A partir del “documento normativo” precedentemente, la tarea “Identificación de Estados” genera como producto el “listado de estados” que se muestra en la Figura 2. Tomando el “documento normativo” y el “listado de estados” la tarea “análisis de dependencias entre estados” genera como producto la “tabla de dependencia entre estados” que se presenta en la Figura 3.

- | |
|---|
| <ul style="list-style-type: none"> ▪ Categoría I ▪ Categoría II ▪ Categoría III ▪ Categoría IV ▪ Categoría V |
|---|

Estado	Categoría 1	Categoría 2	Categoría 3	Categoría 4	Categoría 5
Categoría 1	NO	NO	NO	NO	NO
Categoría 2	NO	NO	NO	NO	NO
Categoría 3	NO	NO	NO	NO	NO
Categoría 4	NO	NO	NO	NO	NO
Categoría 5	NO	NO	NO	NO	NO

Fig. 2. Listado de Estados

Fig. 3. Tabla de dependencia entre estados

Con los insumos “documento normativo” y la “tabla de dependencia entre estados” la tarea “obtención de criterios para la clasificación de un estado” obtiene como salida los “criterios para la clasificación del estado i” que se presenta en la Figura 4.

CRITERIOS PARA LA CLASIFICACIÓN DE DOCENTES-INVESTIGADORES	
Se asignará Categoría I, a los docentes-investigadores que reúnan las siguientes condiciones:	
I. Que hayan desarrollado una amplia producción científica, artística o tecnológica, de originalidad y jerarquía reconocida, acreditada a través del desarrollo de nuevas tecnologías, patentes, libros, artículos publicados en revistas de amplio reconocimiento, preferentemente indexadas, invitaciones como conferencistas a reuniones científicas de nivel internacional, participación con obras de arte en eventos internacionales reconocidos y otras distinciones de magnitud equivalente, y	
II. Que hayan acreditado capacidad de dirección de grupos de trabajo de relevancia, y	
III. Que hayan formado becarios y/o tesis de doctorado o maestría, investigadores o tecnólogos del más alto nivel, y	
IV. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado o Adjunto en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.	
Se asignará Categoría II, a los docentes-investigadores que reúnan las siguientes condiciones:	
I. Que hayan demostrado capacidad de ejecutar, dirigir y planificar en forma exitosa proyectos de investigación científica o de desarrollo tecnológico, acreditada a través de publicaciones o desarrollos de tecnología. En el caso de proyectos artísticos, tal capacidad se acreditará mediante obras de arte originales presentadas en ámbitos nacionales o internacionales reconocidos, y	
II. Que hayan contribuido a la formación de becarios y/o tesis de doctorado o maestría, investigadores o tecnólogos del más alto nivel, y	
III. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado o Adjunto en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.	
Se asignará Categoría III a los docentes-investigadores que reúnan las siguientes condiciones:	
I. Que hayan realizado una labor de investigación científica, artística o de desarrollo tecnológico, debidamente documentada y que acrediten haber dirigido o coordinado exitosamente proyectos de investigación científica, artística o de desarrollo tecnológico, evaluados por entidades de prestigio científico o académico reconocido o que presenten destacados antecedentes en el área disciplinar en la que pretenden categorizar, y	
II. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado, Adjunto o Jefe de Trabajos Prácticos en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.	
Se asignará Categoría IV, a los docentes-investigadores que reúnan las siguientes condiciones:	
I. Que hayan realizado una destacada labor de investigación científica, artística o de desarrollo tecnológico, bajo la guía o supervisión de un docente-investigador I, II, o III o equivalente, durante TRES (3) años como mínimo, y	
II. Que como docentes hayan alcanzado el cargo de Jefe de Trabajos Prácticos, Ayudante de Primera o equivalente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria.	
Se asignará Categoría V, a los docentes-investigadores que reúnan las siguientes condiciones:	
I. Que hayan participado, al menos UN (1) año, en un proyecto de investigación acreditado por la universidad u otro organismo de investigación reconocida a nivel nacional o internacional, y	
II. Que sean graduados universitarios, y	
III. Que como docentes hubieran alcanzado la categoría de Ayudante de Primera o equivalente.	

Fig. 4. Criterios para la clasificación del estado

Con el insumo “criterios para la clasificación del estado i”, la tarea “identificación de conceptos atributos y valores” genera el producto “tabla CAV” que se presenta en la Figura 5. Partiendo de la “tabla CAV” la tarea “codificación de conceptos atributos y valores” genera la salida “tabla CAV codificada” que se describe en la Figura 6.

Concepto	Atributo	Valor
Docente- Investigador	Categoría	I, II, III, IV, V
	Producción Científica, Artística o Tecnológica acreditada	SI / NO
	Dirección de grupos de trabajo de relevancia	SI / NO
	Formación de becarios y/o tesis, investigadores o tecnólogos del más alto nivel	SI / NO
	Contribuido a la formación de becarios y/o tesis, investigadores o tecnólogos del más alto nivel	SI / NO
	Puesto en la universidad que lo presenta	Titular, Asociado, Adjunto, Jefe de Trabajos Prácticos, Ayudante de Primera, Interino
	Antigüedad en el puesto	Número [0..N]
	Capacidad acreditada de ejecutar, dirigir y planificar proyectos de investigación científica, de desarrollo tecnológico o artísticos	SI / NO
	Labor de investigación científica, artística o de desarrollo tecnológico	SI / NO
	Dirección o coordinación de proyectos de investigación científica, artística o de desarrollo tecnológico	SI / NO
	Años de duración de proyecto de investigación guiado por docente investigador I, II, III	Número [0..N]
	Años de participación en proyectos de investigación acreditados	Número [0..N]
	Graduado Universitario	SI / NO

Fig. 5. Tabla CAV

Concepto	Codif. Concepto	Atributo	Codif. Atributo	Valor	Codif. Valor
Docente- Investigador	DOCINV	Categoría	CAT	I II III IV V	1 2 3 4 5
		Producción Científica, Artística o Tecnológica acreditada	PROD	SI NO	S N
		Dirección de grupos de trabajo de relevancia	DIRG	SI NO	S N
		Formación de becarios y/o tesis, investigadores o tecnólogos del más alto nivel	FORMT	SI NO	S N
		Contribuido a la formación de becarios y/o tesis, investigadores o tecnólogos del más alto nivel	CONFT	SI NO	S N
		Puesto en la universidad que lo presenta	PLIES	Titular Asociado Adjunto Jefe de Trabajos Prácticos Ayudante de Primera Interino	TIT ASOC ADJ JTP AYUDP INTER
		Antigüedad en el puesto	ANT	Número [0..N]	Número [0..N]
		Capacidad acreditada de ejecutar, dirigir y planificar proyectos de investigación científica, de desarrollo tecnológico o artísticos	EDYP	SI NO	S N
		Labor de investigación científica, artística o de desarrollo tecnológico	INV	SI NO	S N
		Dirección o coordinación de proyectos de investigación científica, artística o de desarrollo tecnológico	DCP	SI NO	S N
		Años de duración de proyecto de investigación guiado por docente investigador I, II, III	DURI	Número [0..N]	Número [0..N]
		Años de participación en proyectos de investigación acreditados	PARPI	Número [0..N]	Número [0..N]
		Graduado Universitario	GRAD	SI NO	S N

Fig. 6. Tabla CAV codificada

Con los insumos “criterios para la clasificación del estado i” y “tabla CAV codificada”, la tarea “identificación de reglas” genera el producto “Tabla PER” que se describe en la Figura 7.

IDENTIFICADOR	ASIGNACIÓN DE CATEGORÍA 1
PALABRAS DEL EXPERTO	"Se asignará Categoría I, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan desarrollado una amplia producción científica, artística o tecnológica, de originalidad y jerarquía reconocida, acreditada a través del desarrollo de nuevas tecnologías, patentes, libros, artículos publicados en revistas de amplio reconocimiento, preferentemente indexadas, invitaciones como conferencistas a reuniones científicas de nivel internacional, participación con obras de arte en eventos internacionales reconocidos y otras distinciones de magnitud equivalente, y II. Que hayan acreditado capacidad de dirección de grupos de trabajo de relevancia, y III. Que hayan formado becarios y/o tesis de doctorado o maestría, investigadores o tecnólogos del más alto nivel, y IV. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado o Adjunto en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria."
REGLA	SI (DOCINV.PROD = S Y DOCINV.DIRG = S Y DOCINV.FORMT = S Y (DOCINV.PUES = TIT O DOCINV.PUES = ASOC O DOCINV.PUES = ADJ O (DOCINV.PUES = INTER Y DOCINV.ANT >= 3))) ENTONCES DOCINV.CAT = 1
IDENTIFICADOR	ASIGNACIÓN DE CATEGORÍA 2
PALABRAS DEL EXPERTO	"Se asignará Categoría II, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan demostrado capacidad de ejecutar, dirigir y planificar en forma exitosa proyectos de investigación científica o de desarrollo tecnológico, acreditada a través de publicaciones o desarrollos de tecnología. En el caso de proyectos artísticos, tal capacidad se acreditará mediante obras de arte originales presentadas en ámbitos nacionales o internacionales reconocidos, y II. Que hayan contribuido a la formación de becarios y/o tesis de doctorado o maestría, investigadores o tecnólogos del más alto nivel, y III. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado o Adjunto en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria."
REGLA	SI (DOCINV.EDYP = S Y DOCINV.CONFT = S Y (DOCINV.PUES = TIT O DOCINV.PUES = ASOC O DOCINV.PUES = ADJ O (DOCINV.PUES = INTER Y DOCINV.ANT >= 3))) ENTONCES DOCINV.CAT = 2
IDENTIFICADOR	ASIGNACIÓN DE CATEGORÍA 3
PALABRAS DEL EXPERTO	"Se asignará Categoría III a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan realizado una labor de investigación científica, artística o de desarrollo tecnológico, debidamente documentada y que acrediten haber dirigido o coordinado exitosamente proyectos de investigación científica, artística o de desarrollo tecnológico, evaluados por entidades de prestigio científico o académico reconocido o que presenten destacados antecedentes en el área disciplinar en la que pretenden categorizar, y II. Que como docentes hayan alcanzado la categoría de profesor Titular, Asociado, Adjunto o Jefe de Trabajos Prácticos en la universidad que los presente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria."
REGLA	SI (DOCINV.INV = S Y DOCINV.DCP = S Y (DOCINV.PUES = TIT O DOCINV.PUES = ASOC O DOCINV.PUES = ADJ O DOCINV.PUES = JTP O (DOCINV.PUES = INTER Y DOCINV.ANT >= 3))) ENTONCES DOCINV.CAT = 3
IDENTIFICADOR	ASIGNACIÓN DE CATEGORÍA 4
PALABRAS DEL EXPERTO	"Se asignará Categoría IV, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan realizado una destacada labor de investigación científica, artística o de desarrollo tecnológico, bajo la guía o supervisión de un docente investigador I, II, o III o equivalente, durante TRES (3) años como mínimo, y II. Que como docentes hayan alcanzado el cargo de Jefe de Trabajos Prácticos, Ayudante de Primera o equivalente. En el caso de que sean interinos, se requerirá TRES (3) años de antigüedad mínima en la docencia universitaria."
REGLA	SI (DOCINV.INV = S Y DOCINV.DURI >= 3 Y (DOCINV.PUES = JTP O DOCINV.PUES = AYUDP O (DOCINV.PUES = INTER Y DOCINV.ANT >= 3))) ENTONCES DOCINV.CAT = 4
IDENTIFICADOR	ASIGNACIÓN DE CATEGORÍA 5
PALABRAS DEL EXPERTO	"Se asignará Categoría V, a los docentes-investigadores que reúnan las siguientes condiciones: I. Que hayan participado, al menos UN (1) año, en un proyecto de investigación acreditado por la universidad u otro organismo de investigación reconocido a nivel nacional o internacional, y II. Que sean graduados universitarios, y III. Que como docentes hubieran alcanzado la categoría de Ayudante de Primera o equivalente."
REGLA	SI (DOCINV.PARPI >= 1 Y DOCINV.GRAD = S Y DOCINV.PUES = AYUDP) ENTONCES DOCINV.CAT = 5

Fig. 7. Tablas PER

La tarea “modelado conceptual de reglas” toma como insumo a las “tablas PER” y produce el “grafo causal” que se muestra en la Figura 8. Tomando como insumo las

“tablas PER”, la tarea “identificación de encadenamiento de tareas”, genera como producto el “diagrama jerárquico de tareas” que se muestra en la Figura 9.

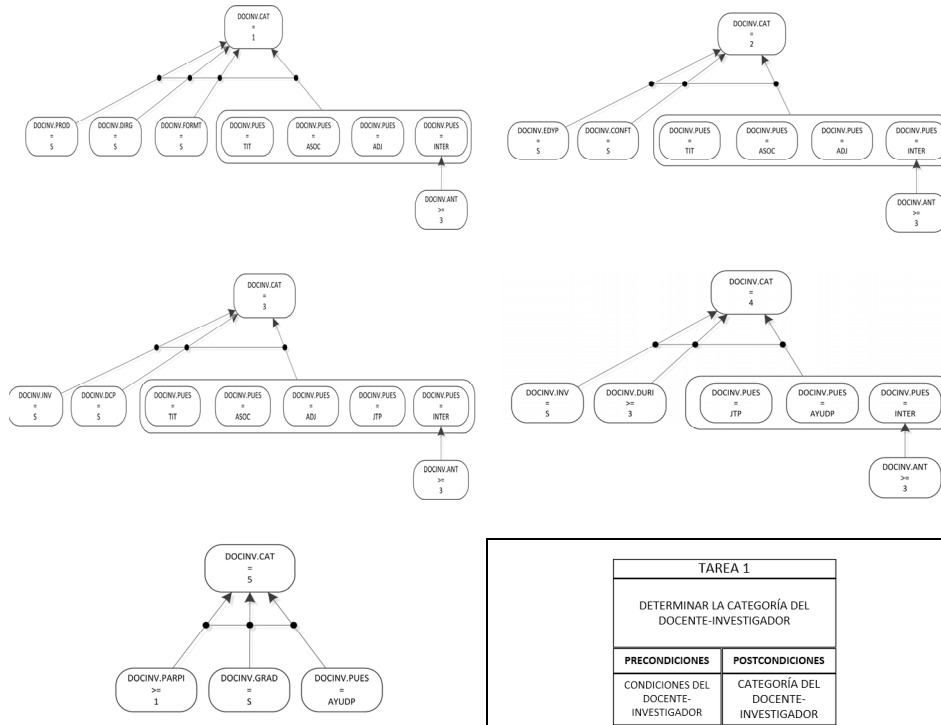


Fig. 8. Grafo Causal

Fig. 9. Diagrama Jerarquico de Tareas

A partir de la “tabla CAV codificada”, la tarea “modelado de estructuras en CLIPS” genera como producto las “estructuras de CLIPS” que se muestra en la Figura 10. Tomando el “grafo causal” y las “estructuras de CLIPS” la tarea “transformación de reglas conceptuales en reglas CLIPS” genera como producto la “reglas en sintaxis de CLIPS” que se presenta en la Figura 11.

```
(deftemplate docinv
  (slot cat (type INTEGER))
  (slot prod (type SYMBOL))
  (slot dirg (type SYMBOL))
  (slot formit (type SYMBOL))
  (slot conff (type SYMBOL))
  (slot pues (type SYMBOL))
  (slot ant (type INTEGER))
  (slot edyp (type SYMBOL))
  (slot inv (type SYMBOL))
  (slot durj (type INTEGER))
  (slot parpi (type INTEGER))
  (slot grad (type SYMBOL))
  (slot grad (type SYMBOL))
)
```

Fig. 10. Estructuras de CLIPS

```
Regla CAT1:
(defrule CAT1
  ?docente <- (docinv (prod S)(dirg S)(pues ?pues)(ant ?ant))
  (test (or
    (eq ?pues TIT)
    (eq ?pues ASOC)
    (eq ?pues ADJ)
    (eq ?pues JTP)
    (and (eq ?pues INTER)>= ?ant 3)
  ))
) => (printout t "El Docente-Investigador es Categoría 1." crlf)
(modify ?docente (cat 1))

Regla CAT2:
(defrule CAT2
  ?docente <- (docinv (edyp S)(conff S)(pues ?pues)(ant ?ant))
  (test (or
    (eq ?pues TIT)
    (eq ?pues ASOC)
    (eq ?pues ADJ)
    (and (eq ?pues INTER)>= ?ant 3)
  ))
) => (printout t "El Docente-Investigador es Categoría 2." crlf)
(modify ?docente (cat 2))

Regla CAT3:
(defrule CAT3
  ?docente <- (docinv (inv S)(dcp S)(pues ?pues)(ant ?ant))
  (test (or
    (eq ?pues TIT)
    (eq ?pues ASOC)
    (eq ?pues ADJ)
    (eq ?pues JTP)
    (eq ?pues AYUDP)
    (and (eq ?pues INTER)>= ?ant 3)
  ))
) => (printout t "El Docente-Investigador es Categoría 3." crlf)
(modify ?docente (cat 3))

Regla CAT4:
(defrule CAT4
  ?docente <- (docinv (inv S)(durj ?durj)(pues ?pues)(ant ?ant))
  (test (>= ?durj 3))
  (test (or
    (eq ?pues JTP)
    (eq ?pues AYUDP)
    (and (eq ?pues INTER)>= ?ant 3)
  ))
) => (printout t "El Docente-Investigador es Categoría 4." crlf)
(modify ?docente (cat 4))

Regla CAT5:
(defrule CAT5
  ?docente <- (docinv (parpi ?parpi)(grad S)(pues AYUDP))
  (test (>= ?parpi 1))
) => (printout t "El Docente-Investigador es Categoría 5." crlf)
(modify ?docente (ca))
```

Fig. 11. Reglas en sintaxis de CLIPS

A partir de la “tabla de dependencia entre estados”, “estructuras de CLIPS”, “reglas en sintaxis de CLIPS” y “diagrama jerárquico de tareas”, la tarea “armado de contexto CLIPS” genera como producto las “sistema basado en conocimiento en CLIPS” que se muestra en la Figura 12.

<pre>(slot cat (type INTEGER)) (slot prod (type SYMBOL)) (slot dirg (type SYMBOL)) (slot formt (type SYMBOL)) (slot conft (type SYMBOL)) (slot pues (type SYMBOL)) (slot ant (type INTEGER)) (slot edyp (type SYMBOL)) (slot inv (type SYMBOL)) (slot dcp (type SYMBOL)) (slot duri (type INTEGER)) (slot parpi (type INTEGER)) (slot grad (type SYMBOL)) (deffacts CAT1 (revisarCAT1)) (defrule CAT1 (declare (salience 100)) ?borrar <- (revisarCAT1) ?docente <- (docinv (prod S)(dirg S)(formt S) (pues ?pues)(ant ?ant)) (test (or (eq ?pues TIT) (eq ?pues ASOC) (eq ?pues ADJ) (and (eq ?pues INTER)>= ?ant 3))) => (printout t "El Docente-Investigador es Categoría 1." crff) (modify ?docente (cat 1)) (retract ?borrar) s))</pre>	<pre>(deffacts CAT2 (revisarCAT2)) (defrule CAT2 (declare (salience 100)) ?borrar <- (revisarCAT2) ?docente <- (docinv (edyp S)(conft S)(pues ?pues)(ant ?ant)) (test (or (eq ?pues TIT) (eq ?pues ASOC) (eq ?pues ADJ) (and (eq ?pues INTER)>= ?ant 3))) => (printout t "El Docente-Investigador es Categoría 2." crff) (modify ?docente (cat 2)) (retract ?borrar))) (deffacts CAT3 (revisarCAT3)) (defrule CAT3 (declare (salience 100)) ?borrar <- (revisarCAT3) ?docente <- (docinv (inv S)(dcp S)(pues ?pues)(ant ?ant)) (test (or (eq ?pues TIT) (eq ?pues ASOC) (eq ?pues ADJ) (eq ?pues JTP) (and (eq ?pues INTER)>= ?ant 3)))))</pre>	<pre>=> (printout t "El Docente-Investigador es Categoría 3." crff) (modify ?docente (cat 3)) (retract ?borrar))) (deffacts CAT4 (revisarCAT4)) (defrule CAT4 (declare (salience 100)) ?borrar <- (revisarCAT4) ?docente <- (docinv (inv S)(duri ?duri)(pues ?pues)(ant ?ant)) (test (>= ?duri 3)) (test (or (eq ?pues JTP) (eq ?pues AYUDP) (and (eq ?pues INTER)>= ?ant 3)))) => (printout t "El Docente-Investigador es Categoría 4." crff) (modify ?docente (cat 4)) (retract ?borrar))) (deffacts CAT5 (revisarCAT5)) (defrule CAT5 (declare (salience 100)) ?borrar <- (revisarCAT5) ?docente <- (docinv (parpi ?parpi)(grad S)(pues AYUDP)) (test (>= ?parpi 1)) => (printout t "El Docente-Investigador es Categoría 5." crff) (modify ?docente (cat 5))))</pre>
---	---	---

Fig. 12. Sistema basado en conocimiento en CLIPS

5. Conclusiones

En este trabajo se presenta un modelo de proceso para uso didáctico en la enseñanza de la construcción de sistemas basados en conocimiento (SBC). El modelo de proceso presentado se articula con los formalismos de modelado de conocimiento que se enseñan en la asignatura “Sistemas Basados en Conocimiento” de la Licenciatura en Sistemas de la Universidad Nacional de Lanús.

Se podría considerar una limitante del modelo de proceso propuesto la restricción de usar el ambiente CLIPS para el desarrollo de SBC, sin embargo considerando que este es un ambiente de desarrollo “de facto” impuesto por DARPA, los ambientes de desarrollo alternos tienen funcionalidades similares.

El modelo de proceso propuesto se utilizara por primera vez en el segundo semestre del año académico 2016 de la asignatura “Sistemas Basados en Conocimiento”, por lo que se espera formular conclusiones preliminares sobre el uso del modelo y su integración a los contenidos de la asignatura a partir del año 2017.

Financiamiento. Las investigaciones que se reportan en este capítulo han sido financiadas por el Proyecto de Investigación 33B170 de la Secretaria de Ciencia y Técnica de la Universidad Nacional de Lanús (Argentina).

Referencias

- García Martínez, R. Y Britos, P. 2004. *Ingeniería de Sistemas Expertos*. 649 páginas. Editorial Nueva Librería. ISBN 987-1104-15-4.
- García-Martínez, R., Martins, S., Merlino, H., Amatriain, H., Ribeiro, F., Bianco, S. 2016. *Propuesta de Articulación de Temas de Sistemas Inteligentes n la Currícula de Licenciatura en Sistemas*. Proceedings del XI Congreso de Tecnología en Educación y Educación en Tecnología. Pág. 48-56. ISBN 978-987-3977-30-5.
- García-Martínez, R., Servente, M. y Pasquini, D. 2003. *Sistemas Inteligentes*. 347 páginas. Editorial Nueva Librería. ISBN 987-1104-05-7.
- Gómez-Pérez, A., Juristo, N., Montes, C., & Pazos, J. 1997. *Ingeniería del Conocimiento: Diseño y Construcción de Sistemas Expertos*. Ceuta, Madrid, Spain.
- Rodríguez, D. 2015. *Conceptualización de Espacios Virtuales de Trabajo*. Tesis Doctoral en Ciencias informáticas. Facultad de Informática. Universidad Nacional de La Plata.
- Brulé, J., Blount, A. 1989. Knowledge acquisition. McGraw-Hill, Inc.
- Carrillo Verdun, J. 1987. Metodología para el Desarrollo de Sistemas Expertos. Tesis Doctoral. Universidad Politécnica de Madrid.
- Grover, M. 1983. A Pragmatic Knowledge Acquisition Methodology. IJCAI Vol. 83: 436-438.
- Schreiber, G. 2000. Knowledge engineering and management: the CommonKADS methodology. MIT press.

Conjunto de Herramientas de Ayuda a la Presencialidad

Darío Rodríguez, Sebastian Martins, Ezequiel Baldizzoni, Santiago Bianco,
Ramón García-Martínez

Laboratorio de Ingeniería de Espacios Virtuales de Trabajo
Grupo de Investigación en Sistemas de Información.
Departamento de Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús
darodriguez@unla.edu.ar, rgm1960@yahoo.com

Resumen. La estrategia de virtualización de contenidos en apoyo a la presencialidad busca coadyuvar en la retención de los estudiantes universitarios. En este trabajo se presentan tres herramientas que, desarrolladas para educación a distancia pueden ser utilizadas como ayudas a la presencialidad: (i) Espacio Virtual de Trabajo Colaborativo Académico (EVTCA), (ii) Ambiente de Desarrollo de Objetos de Aprendizaje (ADOA), y Ambiente de Auto-Evaluación (AAE)..

Palabras Clave. Ayudas a la presencialidad, Espacios Virtuales de Trabajo, Objetos de Aprendizaje, Auto-evaluación.

1. Introducción

El plan de carrera de la Licenciatura en Sistemas de la Universidad Nacional de Lanús (UNLa) prevé el título de pregrado de Analista Programador Universitario. Se observa que para el conjunto de estudiantes, alcanzar esta titulación constituye una meta que los habilita formalmente a ingresar en el mercado laboral de la Industria del Software. Por otra parte, también se advierte que esta búsqueda laboral comienza ni bien se consolidan las competencias específicas desarrolladas en el área de programación (entre 3^{er} y 4^{to} año).

Producido el ingreso al mundo laboral, se les dificulta continuar sus estudios al no poder coordinar los horarios de cursada con los horarios laborales. Experiencias en otras Casas de Estudios con el mismo plan de carrera con titulación de pregrado, sugieren que la estrategia de virtualización de contenidos de apoyo a la presencialidad del estudiante en la Universidad, se estime como una estrategia con un grado de alta efectividad en orden a que continúe con las asignaturas de cuarto y quinto año.

El carácter teórico-práctico que implica el desarrollo en equipo de proyectos informáticos integradores, que la mayoría de las asignaturas de ciclo superior de la Licenciatura en Sistemas establece como requisito de aprobación, requiere de los estudiantes dedicar una cantidad de horas a reuniones para acordar, desarrollar y consolidar la resolución del proyecto que les fuera planteado por la asignatura.

El ingreso al mundo laboral implica también una reducción al mínimo necesario del tiempo dedicado a la asistencia a clases, y una merma en el contacto con los docentes

en los tiempos dedicados a consultas sobre aspectos teóricos o resolución de ejercicios de la asignatura.

En este contexto, se delimita el problema (sección 2), se propone una solución (sección 3) y se presentan algunas conclusiones preliminares (sección 4).

2. Delimitación del Problema

El desarrollo de un proyecto conlleva la interacción entre las personas que conforman el equipo de trabajo y un supervisor que los guía y controla. En el marco de una carrera de grado, un proyecto de desarrollo es una actividad práctica dentro de una asignatura, en la cual el equipo de trabajo está conformado por un grupo de estudiantes y un docente (o varios) que oficia de supervisor o tutor. [Garbarini et al., 2012] Teniendo en cuenta la carga horaria correspondiente al cursado de las asignaturas y a la actividad laboral, los estudiantes se enfrentan con dificultades para compartir un espacio físico donde encontrarse para llevar adelante el proyecto.

Los avances tecnológicos han generado la posibilidad de disponer de nuevos dispositivos de aprendizaje como lo son los Objetos de Aprendizaje (OA) [Brito, 2010; Bergin et al., 2012], que amplían los recursos disponibles y el alcance de los contenidos en las aulas, potenciando las mismas a través de los beneficios inherentes al uso educativo de las TIC. Sin embargo, un docente no informático se enfrenta con la dificultad de carecer de conocimiento de programación, o no tener a su disposición un equipo interdisciplinario que le ayude a desarrollar los objetos de aprendizaje que necesita.

La evaluación viene acompañada con demasiada frecuencia de una importante carga de trabajo para el docente, tanto por el tiempo empleado en el diseño y producción de las evaluaciones, como por el tiempo empleado en la corrección de las mismas, lo que en gran medida lleva a que en la mayoría de las ocasiones la retroalimentación para el proceso de enseñanza-aprendizaje, tanto para el alumno, como para el profesor, llegue de manera tardía.

3. Solución Propuesta

En el marco de las nuevas tecnologías de la información y de las comunicaciones, el Espacio Virtual de Trabajo Colaborativo Académico (EVTCA) abre la posibilidad de disponer de un espacio donde, el encuentro del equipo de trabajo, sea virtual y mediado por tecnología, permitiendo mejorar la productividad del grupo ya que pueden agendar trabajo colaborativo desde espacios discontinuos física y temporalmente; permitiéndole además al docente, documentar la trazabilidad de las interacciones de los estudiantes. En la sección 3.1. se presenta el EVTCA desarrollado.

El Ambiente de Desarrollo de Objetos de Aprendizaje (ADOA) tiene como objetivo satisfacer la necesidad que el docente desarrolle Objetos de Aprendizaje ocupándose de los aspectos didácticos y de contenidos y prescindiendo de los aspectos informáticos. El artefacto software que desarrolla OA integra las mejores prácticas de la ingeniería de software al desarrollo de objetos de aprendizaje, enfocando el

desarrollo del proyecto en las características relevantes para la consecución de los conceptos de reusabilidad, simplicidad, amigabilidad y accesibilidad, brindando adicionalmente una visualización e interactividad pensada y adaptada a la simplicidad del usuario. En la sección 3.2. se describe el ADOA desarrollado.

La auto-evaluación entendida como un proceso integrado al proceso de aprendizaje se propone servir de retroalimentación a este, constituyéndose en una de las etapas fundamentales para su mejoramiento continuo. Poder contar con una herramienta que no solo le sirva al docente sino que también le ayude al alumno a tener una referencia del alcance de sus conocimientos mediante la auto-evaluación es el objetivo general que busca cumplir el Ambiente de Auto-Evaluación (AAE) que se presenta en la sección 3.3.

3.1. Espacio Virtual de Trabajo Colaborativo Académico (EVTCA)

El Espacio Virtual de Trabajo Colaborativo Académico [Rodríguez et al., 2015] es un producto software generado a partir de la interacción e integración del trabajo de un grupo de estudiantes y de docentes-investigadores de la Licenciatura en Sistemas. El objetivo fue desarrollar un artefacto software basado en tecnología web que permita dar solución a la necesidad de proveer un espacio de trabajo sincrónico en tiempo pero discontinuo en espacio. Este espacio puede ser utilizado por estudiantes que por el desarrollo de su actividad laboral se les dificultan las reuniones presenciales de trabajo de equipo en asignaturas que a partir del tercer año de la carrera lo requieren. La solución propuesta es una herramienta orientada a favorecer la comunicación e interacción de los miembros de un equipo de desarrollo, los cuales utilizan la misma para llevar a delante un proyecto software. Adicionalmente, el sistema permite realizar controles sobre los productos intermedios del proyecto y su progreso, brindando además la posibilidad de evaluar las interacciones y participaciones de los miembros del equipo durante el proceso.

El EVTCA debe mediar las actividades en equipo de docentes, de estudiantes, y de evaluación y seguimiento de los equipos de estudiantes realizado por los docentes. En consecuencia el artefacto software que soporte el espacio virtual de trabajo colaborativo debe proveer los siguientes elementos de trabajo:

- Pizarra virtual colaborativa
- Elementos de comunicación a través de audio/video y chat
- Repositorio de documentos de producción propia con control de versiones
- Elementos de control de acceso de usuario con diferentes privilegios
- Facilidades de creación de grupos de trabajo temporales (conferencias)
- Correo interno, con reporte de intercambios
- Mecanismos de awareness

En la Figura 1 se muestra el prototipo del EVTCA en el que los estudiantes hacen uso concurrente de la pizarra, intercambian opiniones a través de videoconferencia y se pasan documentos de trabajo mediante el email interno [Rodríguez et al., 2015].

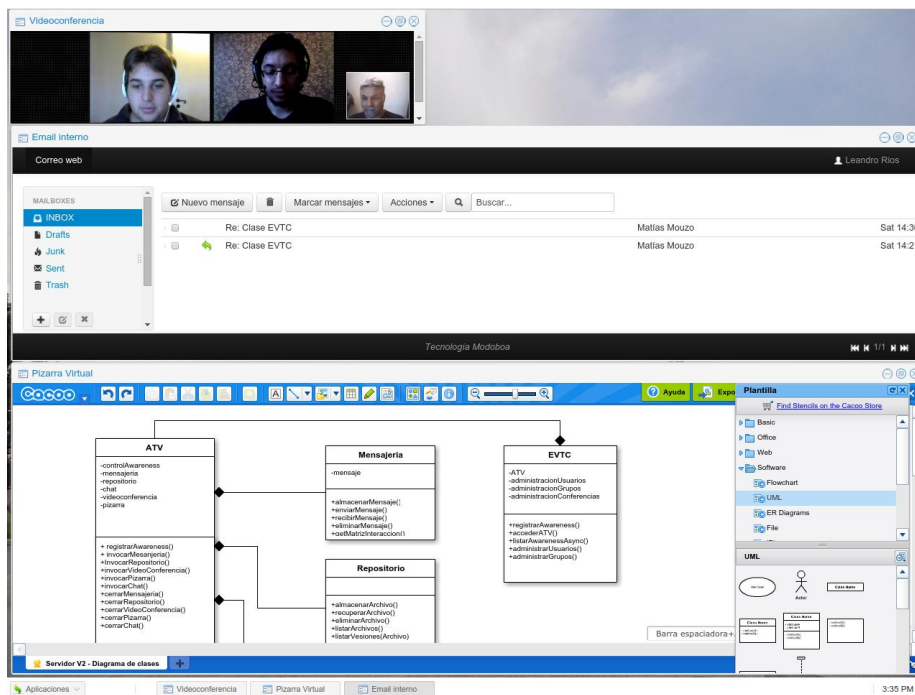


Fig. 1. Caso Piloto: Uso Concurrente de Pizarra, Videoconferencia y Email Interno [Rodríguez et al., 2015]

3.2. Ambiente de Desarrollo de Objetos de Aprendizaje (ADOA)

El ambiente ADOA [Martins et al., 2015] provee un ambiente destinado a facilitar la creación de objetos de aprendizaje reutilizables, independientes y dinámicos, los cuales puedan ser integrados a distintos Entornos Virtuales de Enseñanza y Aprendizaje (EVEA), adaptándose de manera automática a las distintas tecnologías y dispositivos existentes. Dichos objetivos son alcanzados a través de la implementación de distintos estándares y especificaciones ampliamente validados en la industria del software.

ADOA se centra principalmente en el concepto de patrón, entendiendo a esta como la resolución comprobada de un problema, la cual puede ser aplicada a otros problemas de características similares (pero en circunstancias distintas) para dar solución al mismo, garantizando la utilización de las mejores prácticas conocidas para solventar una situación determinada favoreciéndose de la experiencia de la comunidad.

Si bien ADOA es construido considerando ambos tipos de patrones, es relevante destacar la importancia en la implementación de los patrones pedagógicos con el fin de solventar distintas dificultades previamente señaladas como el alto costo, tiempo y complejidad para la creación de objetos de aprendizaje, siendo este una de las principales carencias identificadas en la sección previa.

Sobre el concepto de patrón previamente descrito se construye al ambiente, entendiendo esta idea como la principal vía de reutilización de los elementos didácticos.

ADOA permite la generación de un objeto de aprendizaje en 6 pasos generales, simplificando el proceso de edición y/o generación del mismo, reduciendo además la carga cognitiva requerida por el usuario. Los pasos requeridos son:

- Paso 1: Seleccionar el patrón de diseño pedagógico a utilizar.
- Paso 2: Cargar la introducción al tema a tratar.
- Paso 3: Cargar el contenido del tema, cuya estructura varia acorde al patrón seleccionado en el primer paso.
- Paso 4: Seleccionar el tipo de actividad mediante la cual se le permitirá al alumno evaluar su proceso de aprendizaje y generar la misma.
- Paso 5: Generar la evaluación.
- Paso 6: Exportar el proyecto.

En el primer caso se incorporó a la Asignatura Ingeniería del Software 1 (perteneciente al segundo año de la Carrera) el OA como elemento de revisión y consolidación de los temas abordados en la guía de estudio y práctica sobre análisis y diseño de software estructurado. El objetivo principal de este caso es proveer al alumno de un mecanismo que le permita vincular los conceptos teóricos desarrollados en los materiales de lectura, con los ejercicios prácticos realizados, proveyendo un mismo medio para auto-evaluar el grado de apropiación de los contenidos.

En la figura 2 se ilustra la visualización del primer contenido a apropiar por el alumno (la comprensión y construcción del diagrama de contexto).

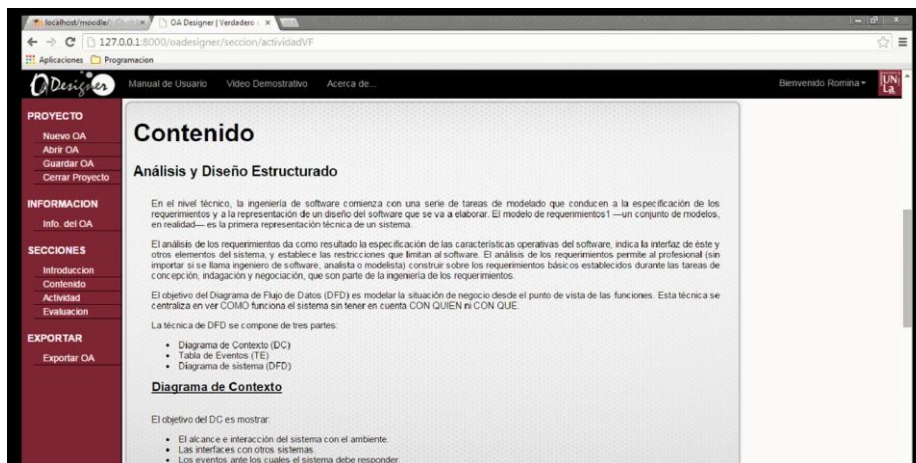


Fig. 2. Caso de Uso Ingeniería del Software 1 - Contenido [Martins et al., 2015]

Posteriormente, a partir del tipo de actividades existentes para la práctica y evaluación de dicho contenido, se selecciona “identificación” como tipo de actividad, mediante la cual se requiere al alumno identificar y comprender una serie de conceptos a partir de un diagrama de contexto abstracto presentado como consigna (figura 3).

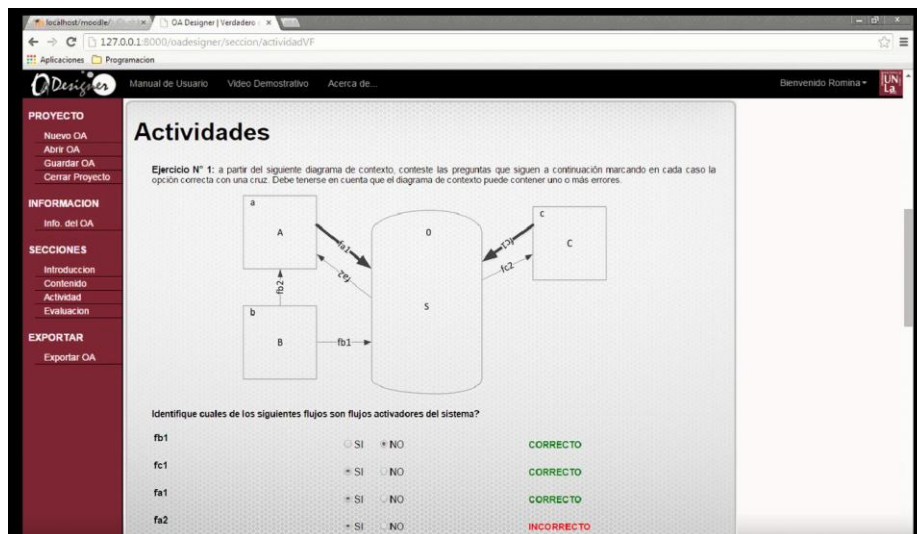


Fig. 3. Caso Asignatura Ingeniería del Software 1 - Actividades [Martins et al., 2015]

3.3. Ambiente de Auto-Evaluación (AAE)

El ambiente AAE [Khusnulina y Zavaglia, 2016] administra el proceso de creación, aplicación y calificación de evaluaciones virtuales de tipo desarrollo y selección de respuesta (verdadero/falso, opción múltiple y unión de conceptos), a partir de una base de datos de preguntas, así como el seguimiento de las preguntas a fin de corregir o actualizar las que así lo requieran y eliminar aquellas que ya no sean requeridas. En la figura se muestra un ejemplo de evaluación parcial gestionada por la herramienta.

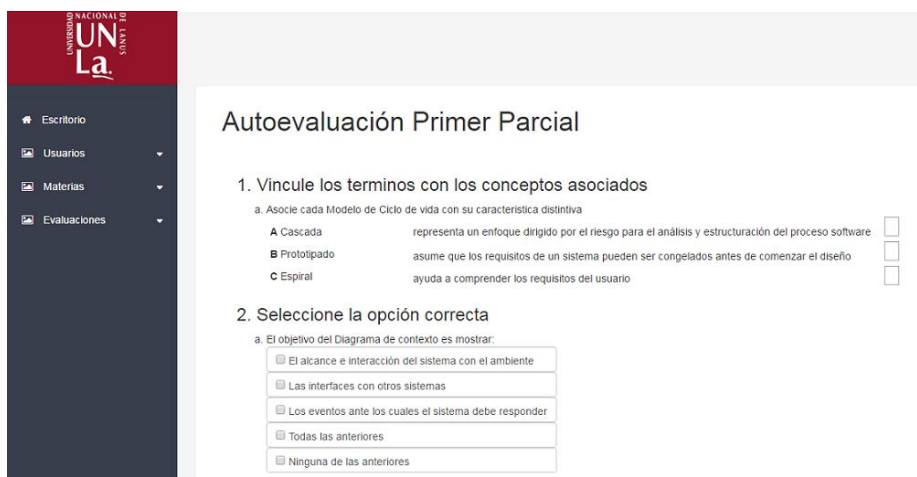


Fig. 4. Ejemplo de evaluación parcial

4. Conclusiones

Se ha presentado un conjunto de herramientas de ayuda a la presencialidad que están siendo desarrolladas en el Laboratorio de Ingeniería de Espacios Virtuales de Trabajo del Grupo de Investigación en Sistemas de Información de la Universidad Nacional de Lanús. Estas son: (i) el Espacio Virtual de Trabajo Colaborativo Académico (EVTCA), (ii) el Ambiente de Desarrollo de Objetos de Aprendizaje (ADOA), y el Ambiente de Auto-Evaluación (AAE). Las mismas se encuentran en etapa de prototipo y se está trabajando para ponerlas en producción.

Se ha descrito un espacio virtual de trabajo colaborativo que provee un conjunto de servicios integrados (videoconferencia, pizarra virtual colaborativa, gestión de archivos, mail interno, entre otros) los cuales posibilitan el trabajo a distancia, y el seguimiento del mismo por parte de los docentes.

Se ha presentado un ambiente de trabajo para la generación de objetos de aprendizaje reutilizables y adaptables, mediante la integración de los patrones de diseño pedagógico, estándares y especificaciones, y una visión integral del proceso enfocada a la autosuficiencia del docente para generar OA.

Se ha presentado un ambiente de auto-evaluación que permite a los estudiantes autoevaluarse en sus aprendizajes, facilitando los procesos de corrección y ahorrando a los docentes parte del tiempo dedicados a correcciones de parciales.

Las herramientas desarrolladas buscan favorecer el trabajo en equipo realizado por los estudiantes (EVTCA), y el uso de dispositivos de aprendizaje extra-áulico (ADOA), buscando estimular los procesos de autoevaluación (AAE).

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por el Proyecto de Investigación 33B180 del Departamento de Desarrollo Productivo y Tecnológico de la Universidad Nacional de Lanús.

Referencias

- Bergin, J., Eckstein J., Völter, M., Sipos, M., Wallingford, E., Marquardt, K., Chandler, J., Sharp, H., Lynn Manns, M. 2012. *Pedagogical Patterns: Advice For Educators*. ISBN 1479171824.
- Brito, J. 2010. *Objetos de aprendizaje 2.0, patrones de diseño de OA y recursos educativos abiertos. Una aproximación reflexiva en torno al desarrollo de materiales para la EAD*. Virtualidad, Educación y Ciencia, (1): 09-36. ISSN 1853-6530.
- Garbarini, R., Rodríguez, D., Pollo-Cattaneo, F., García-Martínez, R. 2012. *Elementos para una Arquitectura de Espacio Virtual para Trabajo Colaborativo Orientado a Trabajos Finales Grupales en Carreras de Grado*. Proceedings del XVIII Congreso Argentino de Ciencias de la Computación. Pag. 619-626. ISBN 978-987-1648-34-4.
- Khusnulina, V., Zavaglia, M. 2016. *Ambiente de Auto-Evaluación Virtual*. Trabajo Final de Asignatura Proyecto de Software. Licenciatura en Sistemas. Departamento de Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús

- Martins, S., Rodríguez, D., Baldizzoni, E., Mansilla, R., Weilli, A., Violi, P., Scordamaglia, E., Amatriain, H., García-Martínez, R. 2015. *Prototipo de Ambiente de Desarrollo de Objetos de Aprendizaje (ADOA)*. Proceedings X Congreso de Tecnología en Educación y Educación en Tecnología. Pág. 463-471. ISBN 978-950-656-154-3
- Rodríguez, D., Martins, S., Baldizzoni, E., Ríos, R., Mouzo, M., Penayo, G., García-Martínez, R. 2015. *Espacio Virtual de Trabajo Colaborativo Académico (EVTCA)*. Proceedings X Congreso de Tecnología en Educación y Educación en Tecnología. Pág. 472-479. ISBN 978-950-656-154-3.

Modelo de seguimiento de una actividad de foro en la enseñanza universitaria. Aplicación en un estudio de caso.

Cecilia Sanz¹, Alejandra Zangara¹, and Paula Dieser²

¹ Instituto de Investigación en Informática LIDI (III LIDI), UNLP, La Plata, Argentina

² Facultad de Ciencias Exactas y Naturales, UNLPam, Santa Rosa, Argentina
csanz@lidi.info.unlp.edu.ar, alejandra.zangara@gmail.com,
pauladieser@exactas.unlpam.edu.ar

Resumen El presente trabajo aporta valor a las actividades de producción colectiva y colaborativa en educación superior, los foros en este caso. En estos escenarios existen varios desafíos, didácticos y tecnológicos, que hacen al diseño de la actividad: tipo de foro, consigna, fases, conformación de grupos; mediación; seguimiento y evaluación.

El seguimiento de este tipo de actividades debe apuntar a la construcción de conocimiento y de metacognición. Esto significa que los estudiantes aprendan no solamente sobre lo que construyen, sino de la estrategia que utilizan: cuándo, cómo y con qué medios se comunican durante el proceso.

En este trabajo se presenta un modelo de seguimiento de una actividad de foro, los indicadores (individuales y colectivos) que se utilizan y la estrategia de visualización a disposición de los estudiantes, en el marco de una Maestría de la UNLP. En los resultados se vislumbran algunas de las posibilidades ofrecidas por el modelo.

Keywords: Indicadores de seguimiento de foro, foro en fases de trabajo, foro como resolución de problemas, foro en la enseñanza universitaria

1. Introducción

La interacción entre estudiantes es un componente crítico en los contextos de aprendizaje [1].

En [2] se destaca la importancia de la interacción en los cursos *online*. Citan las investigaciones de varios autores tales como [3] con los que acuerdan que estas interacciones, ya sea entre docentes y alumnos como de alumnos entre sí, son críticas en la efectividad del aprendizaje en línea y afectan positivamente en el aprendizaje de los alumnos. Estos estudios permiten resaltar la importancia de la profundización en la investigación en los foros en los entornos virtuales de enseñanza y aprendizaje, como espacio central para la interacción, y el desarrollo de actividades didácticas que permitan un real aprovechamiento de estas interacciones.

Los foros de debate proveen características únicas de comunicación, que los hacen candidatos perfectos para el manejo de la comunicación en entornos virtuales de enseñanza y aprendizaje. Esto es porque ofrecen comunicación asincrónica y por tanto,

relajan la situación de establecer horarios específicos, además, las interacciones y conocimientos compartidos quedan a disposición dentro del espacio del foro para que sean consultados cuando un participante lo necesite, lo que resulta de interés para mantener un registro de las interacciones compartidas [4]. También en [4], se afirma que los foros pueden ser una actividad que motive a los alumnos, ya que al publicarse en forma frecuente nuevo contenido les resulta atractivo para volver a visitar el espacio de debate, y además, cada vez que un alumno expresa su opinión buscará defenderla frente a la mirada de los compañeros, lo que puede favorecer a que el alumno se vuelva más activo.

Existen varias investigaciones en relación a los foros, sus posibilidades para la interacción y la configuración de diferentes actividades didácticas con ellos. También hay trabajos que avanza sobre las estrategias de seguimiento de una actividad de foro en procesos educativos, algunos en particular en entornos virtuales de enseñanza y aprendizaje [5,6,7,8,9]. La investigación que aquí se presenta se relaciona con este último eje relacionado con el seguimiento y evaluación, en particular se define un modelo de seguimiento de actividades de foros, a partir de la consideración de un conjunto de criterios e indicadores, que luego son aplicados a un estudio de caso.

El artículo se organiza de aquí en adelante de la siguiente manera: en la sección 2 se presentan antecedentes en relación a la propuesta de diversos autores sobre indicadores de seguimiento de las actividades en foros de debate *online*, en la sección 3 se presenta la configuración didáctica del foro con la que se trabaja en esta investigación, en la sección 4 se describe el aporte de este trabajo centrado en un modelo propio de seguimiento de una actividad de foro, en particular considerando la configuración previa explicitada. En la sección 5, se detalla la forma en que han sido aplicados estos indicadores al estudio de caso y los resultados preliminares obtenidos. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro.

2. Antecedentes

En esta sección se recuperan algunos antecedentes de interés en relación a lo que otros autores proponen para el seguimiento de una actividad de foro *online*, y que ha servido de base para la presente investigación.

En [10], se presenta un estado del arte en relación a indicadores que permiten analizar interacciones en procesos educativos. Presentan tres dimensiones a las que pueden corresponder estos indicadores: (a) La dimensión cognitiva: refieren a algún aspecto de las operaciones cognitivas vinculadas al contenido propio de la actividad de aprendizaje que se está analizando, y más en general a las acciones propias de la actividad de aprendizaje; (b) La dimensión social: relacionada con la cooperación o colaboración o sólo las acciones comunicativas que se abordan en un grupo de participantes de una actividad educativa; (c) La dimensión afectiva: relacionada con una situación afectiva de los participantes de la actividad de aprendizaje.

Ya más específicamente, referido a la evaluación y seguimiento de actividades de debate en foro, en la literatura pueden encontrarse diferentes propuestas. En [9] sugieren analizar las interacciones en un foro mediante la construcción de una representación visual a modo de grafo en base a los trabajos de [7,8]. Las métricas propuestas consideran

una serie de indicadores como: los hilos, la posición de cada aporte dentro de un hilo, quién ha sido el autor (alumno o educador), claridad del asunto del aporte, momento en que se publica, lapso de tiempo respecto del último aporte, número de veces que fue leído ese aporte, número de palabras, su propósito principal, la calidad del mensaje, el nivel de complejidad del tema, el nivel de conocimiento, etc. [11], son citados en [6] y afirman que: "...consideran dos tipos de contenidos [que pueden encontrarse en los foros]: los manifiestos y los latentes. Los manifiestos son aquellos que se encuentran en la superficie de la comunicación y es por lo tanto fácilmente observable tales como: el número de veces que un estudiante se dirige a otro por su nombre; a quien se dirigen las intervenciones; la participación; la interacción, el uso de emoticones. El contenido latente está relacionado con aspectos interpretativos e intenta comprender la cognición individual y social".

Otros autores proponen un conjunto de indicadores (algunos coincidentes con los autores ya mencionados) como la participación [5,12], la cantidad y calidad de aportes realizados por los estudiantes [8,13,14], la extensión de los hilos de debate [12], el rol del docente o tutor [12,15], las guías y el *feedback* aportados por éste [5]. Otras, incorporan además una metodología gráfica para la identificación de ciertos patrones de interacción [9,16]. Los autores de [5] definen la participación como el porcentaje de estudiantes inscriptos en el foro que deciden realizar algún aporte en él (participantes activos). La cantidad promedio de aportes por estudiante suele considerarse como el cociente entre el número total de aportes y el número de participantes en el foro [13,14]. En [5] proponen una mejora de este indicador considerando el número de participantes activos. La calidad de estos aportes suele evaluarse mediante rúbricas [17] y análisis de contenidos para establecer patrones de interacción [14,18]. [19] propone una metodología que permite categorizar los aportes en relación a su valoración educativa en base a un conjunto de 12 indicadores: exploratorio, motivacional, argumentativo, crítico, heurístico, reflexivo, interpretativo, afectivo, de opinión, experimental, reproductivo, miscelánea. Los siete primeros caracterizan los aportes valorados educativamente. Los hilos de debate, entendidos como un aporte original que genera cierto diálogo entre los participante a través de alguna respuesta, suelen medirse mediante su longitud media, *i.e.* el cociente entre la cantidad de aportes realizados por los participantes y el número de cadenas de discusión [8,12]. En [5] sugieren categorizar los estilos de debates en tres clases: hablar (aporte original que no recibe respuestas); hablar - replicar (aporte original que recibe alguna respuesta a un primer nivel de profundidad); y discutir (aporte original que recibe múltiples respuestas a un nivel de profundidad mayor al primero). La presencia del docente o tutor puede medirse por la cantidad de sus intervenciones o la naturaleza de éstas [8]. Otros indicadores posibles son la proporción de estas intervenciones en el foro [12]; el cociente entre el número de aportes del docente o tutor y el número de estudiantes inscriptos [13]; o considerar, en este último cociente, el número de participantes activos [5]. En relación a las guías y *feedback* aportados por el docente o tutor, [5] proponen registrar su presencia o ausencia.

Estos trabajos son considerados para el modelo de seguimiento de las interacciones de los foros que aquí se propone, y que se aplican en particular en un contexto educativo de postgrado, para un tipo particular de foro. La necesidad de realizar una propuesta propia se orienta a poder explicar de una manera más acabada las interacciones que se

dan en un tipo de foro particular, que es semiestructurado, organizado en fases y donde los participantes deben simular un rol específico para resolver un caso.

3. Etapas y componentes de una actividad de foro. El caso del Seminario de EaD

En [20] se han presentado las etapas que se consideran propias del diseño y configuración de una actividad didáctica de foro. Aquí solo se mencionan: (a) Determinación de los objetivos didácticos del foro; (b) Armado de grupos (si correspondiera); (c) Diseño y comunicación de la consigna a los alumnos/grupos; (d) Docente a cargo de los foros; (e) Trabajo en los foros propiamente dicho, acorde a las consignas establecidas; (f) Cierre, síntesis o trabajo final de cada foro (si correspondiera acorde al objetivo); (g) Análisis cuantitativo y cualitativo de lo ocurrido a partir de los objetivos de los alumnos y la devolución a los alumnos. Estas etapas han sido consideradas en el caso del foro llevado adelante en el Seminario de Educación a Distancia de la Maestría en Tecnología Informática Aplicada en Educación, que ha sido referida en [20]. Como se presenta en [21] esta actividad resulta de importancia en el desarrollo de aprendizaje de los alumnos del Seminario. El principal objetivo de la actividad es la profundización y puesta en común de los temas de una de las unidades del Seminario.

Los grupos para la actividad se arman considerando las variables: disciplina de origen de los participantes, género, y lugares de procedencia. Estas características se conjugan atendiendo a la grupalidad potencial del conjunto. La grupalidad representa la potencialidad de un conjunto de personas de convertirse en un grupo [22]. En esta condición aparecen competencias individuales y la mirada proyectiva de cómo esas personas funcionarán en un colectivo. Asimismo, se busca heterogeneidad para enriquecer los puntos de vista del debate. Además se asigna un coordinador al grupo atendiendo a patrones de desempeño observados en actividades previas del Seminario.

La consigna consiste en la resolución de un caso, donde los alumnos deben tomar decisiones para llegar a su solución, bajo la consideración de los conceptos abordados en una unidad del Seminario. Los alumnos deberán simular ser expertos en la producción de materiales educativos de educación a distancia para realizar una serie de recomendaciones a quienes los contrataron. El foro se estructura en fases: (a) Fase 1 (presentación más un aporte original combinando investigación bibliográfica y reflexión personal tratando de responder a alguno de los puntos de la temática); (b) Fase 2 (implica al menos una réplica a alguno de sus compañeros para complementar y/o debatir sobre lo escrito); (c) Fase 3 (la conclusión que obtienen del foro una vez revisados los comentarios de sus compañeros).

Se designa un tutor específico que acompaña la tarea de los coordinadores de cada grupo, realiza el seguimiento de las interacciones de cada grupo, registra, observa e interviene, en coordinación con los docentes del Seminario, acorde a situaciones concretas que surgieran. El debate se lleva adelante en el curso en el entorno virtual de enseñanza y aprendizaje utilizado en el marco de la Maestría. La herramienta de foro permite abrir la participación a partir de un momento específico y cerrarlo en el momento acordado con los alumnos en la planificación de la actividad. Finalmente, otro aspecto central en esta actividad es el coordinador de cada grupo de debate en el foro,

que es el encargado de abrir y cerrar el debate, motivar a sus compañeros en la participación, reencauzar el debate si se desvía de su objetivo (aportar ejemplo, dar una cita adecuada para guiar a sus compañeros), y realizar el informe de síntesis de las decisiones tomadas durante el proceso para resolver el caso propuesto en la consigna.

De esta manera, dado el tipo de foro con el que se trabaja se propone un modelo de indicadores de seguimiento que permiten profundizar las propuestas de otros autores al respecto.

4. Propuesta de indicadores de seguimiento de la actividad de foro

Siguiendo los lineamientos de propuestas previas [5,8,20], en este trabajo se construye y aplica un conjunto de indicadores que permiten definir una métrica para evaluar la interacción y participación en una actividad de foro desde una perspectiva multidimensional. El modelo de seguimiento propuesto contempla aspectos como: la participación de los estudiantes; la cantidad de aportes por participante; la calidad de los aportes; la longitud de las cadenas de debate; la presencia del coordinador; y la naturaleza de las intervenciones realizadas por éste. A continuación se detallan estos indicadores:

- La participación de los estudiantes es calculada como la proporción de estudiantes en el foro que participan activamente en el debate.
- La cantidad promedio de aportes por participante es calculada como el cociente entre el número de aportes realizados en el foro y el número de participantes activos.
- La calidad de los aportes es evaluada en función de los objetivos de la actividad. Estos indicadores son propios de los autores de este trabajo y enriquecen los aportes previos encontrados: (a) adecuación de la respuesta a la consigna dada; (b) incorporación de ideas trabajadas en el marco teórico o en los documentos marco del curso; (c) investigación en diversas fuentes de información; (d) actualidad, pertinencia y utilización adecuada de referencias bibliográficas; (e) consideración del aporte de los compañeros; (f) aporte de ejemplos; (g) aplicación a la práctica; (h) creatividad; (i) título del aporte; y (j) grado de síntesis de elaboración de conclusiones.
- Las cadenas de debate se categorizan en tres clases: enunciado (aporte original sin respuestas); réplica (aporte original seguido de respuestas a un primer nivel de profundidad); y debate (aporte original seguido de respuestas a un nivel de profundidad mayor al primero). Además, se considera la longitud media de estas cadenas (cociente entre la cantidad de aportes de los participantes y el número de cadenas).
- La presencia del coordinador del foro es cuantificada mediante el número medio de intervenciones por estudiante activo (cociente entre la cantidad de mensajes enviados por el coordinador y el número de estudiantes que participan).
- Con base en las tareas de coordinación asignadas, las intervenciones del coordinador de cada grupo son analizadas a partir de la presencia o ausencia de los indicadores: (a) incorporación de citas; (b) aporte de ejemplos; (c) apertura o cierre de debate; (d) apertura o cierre de fase; (e) intervención o aclaración; (f) resumen de fase o debate; y (g) motivación o agradecimiento.

Adicionalmente, se aporta una representación gráfica adaptada de [9], que facilita el análisis de las participaciones e intercambios en el foro a través de un grafo no dirigido con las siguientes características: (a) existen tres tipos de nodos: coordinador,

participante, e hilo de debate; (b) todos los nodos se representan con círculos coloreados según el tipo; los nodos correspondientes a los hilos de debate son coloreados según la fase utilizando diferentes tonos del color asignado; (c) los nodos correspondientes a personas son etiquetados con las iniciales del nombre; los nodos correspondientes a hilos de debate son etiquetados con la letra H seguida del orden de iniciación del mismo; (d) a las etiquetas nodales se adiciona, según el tipo de nodo, la cantidad de mensajes enviados por la persona, o el total de mensajes que componen la cadena; (e) las aristas que unen una persona (coordinador o participante) con un hilo de debate indican la participación del primero en el segundo; (f) el tamaño de los nodos aumenta conforme se incrementa la cantidad de aristas que salen o llegan a él, denotando, según el tipo de nodo: una mayor actividad de la persona, o un aporte que potencialmente genera mayor reacción entre los participantes; esto puede observarse en los números que acompañan las etiquetas nodales; (g) dos nodos de personas conectados a través de un nodo que representa un hilo de debate es un indicador de que los usuarios correspondientes están interactuando; (h) el grosor de las aristas aumenta conforme se incrementa la participación de una persona en un hilo de debate particular, denotando un especial interés del estudiante o coordinador en dicho tema; (i) el grafo se organiza en torno al nodo del coordinador con especial atención en la visualización clara de todos los elementos (nodos, aristas, y etiquetas). En la Figura 3 se ejemplifica esta representación.

Este modelo de seguimiento conformado por indicadores se aplica al caso específico del foro explicado en la sección 3. Su aplicación ha permitido derivar en los siguientes resultados preliminares.

5. Resultados preliminares

Aquí se presentan algunos resultados surgidos de la aplicación del modelo de indicadores propuesto. Se registra un total de 34 mensajes en el Grupo 1, de los cuales 15 son enviados por el coordinador; mientras que, de los 25 mensajes del Grupo 2, 9 corresponden a la coordinación. Del análisis de estos intercambios, se deduce que:

- La proporción de miembros del grupo que participan activamente en los debates es del 80 % y 100 % en el Grupo 1 y Grupo 2, respectivamente.
- En el Grupo 1 se registra un promedio de 4.75 aportes por estudiante activo. Esta cantidad desciende a 2.67 en el Grupo 2. No obstante, este promedio es más representativo en el Grupo 2 dado que la actividad de sus participantes ha sido más homogénea (Figura 1(b)), en tanto en el Grupo 1 se observan algunos participantes más activos que otros (Figura 1(a)). La calidad de estos mensajes es analizada en términos de los indicadores mencionados en la sección anterior.
- Los mensajes enviados en cada grupo se distribuyen entre 10 y 20 hilos de debate en los Grupos 1 y 2, respectivamente. Estos hilos tienen características diferentes en sendos grupos: mientras que el 60 % (6) de los hilos generados en el Grupo 1 representan debates genuinos; sólo el 5 % (1) de los emitidos en el Grupo 2 corresponden a esta categoría, siendo la mayoría (16) enunciados sin respuesta alguna. Estas particularidades hacen que los hilos del Grupo 1 tengan una longitud media de 3.4 mensajes, mientras que esta medida desciende a 1.25 en el Grupo 2.

- Los coordinadores muestran una presencialidad cuantitativamente diferente. El coordinador del Grupo 1 muestra una intervención promedio de 3.75 mensajes por participante activo. El coordinador del Grupo 2 publica, en media, 1.5 mensajes por estudiante. Ambos se ocupan de la apertura o cierre de cada fase y el debate en general, con resúmenes parciales; recuerdan la consigna, metodología, y tiempos, o ante retrasos u olvidos de sus compañeros (en forma privada vía la mensajería); e invitan a la participación. La coordinación del Grupo 1 se caracteriza también por formular nuevas preguntas, aportando referencias y ejemplos (Figura 2).

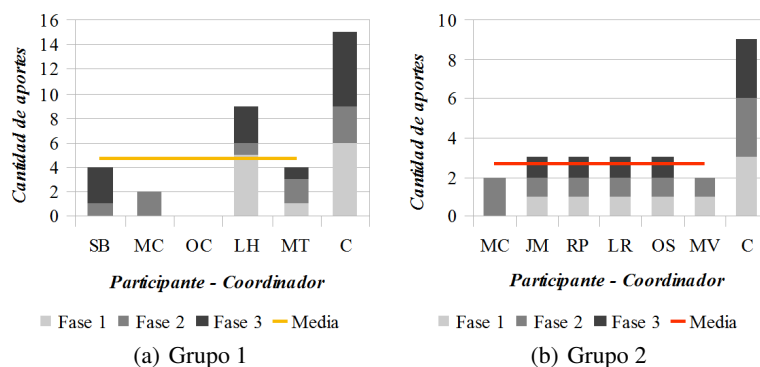


Figura 1. Participación de los integrantes de cada grupo (por fase y en media). Los rótulos de las columnas corresponden a las iniciales de los participantes y el rol de coordinador (C).

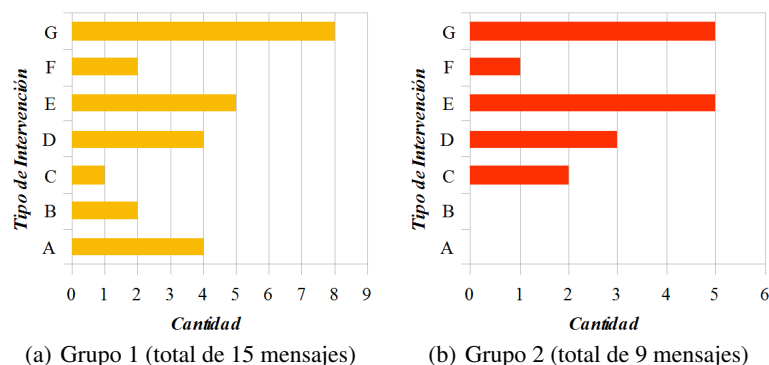


Figura 2. Intervenciones de coordinadores (A: inclusión de citas; B: aporte de ejemplos; C: apertura/cierre de debate; D: apertura/cierre de fase; E: intervención/aclaración; F: resumen de fase/debate; G: motivación/agradecimiento). Algunos mensajes incluyen varios tipos.

Siguiendo los lineamientos presentados en la sección 4, se construyen los grafos que representan los intercambios en los foros (Figura 3). Estas representaciones han sido creadas mediante Gephi³, un *software* de código abierto que permite graficar y analizar redes. El análisis del grafo del Grupo 1 (Figura 3(a)) permite concluir que:

³ <https://gephi.org/>

- La participación es del 80 % (OC no participa). Todos los participantes activos están en contacto a través de algún hilo de debate, enviando entre 2 y 9 mensajes cada uno (un análisis del contenido demuestra que varios mensajes corresponden a informaciones que otorgan valor a la comunicación interna del grupo). El coordinador envía la mayor cantidad de mensajes (15).
- Se generan 10 hilos de debate, distribuidos entre 2 mensajes enviados por el coordinador sin respuestas; 2 mensajes con una única respuesta; y 6 aportes originales con más de una respuesta, generando un debate genuino.
- El debate de mayor longitud (H9) se observa en la Fase 3 y corresponde a la elaboración de conclusiones y el resumen a presentar. Todos los miembros activos participan en este tópico (MC coordina el grupo y participa desde este rol), representando un ejemplo de aprendizaje social.

En el Grupo 2 (Figura 3(b)), en cambio:

- El 100 % de los miembros participa en el foro, enviando entre 2 y 3 mensajes cada uno. El coordinador envía la mayor cantidad de mensajes (9).
- El debate no se evidencia a través de la herramienta (no se usa el “Responder”) sino que se generan nuevos aportes mencionando a los compañeros en su contenido.
- Se generan 20 cadenas de debate distribuidos entre 16 mensajes enviados sin respuesta (al menos no desde la herramienta), de éstos 7 corresponden al coordinador; sólo un debate en tanto se trata de un aporte que recibe más de una réplica en profundidades mayores a uno; los restantes son mensajes con una única respuesta.

A partir de estos análisis se elaboran dos informes, uno escrito y otro audiovisual, para hacer las devoluciones de desempeño y resultados a los estudiantes.

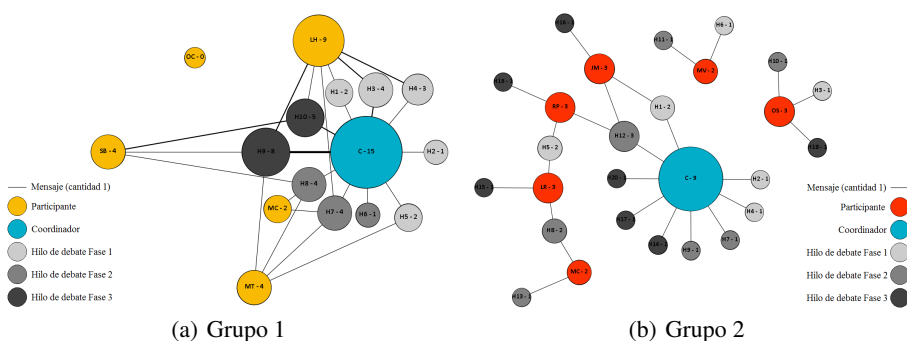


Figura 3. Grafos de intercambios en el foro.

6. Discusión y conclusiones

En la literatura se han encontrado varios trabajos de interés en relación a la propuesta de indicadores para el seguimiento de las interacciones en cursos *online*, y en particular, en foros de debate. Estos trabajos han sido la base para la propuesta que aquí se plantea. En particular, en la propuesta propia presentada se refinan los indicadores que corresponden a la calidad de los aportes, a la naturaleza de los aportes del

coordinador, y a la graficación de las diferentes intervenciones, aspectos que se consideran importantes en el tipo de foro con el que se ha trabajado en el caso de estudio aquí presentado. La aplicación de estos indicadores en el caso del foro del Seminario de Educación a Distancia, ha permitido poner en juego sus posibilidades para el aprovechamiento de los docentes en cuanto a seguimiento y evaluación de lo ocurrido. En relación a esto se observa que gracias a los indicadores referidos a hilos de comunicación se ha podido comparar lo sucedido en los dos grupos de debate analizado, y los docentes pudieron identificar que, mientras en un grupo se dieron debates genuinos, en el otro se abrieron varios hilos pero no se profundizaron.

Los gráficos han sido un elemento fundamental para la visualización de los aspectos más cuantitativos de seguimiento y han facilitado la visualización de las intervenciones del coordinador y del resto de los participantes así como también el seguimiento de los hilos. La visualización representa un tema de enorme importancia en el seguimiento de este tipo de actividades, no sólo para los docentes sino para los propios estudiantes, quienes pueden rápidamente ver de manera modelizada y sintética la forma que ha tomado la construcción, a la luz de los indicadores focalizados, y su propia función dentro de ese entramado. Al mismo tiempo, los indicadores en relación a la calidad de los aportes han facilitado tanto la evaluación de la apropiación de los contenidos por parte de los alumnos en relación a la unidad temática involucrada en el debate, como las estrategias utilizadas como recuperación de citas, ejemplos, o experiencias propias. También han permitido visibilizar las habilidades de los alumnos para construir su discurso a partir del de sus compañeros.

Se considera que esta primera experiencia ha sido exitosa, si bien se trata de un trabajo incipiente en el que aún resta formalizar el análisis de ventajas y desventajas del modelo propuesto a partir de la opinión de los alumnos involucrados, y aplicar estos indicadores a otros casos para extender el análisis. Ha permitido a los docentes (autores del trabajo), profundizar en el seguimiento que años previos se realizaba de esta actividad, y enriquecer el *feedback* tradicionalmente dado a los alumnos en su informe de cierre, para constituirlo en una instancia más de aprendizaje. Como se presentó en el objetivo de este trabajo, el modelo de seguimiento del trabajo del foro, con los indicadores y su visualización, se ampara en la idea de que los estudiantes aprenden no solamente del contenido propiamente dicho de una actividad, sino de la forma que va tomando la construcción de ese contenido. Ese modelo de construcción, que se muestra gráficamente en la sección anterior, se constituye en un nuevo “metacontenido”, que puede consolidarse en metaconocimiento.

Referencias

1. Russo, T., Benson, S.: Learning with Invisible Others: Perceptions of Online Presence and their Relationship to Cognitive and Affective Learning. *Educational Technology & Society* 8(1), 54–62 (2005)
2. Jaggars, S., Xu, D.: How do online course design features influence student performance? *Computers and Education* 95, 270–284 (2016)
3. Bernard, R.M., Abrami, P.C., Borokhovski, E., Wade, C.A., Tamim, R.M., Surkes, M., Bethel, E.C.: A meta-analysis of three types of interaction treatments in distance education. *Review of Educational Research* 79, 1243–1289 (2009)

4. Abel, F., Bittencourt, I., Costa, E., Henze, N., Krause, D., Vassileva, J.: Recommendations in Online Discussion Forums for E-Learning Systems. *IEEE Transactions On Learning Technologies* 3(2) (2010)
5. Bliss, C.A., Lawrence, B.: From posts to patterns: A metric to characterize discussion board activity in online courses. *Journal of Asynchronous Learning Networks* 13(2), 15–32 (2009)
6. Gros Salvat, B., Silva, J.: El problema del análisis de las discusiones asincrónicas en el aprendizaje colaborativo mediado. *RED. Revista de Educación a Distancia* 16 (2006)
7. Kay, R.H.: Developing a metric for evaluating discussion boards. In: Nall, J., Robson, R. (eds.) *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2004*. pp. 1946–1953. Association for the Advancement of Computing in Education (AACE), Washington, DC, USA (2004)
8. Kay, R.H.: Developing a comprehensive metric for assessing discussion board effectiveness. *British Journal of Educational Technology* 37(5), 761–783 (2006)
9. Roig Vila, R., Rosales Statkus, S.E.: Assessing participation in online discussion forums: A proposal for multidimensional analysis. *Píxel-Bit. Revista de Medios y Educación* 40, 137–149 (2012)
10. Dimitrakopoulou, A., Petrou, A., Martinez, A., Marcos, J.A., Kollias, V., Jermann, P., Harrer, A., Dimitriadis, Y., Bollen, L.: State of the art of interaction analysis for Metacognitive Support (2006)
11. Rourke, L., Anderson, T., Garrison, D., Archer, W.: Cuestiones metodológicas relativas al análisis de contenidos de las transcripciones de clases por ordenador. In: Garrison, D.R y Anderson, T. (ed.) *El e-learning en el Siglo XXI: Investigación y Práctica*, pp. 175 – 202 (2005)
12. Mazzolini, M., Maddison, S.: Sage, guide or ghost? The effect of instructor intervention on student participation in online discussion forums. *Computers and Education* 40, 237–253 (2003)
13. Jiang, M., Ting, E.: A study of factors influencing students' perceived learning in a web-based course environment. *International Journal of Educational Telecommunications* 6(4), 317–333 (2000)
14. Nisbet, D.: Measuring the quantity and quality of online discussion group interaction. *Journal of eLiteracy* 1(2), 122–139 (2004)
15. Anderson, T., Rourke, L., Garrison, D.R., Archer, W.: Assessing teaching presence in a computer conferencing context. *Journal of Asynchronous Learning Networks* 5(2), 1–17 (2001)
16. Ralló, R., Gisbert, M.: Análisis de una comunidad on-line a partir de su lista de discusión: El caso de Edutec-L. *EDUTECH. Revista Electrónica de Tecnología Educativa* 25, 1–26 (2008)
17. Bali, M., Ramadan, A.R.: Using Rubrics and Content Analysis for Evaluating Online Discussion : a Case Study From an Environmental Course. *Journal of Asynchronous Learning Networks* 11(4), 19–33 (2007)
18. Meyer, K.A.: Evaluating online discussions: Four different frames of analysis. *Journal of Asynchronous Learning Network* 8(2), 101–114 (2004)
19. Uzuner, S.: Educationally valuable talk: A new concept for determining the quality of online conversations. *MERLOT Journal of Online Learning and Teaching* 3(4), 400–410 (2007)
20. Sanz, C., Zangara, A.: El desarrollo de los foros en el marco de las e-actividades en una propuesta de educación mediada. In: 21. XVII Congreso Argentino de Ciencias de la Computación, CACIC 2011. pp. 538–547. La Plata, Argentina (2011)
21. Sanz, C., Zangara, A.: La formación de docentes en el ámbito de la Educación a Distancia: aspectos epistemológicos y metodológicos. Análisis a partir de un caso. *Signos Universitarios* pp. 487–497 (2014)
22. Schmuck, R.A., Schmuck, P.A.: *Group Processes in the Classroom Second Edition*. WM. C. Brown Company Publishers, Dubuque, Iowa (1975)

4WindGame: Un juego de Armado de Piezas Asistido con una impresora 3D

De Castro, Alejandro ¹, Brussa, Federico ¹,
Sartorio, Alejandro ², Vaquero, Marcelo ², Tedini, Daniel ²

¹ Grupo de desarrolladores e investigadores del proyecto ² Grupo de docentes
Universidad Abierta Interamericana (UAI)
Centro de altos estudios en tecnología informática (CAETI)

Abstract: El presente trabajo detalla los avances del proyecto 4WindGame, un juego lúdico de diseño propio creado mediante el empleo de la tecnología de impresión en 3D y software de diseño libre, con el fin de ser empleado en ambientes académicos. Este permite mediante piezas geométricas aprender conceptos de matemática y geometría. Se presenta un análisis sobre el impacto de los juegos en ambientes académicos primario. Luego se describen aspectos tecnológicos y metodológicos utilizados, con referencias de diseño y recursos de implementación. Por último, se describe una comparativa del juego creado junto con los resultados y las conclusiones finales.

Keywords: impresoras 3d, modelado 3d, educación, juegos, aprendizaje, gamificación, tangram, Scrabbel

1 Introducción

Desde el comienzo en la década del 80, cuando surgen las primeras impresoras 3D, sus aplicaciones fueron destinadas para uso en soluciones industriales, médicas y de ingeniería. Con la evolución de esta tecnología, su alcance fue incrementándose, logrando ser utilizada en producciones personales y hogareñas. Se comenzaron a crear objetos de decoración para casas, autos, juegos de mesas y juguetes para niños. La institución educativa “Cuatro Vientos” [1] planteando la necesidad de atraer a los alumnos hacia el estudio de las matemáticas, propone la utilización de un juego de mesa para lograr tal fin. Con el objetivo de acercar a sus alumnos un juego que facilite la enseñanza e introducción a conceptos sobre geometría, somos invitados a integrar un equipo de trabajo, con el fin de solucionar esta necesidad. El juego creado consiste en utilizar piezas geométricas básicas para formar otras figuras reconocibles, obteniendo un puntaje por la suma de los valores de cada pieza. El equipo de trabajo mediante los requerimientos relevados resuelve, diseñar y crear las piezas mediante la utilización de la tecnología de impresión en 3D. La utilización de esta tecnología permitió gran escalabilidad, ya que las piezas pueden ser modificadas en sus tamaños,

formas, colores y diseño atendiendo solicitudes de docentes y directivos de la institución.

Para lograr el desarrollo se realizó una búsqueda en publicaciones, donde algún juego fuese utilizado como herramienta principal para el estudio de matemáticas y geometría. Encontrando que desde principios del siglo XX se han realizado informes y estudios que respaldan la propuesta de utilizar diferentes tipos de juegos en ámbitos académicos, con el fin de generar en los alumnos un proceso de aprendizaje eficiente mediante la diversión. Entre todos estos informes se encuentra uno que aborda específicamente la temática del aprendizaje a través de recursos lúdicos, llamado “*El Juego Didáctico como estrategia de enseñanza y aprendizaje*” [2]. Este informe fue escrito en el año 2008, identifica cuatro áreas de desarrollo en las que el juego puede tener injerencia. Se mencionan el área físico-biológica, socio-emocional, cognitivo-verbal y la dimensión académica y analiza las características que debe reunir un juego para que sea didáctico y manipular esta clasificación para saber cómo emplearla y cuál sería la adecuada para un determinado grupo de alumnos de rangos de edades específicas.

En el trabajo denominado “*informe Horizon*” [3] se hace referencia a una nueva corriente de aprendizaje basada en juegos, que en los últimos años le otorgó el nombre de gamificación. Según estos informes, esta tendencia educativa potencia la motivación, el esfuerzo, la concentración y demás valores comunes a los juegos para motivar e influenciar a los alumnos, debido a que asimilan el conocimiento de manera simplificada. Este informe también menciona que la tecnología de impresión 3D tendrá impacto en la educación en los próximos años.

2 Requerimientos

Por intermedio de las reuniones mantenidas con el director del centro educativo Cuatro Vientos y de docentes de la asignatura matemática, se optó por la creación de un juego el cual se basa en la formación de figuras geométricas compuestas, partiendo de otras figuras simples, cada figura con un valor numérico. Éste juego además debería tener la suficiente versatilidad en sus niveles de dificultad para atender la variante de edades de alumnos, de entre 8 y 18 años que asisten al centro educativo.

Los requerimientos relevados para la creación del juego fueron los siguientes:

- 1- Presentar diferentes niveles de dificultad para un mismo juego, definiendo las piezas a ser seleccionadas
- 2- Mediante la utilización de distintas piezas geométricas, efectuar la creación de una nueva figura.
- 3- Diseñar las piezas de forma tal que al crear distintas figuras geométricas, se pueda calcular el volumen, área y contorno.

- 4- Presentar un conjunto de problemas a ser resueltos mediante el juego a modo de guía, para que posteriormente los docentes puedan crear o adaptar sus ideas.
- 5- Bajo costo de producción.
- 6- Brindar la posibilidad de ser utilizado en clases de apoyo como en exámenes.
- 7- Escalabilidad en las piezas, teniendo en cuenta la posibilidad de nuevos requerimientos en relación a las formas y tamaños.
- 8- Piezas replicables de fácil reemplazo.
- 9- Cada pieza debe contener un número que otorgue un valor en la formación de la figura final compuesta.

Para diseñar con éxito el juego, además de los requerimientos mencionados, se realizaron búsquedas referentes tanto a la tecnología de impresión 3D como a Juegos en ambientes académicos. Se optó por la creación del juego mediante esta tecnología ya que en esta, en combinación con un software de diseño 3D, otorga la posibilidad de modificación, reconstrucción y reemplazo de cualquiera de las piezas, respondiendo a nuevos requerimientos por parte de docentes.

3 Antecedentes tecnológicos

La tecnología de impresión en 3D se encuentra en desarrollo desde hace varias décadas. Tuvo sus comienzos en los 80, cuando el ingeniero Americano Chuck Hull [4] experimentaba en un laboratorio de la compañía para la cual trabajaba y en donde creaba prototipos de piezas en 3D. Sin embargo, fue en estos últimos años en que esta tecnología ha evolucionado, volviéndose accesible para un segmento no tan específico como lo fue en sus comienzos. Debido a los requerimientos de equipos complejos y a los costos de licencias de softwares de modelado en 3D, estas impresoras eran limitadas solo para ambientes industriales o a avanzados laboratorios de experimentación. El 23 de marzo de 2005 nace el proyecto RepRap [5] el cual cambiaría para siempre el paradigma de la tecnología de impresión en 3 dimensiones. El Doctor Adrián Bowyer [6] profesor de Ingeniería Mecánica de la Universidad de Bath en Reino Unido fue el inventor y fundador de RepRap. Su principal objetivo es la creación de una impresora 3D de bajo costo que tenga la capacidad de ser auto replicable, de tal manera que cualquier persona dispuesta a hacer una inversión mínima pueda hacerse de una. Este paradigma de impresión 3D cambia aún más cuando en el año 2006 la empresa Google lanza al mercado una distribución gratuita de “Sketchup” [7], un sencillo pero potente software multiplataforma que permite con escasos conocimientos modelar piezas en 3 dimensiones, que luego pueden ser traídas a la forma física con una impresora 3D. Pronto la tecnología comenzó a emplearse en

bibliotecas y universidades de todo el mundo, en donde se utiliza como complemento en ciertas materias. Según un informe publicado en 2015 por Maite Comalat [8] de la universidad de Barcelona. En los Estados Unidos existen más de 50 bibliotecas tanto públicas como de universidades privadas que en los últimos años incorporaron al menos una impresora 3D.

3.1 Juegos en la educación

El empleo de juegos para fines educativos es una temática que ha sido extensamente analizada. En 1982 el conocido Informe Cockroft [9] señala en uno de sus párrafos que sin importar el nivel de conocimiento que tengan los alumnos, el uso planificado y estructurado de juegos matemáticos contribuye a desarrollar el pensamiento lógico. En 1989 fue publicado el documento *“El juego en la educación infantil y primaria”* [10] en donde su autora menciona que pensadores clásicos como Platón y Aristóteles otorgaban gran relevancia al aprendizaje mediante juegos, animando a padres a emplear con sus hijos juguetes que contribuyan a formar sus mentes para futuras actividades como personas adultas. A lo largo de la historia muchos pensadores y autores de renombre hacen referencia a la importancia y beneficios de utilizar juegos como factor potenciador para el desarrollo psíquico y físico de los niños.

Como se menciona en el documento; *“El tangram: un recurso educativo para trabajar la geometría en la educación primaria”* [11] este juego sirve como estímulo para fomentar tanto la motivación como la creatividad del alumnado, pudiendo ser empleado para el conocimiento y práctica de las matemáticas, como así también introducir conceptos básicos de geometría. Este juego está compuesto por siete figuras geométricas, esta combinación de piezas permite la creación de múltiples formas, letras y números. Mediante el empleo de este juego, los docentes pretenden que los alumnos puedan asumir y practicar actitudes tales como la colaboración, el trabajo en equipo, desarrollar la creatividad, perseverancia y participación entre los niños. Existe otro documento llamado *“El juego Scrabble como herramienta de Lenguaje Integral”* [12], este afirma que es posible mejorar y perfeccionar el empleo de la escritura y el lenguaje en un ambiente lúdico mediante el empleo del mismo. La esencia de este juego es la formación de palabras, empleando piezas que contienen letras con un valor numérico. Existe otro proyecto llamado *“Creasix”* [13], que haciendo uso de las T.U.I. (Tangible User Interface) propone mediante la combinación de diferentes figuras en una interfaz web y en conjunto con una impresora 3D, el diseño de juguetes por parte del público infantil para luego ser impresos y entregados a sus diseñadores. Es en este contexto, que partiendo del análisis de los documentos y trabajos citados, se diseña un juego que adopta características de cada uno de ellos, logrando la creación de 4WindGame. Un Juego, con la capacidad de cubrir distintos niveles de dificultad a través de la creación de diversos problemas, abarcando tanto a alumnos de ciclo inicial como de años superiores. Este juego pretende acercar conocimiento de geometría por intermedio de la actividad lúdica en el aula, donde los participantes y el docente desarrollen la imaginación, creatividad, colaboración, trabajo en equipo y pensamiento lógico combinando figuras geométricas. Asimismo posibilitando a través del juego el cálculo

de áreas, perímetros y clasificación de ángulos de las distintas piezas. Como otra de sus características importantes, es que el juego cuenta con la posibilidad que los educadores soliciten modificaciones en las piezas estándares o creaciones de nuevas piezas modificando las existentes.

4 Proceso de producción

Una vez relevados los estudios sobre juegos en la educación y los requerimientos establecidos por los docentes se adoptó un esquema de figuras similar a las piezas del juego Tangram. Con la diferencia que el número de piezas es superior a las 7 de éste, pudiendo emplearse la cantidad deseada. Este hecho proporciona mayor versatilidad en la formación de figuras, permitiendo la producción de un mayor número de estas. Como en el juego “Scrabble”, cada pieza de 4WindGame contiene un valor numérico. Es a través de este número que se podrán sumar las piezas, contabilizando qué jugador tiene mayor puntaje, definiendo de este modo un ganador.

Nuestro trabajo partió con el diseño de las piezas que el juego requería, moldeando las figuras en papel para entender la forma en que éstas debían encastrar. Luego una vez obtenido el resultado deseado se continuó con la etapa de diseño en 3D, empleando el software “Sketchup”.

Todos los diseños de las piezas de juegos creados mediante Sketchup fueron exportadas al formato de archivo “STL” [14]. Este tipo de extensión es interpretado por otro software llamado “ReplicatorG” [15], el cual define valores referentes a espesor de las capas de plástico depositadas, temperatura de la salida del material, velocidad de impresión y parámetros que determinan la calidad de las piezas impresas. Inspirados en el proyecto Creasix y pensando que el juego fue creado para ser utilizado en ambientes académicos, junto con cada ejemplar se entrega un conjunto de ejercicios guía.

Para hacer un correcto uso de las posibilidades de reemplazo y modificación de las piezas que componen el juego, fue creada una plataforma de gestión de solicitudes de pedidos. Esta se basa en los formularios de “Google Forms” [16]. Se diseñaron dos formularios, uno para solicitud de reemplazo de piezas (Fig. 1 y Fig. 2) y otro para solicitud de modificación (Fig. 3 y Fig. 4). Al completar cada formulario, el usuario especifica de manera detallada la pieza a ser reemplazada o bien cuáles son las modificaciones que desea realizar sobre alguna de estas. También se encuentra disponible la opción para la creación de una pieza nueva. La información obtenida de estos formularios es persistente en una planilla en donde posteriormente son gestionados los pedidos a ser realizados.

Fig. 1 Formulario de solicitud de piezas

	A	B	C	D	E
1	Marca temporal	Pieza solicitada para reposición?	color de la pieza?	Datos Personales	Fecha de Retiro
2	30/7/2016 19:58:56	Triángulo Rectángulo número 3	Verde	Puebla Alejandro, Ovidio Lagos 944, 156 550160	10/8/2016
3	30/7/2016 20:16:21	Triángulo Rectángulo número 4	Verde	Ezequiel Sendeña, San Luis 2345, 156443256	17/8/2016
4	30/7/2016 20:42:19	Triángulo Rectángulo número 2	Purpura	Alejandro De Castro, San Salvador 314 156 550160	17/8/2016
5	30/7/2016 20:48:59	Triángulo Rectángulo número 2	Purpura	Fernando Amas, Roca 991, 3415 666 443	17/8/2016
6	30/7/2016 21:14:08	Triángulo Rectángulo número 4	Rojo	Pablo Contes, Rioja 2345, 3416 5543332	17/8/2016
7	30/7/2016 21:51:55	Triángulo Rectángulo número 3	Verde	Rocio penciarolo, San Lorenzo 2344, 156 497854	17/8/2016
8	30/7/2016 21:56:11	Triángulo Rectángulo número 3	Verde	Alejandro De castro, Rioja 2776, 156554356	17/8/2016
9	30/7/2016 22:00:13	Paralelogramo número 5	Rosa	Rocio penciarolo, San Lorenzo 2344, 156 497854	17/8/2016
10	30/7/2016 22:57:17	Triángulo Rectángulo número 4	Purpura	Alejandro Sartorio, Corrientes 1245, 154, 665435	17/8/2016
11	31/7/2016 10:41:47	Cubo número 1	Azul	Alvaro pantanetti, 9 de Julio 764, 155 496005	24/8/2016
12	31/7/2016 10:43:07	Triángulo Rectángulo número 2	Rojo	Cristian Cencotto, Tucuman 2645, 155 112321	30/7/2016
13	31/7/2016 10:44:28	Triángulo Rectángulo número 3	Naranja	Brais Monnier, Balcarce 633, 4264433	21/9/2016
14	31/7/2016 10:45:19	Triángulo Rectángulo número 4	Bianco	Damian Pomar, San Juan 4003, 15 568882	26/7/2016
15					

Fig. 2 Planilla de solicitud de Piezas

Fig. 3 Formulario de modificación de piezas

	A	B	C	D	E	F
1	Marca temporal	Que pieza desea Modificar?	Detalle Modificación	Datos personales		Fecha de Hecho
2	2/17/2016 12:45:44	Triángulo 4 (70 milímetros de base)	Por favor recorte esta pieza en 3 colores distintos	Daniel Alaraz, Ulre 3455, 155124695		1/10/2016
3	2/17/2016 11:46:05	Paralelogramo 5 (50 milímetros de base)	cambiar el valor numerico, de 5 a 8	Mario Mico, Formosa 2254, 154330379		20/8/2016
4	2/17/2016 15:00:00	Paralelogramo 5 (50 milímetros de base)	cambiar el valor numerico, de 5 a 8	Mario Mico, Formosa 2254, 154330379		20/8/2016
5	3/17/2016 22:20:51	Triángulo 3 (85 milímetros de Base)	Cambiar el ángulo a 120 grados	Eros marnell, Calbo Jiboa, 166443330		6/7/2016
6	3/17/2016 22:52:32	Cubo 1 (35 milímetros de Base)	hacerlo de 20mm por 50mm	Bill Gates, Madison Avenue 2034, USA		13/8/2016
7	3/17/2016 22:58:28	Cubo 1 (35 milímetros de Base)	Cambio en los milímetros a 60mm por 60 mm	Cesar Gonzalez, San Salvador 314, 1561		9/9/2016
8	3/17/2016 15:54:18	Triángulo 2 (70 milímetros de Base)	hacer la base a 150 milímetros, cambiar el ángulo a 110 grados	Carlos Blanco, Rivioli 1564, 1554348524		21/7/2016
9						

Fig. 4 Planilla de modificación de piezas

Otra característica del juego, es que presenta dos niveles de dificultad, el nivel 1 diseñado para alumnos de entre 8 y 12 años y el nivel 2 para alumnos de 13 a 18 años, teniendo este último un mayor grado de dificultad. A modo de ejemplo se colocan algunos de los problemas planteados para cada nivel, debido a que el grado de conocimientos de geometría en los grados inferiores es menor que en los grados superiores.

Problema para alumnos nivel 1

Esta categoría corresponde a nivel inicial, donde los alumnos se encuentran en el rango de edad entre 8 y 12 años. Se intenta lograr, que mediante la unión de dos o más formas reconocibles puedan crear una tercera, logrando sumar el mayor puntaje posible, tal como se muestra en las figuras Fig.5 y Fig.6.



Fig. 5 Figura de casa compuesta



Fig. 6 Figura de casa simple

Problema para niños nivel 2

La categoría denominada nivel 2, está orientada a alumnos de ciclos superiores, estos alumnos poseen conocimientos de geometría y lógica, por lo cual el grado de dificultad de los problemas a resolver es mayor. Se cita a modo de ejemplo los ejercicios a, b, c y d.

- a - Reúna las piezas necesarias para formar un pentágono
- b - Reúna las piezas necesarias para formar un hexágono
- c - Reúna las piezas necesarias para formar un octágono
- d - Forme un hexágono con 6 triángulos

4.1 Proceso de agregado de piezas.

El agregado de piezas surge de la necesidad de docentes que utilizando el juego encontraron posible crear nuevos problemas a resolver, adicionando piezas. Como ejemplo se cita el caso de un docente que a través del formulario requiere las piezas necesarias para formar un rectángulo de 100 x 50 milímetros, empleando tres triángulos. Con las piezas entregadas en el juego esto no era posible, por lo que se adicionó un tercer triángulo recto obteniendo el resultado solicitado como se observa en la figura Fig. 3.

4.2 Proceso de rediseño de piezas.

Para aumentar la complejidad del juego y extenderlo a niveles de mayor capacidad de resolución, se modifican las figuras básicas de las piezas. Ejercicios más complejos no pueden realizarse a partir de cuadrados sino de figuras hexagonales.

Se cita el ejemplo de un docente que solicitó rediseñar la pieza de cubo para convertirla en hexágono, de forma de poder resolver el siguiente problema: Suponga que H_1 y H_2 son hexágonos regulares. La longitud de la apotema de H_1 es 30 milímetros y la de H_2 es 50 milímetros. Si A_1 es la medida del área de H_1 y A_2 la de H_2 , entonces cuál es la razón A_1/A_2 ? Se observa el resultado en la figura Fig.7 y Fig.8

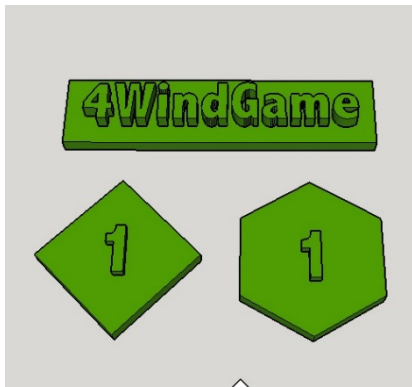


Fig. 7 Modificación de pieza



Fig. 8 Adhesión de pieza

5 Resultados y Descripción de la propuesta educativa

La actividad se llevó a cabo en la institución Cuatro Vientos en un evento que los directores y miembros del establecimiento nombraron como olimpiadas matemáticas. Durante el desarrollo de la actividad se presentaron 180 alumnos divididos en dos grupos, los alumnos de nivel primario formaban un grupo y los de nivel secundario otro.

En la tabla 1 se muestra el resultado de una breve encuesta realizada a alumnos. Para determinar si el aprendizaje, mediante el juego de geometría en el que fueron partícipes durante el periodo de preparación, les resultó más eficiente que la manera tradicional de enseñanza.

Tabla 1 Resultados de encuesta realizada

Nivel	Primario	Secundario
<i>Cantidad de alumnos por nivel</i>	133	56
<i>Porcentaje al que le resultó bueno el aprendizaje mediante piezas 3d</i>	95%	99%
<i>Porcentaje de alumnos que piensa que es igual jugar a aprender que aprender de la manera tradicional</i>	1%	0,40%
<i>Porcentaje de encuestas no contestadas</i>	4%	0,60%

La actividad fue realizada durante 4 sesiones de 50 minutos, 2 sesiones para el nivel primario y 2 para el nivel secundario. Los exámenes fueron realizados en grupo de 5 participantes como máximo y 3 como mínimo en ambos niveles. A cada grupo les fue entregada una hoja con diferentes tipos de ejercicios. Para resolver los problemas de geometría fue utilizado un kit de 4WindGame. En la hoja de examen se debía colocar la hora de inicio y fin, ya que en caso de empate se definiría por tiempo. Cuando los alumnos llegaban a los problemas geométricos, debían utilizar las piezas y eran supervisados por docentes, quien colocaba el resultado en la hoja de examen.

Comparativa entre 4WindGame y otros juegos

La tabla 2 resalta las diferencias entre la enseñanza mediante juegos como Tangram, Scrabble y 4WindGame.

Tabla 2 Comparativa entre Juegos

	Comparación de juegos		
	Tangram	Scrabble	4WindGame
Aprendizaje de geometría y matemática	X		X
Utilización de planos	X		
escalabilidad			X
Niveles de dificultad variables		X	X
Invencción de juego	X		X
Utilización de Tecnología libre de licencias			X
Motricidad Fina			X
Bajo costo de producción	X		X
Fomentación de trabajo en grupo		X	X

Facilidad en el diseño de piezas			X
Posibilidad de reposición de piezas			X

Los atributos de la primera columna se relacionan con las características de cada uno de estos juegos, de esta forma queda demostrado que 4WindGame jugando para aprender abarca la mayoría de los atributos, determinando otro de los principales aportes de este trabajo. El atributo motricidad fina hace referencia a que la formación de figuras mediante las piezas del juego ayudan a los más pequeños a poder manipular las piezas y así crear diferentes formas con estas.

Bajo costo de producción, debido a la utilización de la impresión en 3d y al bajo valor económico del material que estas utilizan, se pudo crear el juego con un precio de producción menor al que costaría cualquier juego de similares características en el mercado.

Conclusión

En las pruebas de campos realizadas se reflejó que la propuesta de utilizar un juego como medio de enseñanza asistido por una impresora 3D, facilita el aprendizaje de matemáticas y geometrías. La utilización de figuras geométricas en 3 dimensiones para resolver problemas, es más intuitiva que la metodología tradicional para la resolución de las operatorias.

Tecnológicamente, se aporta un diseño, implementación e información necesaria para instrumentar soluciones a docentes y alumnos a través de un juego que brinda mejoras en el proceso de aprendizaje. Se demuestra que gracias a la tecnología de impresión 3D fue posible crear y modificar nuevas piezas del juego como se hace mención en los puntos 4.1 y 4.2

Referencias

- 1 A. R. d. A. Solidaria, «ARAS,» <http://www.aras.org.ar/> (2016).
- 2 Paula Chacón, «El Juego Didáctico como estrategia de enseñanza y aprendizaje,» (2008).
- 3 T. N. M. Consortium, «Informe Horizon,» Peggy Snyder Executive, (2016).
- 4 C. H. Biography, «Decolage Contenidos Digitales,» <http://www.decollagedesign.com/chuck-hull/>, (2016).
- 5 RepRap, «RepRap/es,»: <http://reprap.org/wiki/RepRap/es>, (2016).
- 6 Adrian. Bowyer.

- <https://scholar.google.com.ar/citations?user=M4xonK4AAAAJ&hl=es>.
- 7 Sketchup. [En línea]. Available: <https://es.wikipedia.org/wiki/SketchUp>, (2016).
 - 8 Maite Comalat Navarra, «Bibliotecas públicas e impresoras 3D: el debate está servido.,» *El profesional de la información*, (2015).
 - 9 Joaquín Perez Navarro, *Las matemáticas, sí cuentan*, vol. Vol 20, Madrid: Ministerio de Educación., (1985).
 - 10 Janet R. Moyles, *El juego en la educación infantil y primaria*, vol. Vol. 16, Madrid: Ediciones Morata, (1990).
 - 11 Jose Felix Cuadrado Morales, «EL TANGRAM: UN RECURSO EDUCATIVO PARA TRABAJAR LA GEOMETRÍA EN LA EDUCACIÓN PRIMARIA.,» Madrid, 2010.
 - 12 Christian Arguedas Vargas, «EL JUEGO SCRABBLE EN LA BIBLIOTECA COMO HERRAMIENTA DEL LENGUAJE INTEGRAL.,» (2015).
 - 13 Alex Pemán García, «Creasix: app de diseño y producción de juguetes mediante TUI e impresión 3D.,» (2015).
 - 14 todo3d.cl, «Que es un archivo STL.,» (2015).
 - 15 ReplicatorG, «open source 3D printing program.,» <http://replicat.org/start>, (2016).
 - 16 Google. Inc., «Google Forms.,» <https://www.google.com/intl/es-419/forms/about/>, (2016).
 - 17 Maurice S. Kanbar, «Tangram game assembly.,» Patent and Trademark Office., (1981).
 - 18 Mequé Edo i Basté, «Juegos y matemáticas. Una experiencia en el ciclo inicial de primaria.,» *Uno: Revista de didáctica de las matemáticas*, (1998).
 - 19 Jose María Gairín Sallán, «Efectos de la utilización de juegos educativos en la enseñanza de las matemáticas.,», (1990).

Sistema para el Control, Administración y Programación de Robots Educativos

Germán Osella Massa¹, Cecilia De Vito¹, David Fernandez¹,
Mónica Sarobe¹, Claudia Russo¹,

¹ Instituto de Investigación y Transferencia en Tecnología (ITT), Escuela de Tecnología, Universidad Nacional del Noroeste de la Provincia de Buenos Aires (UNNOBA), Sarmiento y Newbery (CP 6000), Junín, Buenos Aires, Argentina. Tel: (0236) 4636945/44 {german.osella, cecilia.devito, david.fernandez, monica.sarobe, claudia.russo}@itt.unnoba.edu.ar

Resumen: El presente trabajo trata sobre el sistema creado para el control y la programación del Robot Educativo Programable desarrollado en la UNNOBA, detallando diferentes casos de uso que permite la arquitectura de software propuesta y describiendo la biblioteca de funciones diseñadas para comandar al mencionado robot. Este sistema permite expandirse tanto para controlar nuevos robots así como para emular otras bibliotecas incluidas con otro tipo de robots.

Palabras clave: Robot, Educación, Programación, Administración.

1 Introducción

En la UNNOBA se está trabajando en el Robot Educativo Programable (REP), que combina la máxima integración de partes nacionales con la posibilidad concreta de ser armado y puesto en marcha por alumnos avanzados de escuelas con carreras técnicas, buscando brindar interfaces de programación acorde a las habilidades que se quieran desarrollar en los alumnos de los diversos niveles educativos.

El prototipo actual, una evolución del descrito en [1], posee un chasis de policarbonato en el que se montan dos motores de corriente continua con rodamientos de espuma de goma junto con una tercera rueda que actúa de apoyo. Cuenta con un procesador Arduino Nano, basado en un ATmega328 de 8 bits funcionando a 16 MHz, con 14 pines de E/S digital, 8 de E/S analógica, 30 KB de memoria flash, 2 KB de memoria SRAM y 1 KB de memoria EEPROM. Se agregan dos sensores ultrasónicos al frente del robot y dos sensores IR orientados hacia abajo para funcionar como detector de líneas. Cuenta con un módulo bluetooth de comunicación inalámbrica. La tecnología electrónica, el diseño industrial y la selección de materiales fue elegida por adaptarse a las posibilidades actuales de nuestro país.

El proyecto es un desarrollo libre y abierto. Las especificaciones técnicas, los esquemáticos de los circuitos y los diseños de varias piezas a producir mediante impresión 3D están disponibles bajo una licencia permisiva. El código fuente del software asociado se encuentra publicado en GitHub [2] bajo licencia GPL versión 3.

Este trabajo describe en detalle el sistema de software diseñado para la utilización del robot dentro del aula. En la sección 2 se presenta la arquitectura del sistema implementado, explorando distintas formas de emplearlo. En la sección 3 se detalla la interfaz de programación provista por la biblioteca usado por el usuario para controlar a los robots. En la sección 4 se ahonda en cuestiones técnicas relacionadas con el diseño interno del sistema. Finalmente, en la sección 5 se expondrán algunas conclusiones y futuros trabajos.

2 Arquitectura del sistema de control

Uno de los objetivos para el robot es que sea apto para la enseñanza, por lo que se consideró importante simplificar las tareas asociadas con su uso, particularmente cuando debe ser programado por usuarios inexpertos. El robot no es autónomo sino que se lo comanda desde una computadora en la que se realizan todas las tareas asociadas con la programación, utilizando órdenes simples dadas en un lenguaje de muy alto nivel. Esto evita que el usuario requiera conocimientos específicos de la arquitectura del hardware para poder cargar un programa en la memoria del microcontrolador que gobierna al robot. No obstante, esto demanda de algún tipo de comunicación inalámbrica entre el robot y la máquina que lo comanda. Por simplicidad operativa, se utiliza el estándar bluetooth, buscando el balance entre las variables costos/disponibilidad/confiabilidad. Se descartó el uso de transmisores RF y XBee por necesitar de hardware adicional específico y por los costos involucrados.

La comunicación serie provista por bluetooth es punto a punto. Si un programa establece una conexión con el robot, el enlace se produce de forma exclusiva, impidiendo que otros programas tengan acceso simultáneo y prohibiendo la creación de monitores del estado interno del robot o de mecanismos de parada de emergencia, fundamentales cuando se observa un comportamiento erróneo o peligroso.

Para sortear esta limitación, el sistema de control se vale de una arquitectura cliente/servidor en donde el servidor, denominado “Administrador”, actúa de nexo entre los robots y los programas escritos por los usuarios. Un administrador adquiere el control exclusivo de uno o más robots a través de sus respectivos enlaces bluetooth, brindando acceso a los mismos a todo cliente que se conecte con él a través de una red TCP/IP, creando así un nivel de indirección adicional entre el usuario y el robot, eludiendo de esta forma la limitación expuesta previamente.

Un programa del usuario se transforma entonces en un cliente de un administrador tras incorporar la biblioteca del sistema de control desarrollada para tal fin. Esta biblioteca contiene la lógica de comunicación con los administradores, al mismo tiempo que provee todos los comandos necesarios para controlar a los robots.

Cuando se está aprendiendo a programar es importante poder inspeccionar que está sucediendo mientras el programa recién escrito se está ejecutando, permitiendo así el desarrollo de modelos de pensamiento que expliquen los resultados observados. Un administrador incorpora la facilidad de generar notificaciones por cada comando enviado a un robot, junto con el resultado de ejecutarlo, habilitando así la creación de monitores que permitan explicar el comportamiento observado.

Para no limitar el uso del robot a un sistema operativo, tanto el servidor de control como la biblioteca para el usuario son multiplataforma, ejecutándose tanto sobre Windows como sobre sistemas basados en GNU/Linux o Mac OS X. La biblioteca también funciona sobre Android, permitiendo controlar al robot desde una tablet o teléfono inteligente. El código del sistema está escrito en Python 3 [3], siendo factible crear nuevas bibliotecas en otros lenguajes de programación.

El sistema es adaptable, pudiendo aplicarse tanto para controlar a un único robot comandado desde una sola computadora hasta facilitar la interacción simultánea con muchos robots controlados desde varias máquinas, en donde la cantidad de robots no es necesariamente la misma que la de computadoras controlándolos.

La progresión de casos de uso planteada para este sistema comienza con una única computadora sobre la cual se realiza toda la programación, controlando a uno o más robots mediante el uso de un adaptador bluetooth (Figura 1). Cada robot debe ser asociado una única vez con la computadora en cuestión, como cualquier otro dispositivo bluetooth, asignándole un nuevo puerto serie propio. Una vez hecha esta asociación, todo programa escrito para controlar a los robots deberá iniciarse indicando que se desea trabajar con un robot en particular, dándole un nombre que lo identifique junto con el puerto serie correspondiente. Esto debe repetirse para cada robot que se desea controlar. Tras registrar el primer robot, automáticamente se crea un administrador local oculto para interactuar entre el programa y los robots, encargándose de la comunicación bluetooth y del diálogo con los robots. Dicho administrador se detendrá automáticamente tras finalizar la ejecución del programa que lo inició. Este esquema es útil para realizar pruebas rápidas o cuando se cuenta con un robot por computadora, no siendo adecuado para compartir el uso de los robots entre varias máquinas.

Para esta segunda situación se requiere de un administrador explícitamente iniciado, actuando de nexo entre los robots y las demás computadoras o dispositivos desde donde controlarlos (Figura 2). Dicho administrador debe configurarse con los nombres de todos los robots a los que proveerá acceso, junto con los puertos de comunicación asociados a los enlaces por bluetooth. El administrador recibirá las órdenes a enviar a los robots, respondiendo a cada una de ellas con el resultado de su ejecución. También comunicará a todo cliente interesado información describiendo las órdenes ejecutadas junto con otros tipos de notificaciones relevantes. Este esquema resulta apropiado cuando se cuenta con varios robots y se los quiere compartir por turnos para controlarlos desde otras máquinas conectadas dentro de la misma red. Se requiere de una computadora donde ejecutar el administrador, la que igualmente puede usarse como puesto de trabajo adicional. Es obligatorio para este esquema disponer de una red de datos que comunique a todas las máquinas o dispositivos con aquella sobre la cual funcione el administrador.

Finalmente, es posible segregar a los robots en grupos controlados por diferentes administradores ejecutándose sobre máquinas distintas (Figura 3). Con este esquema se consigue separar físicamente a los robots, alejando a los administradores entre sí mucho más allá del radio de alcance típico del estándar bluetooth. También permite agrupar a ciertos robots por área temática, equipos o bandos. Un dispositivo podrá tener acceso a todos los robots disponibles en un administrador y también podrá comunicarse con varios administradores a la vez, teniendo así el potencial control de todos los robots que haya. Este esquema es el más complejo que se ha contemplado.

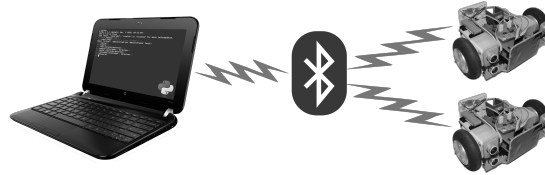


Figura 1. Uno o más robots controlados desde una única computadora vía bluetooth.



Figura 2. Muchos dispositivos controlando a varios robots a través de un Administrador.

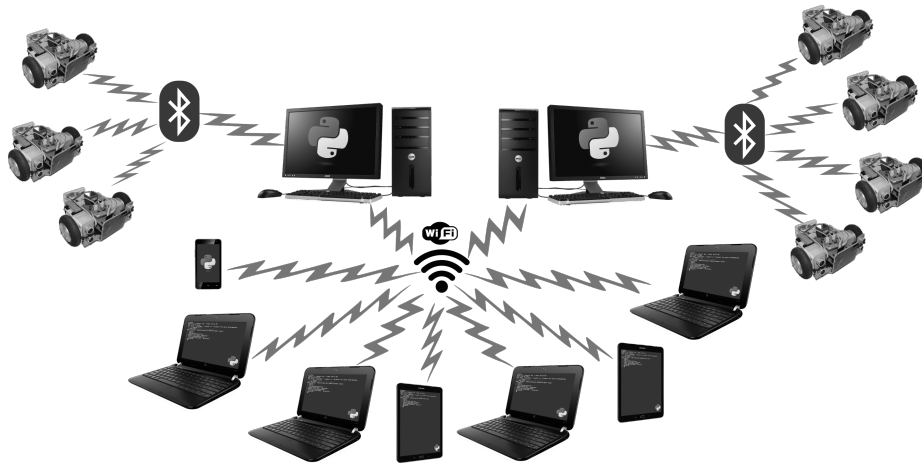


Figura 3. Administradores proveyendo a todos los dispositivos en red el acceso a los robots.

La arquitectura del sistema también es extensible, permitiendo desarrollar nuevos clientes que provean otras interfaces de programación para los robots y previendo la creación de nuevos administradores que comanden a otros tipos de robots. Como prueba de concepto, se desarrolló un módulo que emula la interfaz del DuinoBot de los N6 [4] utilizados en las experiencias llevadas a cabo por docentes investigadores del LINTI [5], traduciendo sus comandos en las correspondientes acciones sobre los robots de la UNNOBA. De esta forma, es posible reutilizar con mínimas modificaciones todos los programas ya escritos para los primeros.

3 Biblioteca para el control del robot

La biblioteca creada para controlar a los robots se diseñó para ser simple pero completa, potente y homogénea, buscando ser empleada con fines didácticos tanto en el aprendizaje de la programación imperativa así como en otras áreas de la currícula. Se eligió a Python 3 como lenguaje de programación utilizado para comandar al robot, por ser claro, predecible y aplicado con éxito en el ámbito académico [6]. Tanto la biblioteca de alto nivel como el administrador están escritos en este lenguaje.

La biblioteca del robot expone toda la funcionalidad necesaria para su uso, la que puede catalogarse en agregar y quitar robots locales y administradores remotos, solicitar a un robot que realice varios tipos de movimientos y encuestarlo por el estado de sus sensores. También incluye funciones auxiliares convenientes para escribir programas que controlen robots, incorporando manejo de tiempos, esperas y toma de decisiones al azar.

Los nombres de las funciones expuestas por la biblioteca están en español, escritos usando acentos y ñes donde corresponda, aprovechando el soporte nativo de Unicode [7] que Python 3 posee para los identificadores de un programa.

La biblioteca provee dos formas de ser utilizada, según el paradigma de programación que se desee emplear. Es posible usar una interfaz netamente procedural, la cual resulta sencilla cuando se trabaja con un único robot. También se provee una interfaz basada en objetos, que simplifica notablemente el control de varios robots simultáneamente. Una vez instalado el intérprete de Python 3 junto con el software para el control del robot, se sugiere utilizarla tanto en un programa como en el shell interactivo importándola directamente con `from edubots import *`

Este encantamiento mágico importa el módulo principal de la biblioteca del robot y trae al ámbito global del programa todas las funciones provistas por la misma. Si bien esta no es una práctica recomendable cuando se escriben programas grandes compuestos por varios módulos desarrollados en forma independiente por distintos programadores, resulta extremadamente conveniente cuando se intenta dar los primeros pasos introductorios, en donde los programas son relativamente pequeños y contenidos en un único módulo.

Los comandos disponibles tras importar la biblioteca pueden agruparse según la tarea asociada a los mismos. A continuación se describen los comandos asociados netamente con cuestiones administrativas, que permiten registrar robots para usarse directamente desde la máquina en la que se lo programa o para informar al sistema de la existencia de administradores que provean acceso a los robots:

`robot_local(nombre, dispositivo)`: Registra la existencia de un robot que se identificará con el nombre dado, comunicándose a través del dispositivo indicado. Esta función retorna un objeto que representa al robot recién registrado.

`olvidar_robot(nombre)`: Solicita al sistema la desactivación y olvido del robot identificado con el nombre dado. Tras olvidar exitosamente al robot, éste no podrá seguir usándose hasta tanto no se lo vuelva a registrar.

`administrador_remoto(dirección)`: Registra un administrador ubicado en la dirección IP dada como parámetro. Una vez agregado, todos los robots que ese administrador controle estarán disponibles para ser usados desde el programa. Retorna un objeto que representa al administrador registrado.

`olvidar_administrador(dirección)`: Quita un administrador previamente registrado con esa dirección. Los robots que ese administrador poseía desaparecerán.

`administradores()`: Devuelve una lista con los administrados registrados.

`robots()`: Devuelve una lista con los nombres de todos los robots conocidos.

`robot([nombre])`: Establece al robot identificado con el nombre dado para que sea el que ejecutará todos los sucesivos comandos. En caso que no se dé un nombre, se elige automáticamente al primer robot disponible. Retorna el robot elegido.

`nombre()`: Devuelve el nombre que identifica al robot usado por defecto.

Los siguientes comandos permiten realizar una interacción básica con un robot:

`suspender()`: Causa que el robot por defecto se detenga e ignore los sucesivos comandos recibidos, hasta tanto no se vuelva a reanudar su funcionamiento.

`reanudar()`: Reactiva al robot por defecto para que vuelva a aceptar comandos.

`detenerse()`: Para los motores del robot por defecto, deteniéndolo.

También se dispone de comandos asociados con el movimiento del robot, los que son simples, homogéneos y prevén un amplio abanico de movimientos pre-programados. No se describirán por separado a cada una de las funciones disponibles puesto que se espera su significado sea obvio. Las funciones con las que se cuenta son:

```
avanzar([velocidad], [duración])
retroceder([velocidad], [duración])
doblar_izquierda([velocidad], [duración])
doblar_derecha([velocidad], [duración])
doblar_izquierda_avanzando([velocidad], [duración])
doblar_derecha_avanzando([velocidad], [duración])
doblar_izquierda_retrocediendo([velocidad], [duración])
doblar_derecha_retrocediendo([velocidad], [duración])
rotar_izquierda([velocidad], [duración])
rotar_derecha([velocidad], [duración])
```

Todas las funciones efectúan la acción que su nombre refleja y, opcionalmente, la realizan a la velocidad dada y durante la duración indicada. La velocidad se especifica como un valor dentro del intervalo [0, 1], permitiendo así que sea expresada como valores con decimales (0,5 para moverse a “media máquina”) o como una fracción (siendo $\frac{1}{4}$, $\frac{1}{2}$ o $\frac{3}{4}$ valores usuales). Si no se especifica la velocidad, se asume un desplazamiento a media potencia (es decir, a 0,5 o $\frac{1}{2}$). La duración se expresa en segundos, dentro del intervalo [0, 60], pudiendo indicarse valores con parte fraccionaria (2,25 segundos, por ejemplo). En el caso que efectivamente se dé una duración, la acción tardará en ejecutarse exactamente el tiempo señalado, deteniendo el flujo normal de control del programa durante ese tiempo. Si en cambio no se da una duración, se asume que el movimiento será infinito y la ejecución del programa se reanuda tan pronto como el robot comience a moverse. Este manejo de la duración permite escribir fácilmente dos clases diferentes de programas: Aquellos que se asemejen a una coreografía se beneficiarán al especificar movimientos con duración, evitando tener que introducir pausas explícitas. Por otro lado, los que deben reaccionar instantáneamente a cambios en el ambiente necesitan tomar decisiones a partir de la información captada por los sensores al mismo tiempo que ajustar el desplazamiento que está realizando el robot, haciendo prohibitivo el uso de pausas.

La biblioteca provee una función de movimiento adicional que permite un control directo de las velocidades individuales de cada motor:

`moverse([izquierda],[derecha],[duración])`: Indica por separado la velocidad para los respectivos motores izquierdo y derecho del robot. La velocidad está expresada dentro del intervalo $[-1, 1]$, donde un valor positivo indica una velocidad de avance mientras que un valor negativo marca una velocidad de retroceso. Una velocidad en cero detiene al motor y es el valor por defecto si no se la indica. La duración se interpreta igual que en los otros comandos de movimiento.

Para cubrir la necesidad de acceder al estado reportado por los sensores del robot, se decidió crear funciones específicas para cada tipo de sensor, dejando en evidencia el origen de la medición obtenida. El robot posee al frente dos sensores ultrasónicos ubicados en los laterales, que son capaces de detectar obstáculos hasta una distancia de 50 cm. Las siguientes funciones reportan sus mediciones:

`sensor_distancia(sensor)`: Retorna la distancia en centímetros al objeto más próximo (o 51 si no se detectó objeto alguno) usando el sensor indicado en el parámetro dado de la forma 'delantero-izquierdo' o 'delantero-derecho' (también se admiten los nombres más generales de 'izquierda' o 'derecha' dado que no hay sensores de proximidad traseros). Como cada medición consume cierto tiempo en realizarse, puede resultar más beneficioso obtener sólo el valor de un único sensor.

`sensores_distancia()`: Retorna una tupla con las distancias provistas por los sensores de proximidad. Es equivalente a invocar dos veces a la función anterior pero involucra un diálogo más corto con el robot.

El robot también posee dos sensores IR en su parte inferior. Como la medición de ambos sensores se realiza simultáneamente, se provee una única forma de obtenerlas:

`sensores_línea()`: Retorna una tupla con las mediciones provistas por los dos sensores IR. Los valores arrojados se encuentran en el rango $[0, 1023]$.

Ciertas acciones aparecen en forma recurrente en los programas que controlan al robot. Si bien todas las funciones que se describirán a continuación pueden ser provistas por la biblioteca estándar de Python, para usarlas es necesario introducir los conceptos de espacios de nombres, módulos y como importarlos (o continuar con el uso de encantamientos mágicos, los que resulta preferible reducir o directamente eliminar). Para evitar esto, se proveen las siguientes funciones:

`esperar(duración)`: Detiene la ejecución del programa haciendo una pausa durante la duración indica, expresada en segundos como en los comandos anteriores.

`tiempo()`: Devuelve el tiempo actual expresado en segundos. Resulta principalmente útil para calcular el tiempo transcurrido haciendo la diferencia entre los resultados obtenidos en dos invocaciones diferentes a esta función.

`al_azar([probabilidad=50])`: Devuelve verdadero un porcentaje de las veces que se la invoca, en forma acorde a la probabilidad dada, expresada en el rango $[0, 100]$. Si no se indica una probabilidad, retorna verdadero el 50% de las veces.

Todas las funciones descriptas fueron presentadas en forma procedural, donde los comandos se traducen en órdenes enviadas a un robot predefinido que, por defecto, es el primer robot local que se haya registrado o, si no hubiera alguno local, será el primer robot disponible en el primer administrador conocido. Si tampoco se hubiera configurado al menos un administrador, la ejecución de cualquier comando para el robot levantará una excepción señalando este problema.

Para controlar a más de un robot a la vez, la biblioteca provee la función `robot()` que, tras su invocación exitosa, cambia el robot al que implícitamente se envían los siguientes comandos. El robot seleccionado forma parte del estado global del programa pero no se tiene acceso explícito para modificarlo.

La función `robot()` también es la puerta de acceso al enfoque orientado a objetos provisto por la biblioteca ya que retorna un objeto que representa al robot identificado con el nombre dado como argumento (sin importar si el robot fuera local o remoto). Dicho objeto provee a través de sus métodos y atributos toda la funcionalidad detallada anteriormente, con exactamente los mismos nombres y parámetros que su contrapartida procedural. El siguiente código contrasta lado a lado el mismo programa escrito usando el paradigma procedural (izquierda) y el orientado a objetos (derecha):

<pre>... robot('Uniqua') avanzar(duración=1.5) esperar(2) rotar_derecha(duración=4.5) ...</pre>		<pre>... uniqua = robot('Uniqua') uniqua.avanzar(duración=1.5) esperar(2) uniqua.rotar_derecha(duración=4.5) ...</pre>
---	--	--

El enfoque orientado a objetos requiere del uso explícito del objeto que representa al robot, volviéndolo más largo de escribir. Desde el punto de vista del aprendizaje, involucra más conceptos que el procedural, en donde no se necesita hablar de objetos ni explicar la diferencia entre invocar a un método e invocar a una función.

En contrapartida, cuando en un programa se debe controlar a más de un robot a la vez, el enfoque orientado a objetos resulta más simple de razonar, dado que cada comando indica explícitamente a qué robot va dirigido. En el enfoque procedural, los comandos son enviados al último robot seleccionado, que forma parte del estado global implícito del programa, causando que no sea obvio quién los ejecutará. Esto se ve agravado cuando se debe cambiar constantemente de robot para poder intercalar el envío de órdenes a cada uno de ellos: No es posible determinar con qué robot se está trabajando en cada momento sin seguir el código de principio a fin. Cabe remarcar que si se trabaja con un único robot, la cuestión anterior pierde importancia dado que como el robot es siempre el mismo, se lo puede considerar efectivamente como una constante global e ignorarlo en la interpretación de los comandos.

4 Detalles de implementación del sistema de control

La comunicación por bluetooth entre el administrador y los robots es regida por un protocolo propio desarrollado específicamente con el objetivo de ser simple y austero pero cubriendo todas las necesidades actuales, previendo ser extendido en el futuro si se amplían las capacidades del robot. Codifica cuatro comandos fundamentales:

- Arranque: Tras encenderse, el robot se inicia ignorando todos los comandos recibidos excepto éste, que lo habilita para actual ante el resto de los comandos.
- Finalización: Detiene los motores del robot y lo vuelve al estado inicial.

- **Movimiento:** Indica con qué velocidad se debe accionar a cada rueda del robot. Las velocidades se dan en el rango [-255, 255] donde un valor positivo hace que la rueda avance mientras que uno negativo la hace retroceder y cero la detiene. El comando incluye una duración en el rango [0, 60000] representando el tiempo en mili-segundos durante el cual se lleva a cabo el movimiento. Se admite el valor especial 65535 (o FFFFh) que representa una duración infinita.
- **Sensores:** Encuesta el estado de los sensores del robot. Es posible recibir en la respuesta información sobre todos o sólo un subconjunto de los sensores.

Los comandos de movimiento y de sensores son respondidos inmediatamente tras ser recibidos, confirmando la recepción exitosa. Adicionalmente, para los comandos con duración finita se envía un segundo mensaje cuando termina su ejecución.

Para implementar la comunicación entre el administrador y sus clientes se utiliza la biblioteca ØMQ [8], que provee una abstracción sobre los sockets de TCP/IP, facilitando la creación de protocolos de alto nivel y simplificando la tarea de implementar distintos patrones de comunicación. La información intercambiada a través de los sockets es codificada en formato JSON [9] para que sea fácilmente generada y/o consumida desde distintos dispositivos o lenguajes de programación.

Cuando el administrador entra en actividad, éste abrirá dos sockets ØMQ para interactuar con los potenciales clientes: El primero, ubicado en el puerto 7060, emplea un modelo de comunicación de tipo REQ/REP de ØMQ (de pedido y respuesta), donde se recibe un pedido asociado con la administración de los robots para luego enviarse la respuesta con el resultado antes de volver a atenderse el siguiente pedido.

Los pedidos reconocidos son: Obtener los nombres de los robots, solicitar el acceso a un robot en particular, suspender el envío de comandos a un robot, deteniéndolo y reanudar el envío de comandos a un robot previamente suspendido.

El administrador abre adicionalmente un segundo socket ØMQ ubicado en el puerto 7061, que emplea un modelo de comunicación del tipo PUB/SUB de ØMQ (de publicación y suscripción). A través de este socket se distribuyen notificaciones sobre cada comando ejecutado junto con su resultado. El propósito de este socket es diseminar toda la actividad que fluye a través del administrador, con el fin de ser usada por programas monitores o de depuración. El modelo PUB/SUB permite suscribirse para recibir todas o sólo un subconjunto de las posibles notificaciones.

Cuando se solicita el acceso a un robot, el administrador abre un nuevo socket ØMQ ubicado en un puerto al azar, informándose al cliente que desea controlarlo. A través de ese socket de carácter transitorio, que emplea un patrón de comunicación REQ/REP, es que la biblioteca finalmente puede enviar órdenes al robot. Las órdenes tienen una correspondencia directa con las instrucciones que se describieron en la sección 3. Actualmente se emplea una política permisiva con respecto al envío de comandos, aceptando que dos o más clientes envíen comandos simultáneamente al mismo robot. Los comandos serán ejecutados en un orden secuencial FIFO, es decir, que el primero en llegar será el primero en ser atendido, pero no habrá superposición en la ejecución de los mismos. Esto puede traer consecuencias no deseadas si se ejecutan dos o más programas simultáneamente, ya que probablemente ninguno de ellos logrará el comportamiento esperado.

5 Conclusiones y trabajo futuro

El sistema presentado en este artículo permite lograr el control de robots diseñados con fin educativo utilizando una interfaz simple y uniforme, tanto desde un enfoque procedural como desde uno orientado a objetos. Oculta al usuario todos los detalles relacionados con la comunicación inalámbrica y la coordinación con varios robots simultáneamente. Permite además que la organización que posea los robots elija la mejor forma de utilizarlos, adaptándose a las posibilidades y necesidades de la misma.

Como trabajo a futuro, se desea validar el uso del sistema en situaciones complejas, para detectar posibles debilidades que pudieran surgir en la arquitectura desarrollada. Se estudiará si existe la necesidad de incorporar algún mecanismo de autenticación y seguridad al sistema, usándolo para otorgar o revocar el uso exclusivo de los robots a determinados usuarios durante cierto tiempo, en contrapartida al uso actual más libre y permisivo aunque también más anárquico o necesitando de una coordinación ad-hoc. También resulta de interés crear ambientes integrados de desarrollo específicos para el robot junto con la exploración de otros lenguajes de programación desde los cuales controlar al robot, analizando sus fortalezas y debilidades comparándolas con el actualmente elegido. Se está comenzando a trabajar en la aplicación de lenguajes gráficos visuales basados en bloques como lenguaje alternativo, proveyendo además un puente de traducción automática entre los bloques visuales y el código textual de Python 3. Finalmente, queda planteado el desafío de estudiar la ampliación de la arquitectura para que la administración y control de los robots sea a través de internet, incorporándolos a un laboratorio virtual.

Referencias

1. Osella Massa, G., Álvarez, E., Useglio, G., Luengo, P., Llanos, E., Sarobe, M., Russo, C.: Programando y Pensando Robots en la Universidad y en Escuelas Secundarias. En el 1er. Congreso Nacional de Ingeniería Informática / Sistemas de Información (CoNaIISI 2013).
2. Repositorio conteniendo el código fuente del proyecto: <https://github.com/gosella/edubots>
3. Guido van Rossum et al, "The Python Language Reference", Python Software Foundation; <https://docs.python.org/3/reference/index.html>
4. Programando con robots, LINTI, UNLP, <http://robots.linti.unlp.edu.ar/>
5. Díaz, F.J., Banchoff, C.M., Martín, E.S., & López, F.: Aprendiendo a programar con juegos y robots. VII Congreso de Tecnología en Educación y Educación en Tecnología (TE&ET 2012).
6. Guo, P.: Python is Now the Most Popular Introductory Teaching Language at Top U.S. Universities. BLOG@CACM, Communications of the ACM: <http://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-us-universities/>
7. El estándar de codificación de caracteres del Consorcio Unicode: <http://unicode.org/>
8. ØMQ Distributed Messaging. The ZeroMQ Reference Manual: <http://api.zeromq.org/>
9. JSON (JavaScript Object Notation) data-interchange format: <http://www.json.org/>



XIII -Workshop Ingeniería de Software (WIS)

Comité Científico:

Astudillo, Hernán (Chile)
Barbosa, Luis (Portugal)
Dujmovic, Jozo (USA)
Fillottrani, Pablo (Argentina)
Genero, Marcela (España)
Janowski, Tomasz (Naciones Unidas)
Navarro Martin, Antonio (España)
Olivas Varela, Jose Angel (España)
Piattini, Mario (España)
Riesco, Daniel (Argentina)
Rossi, Gustavo (Argentina)
Ruiz, Francisco (España)
Sousa Pinto, Jorge (Portugal)



XIII -Workshop Ingeniería de Software (WIS)

ID	Trabajo	Autores	
9103	Modelo de Proceso para la Evaluación y Selección de Soluciones Software de Gestión Logística	Ana María de Keravenant(UTN-FRBA), Pablo Pytel (UTN-FRBA), María Florencia Pollo Cattaneo (UTN-FRBA)	465
9118	Communication Issues in Agile Software Development	Salvador Esquivel (UNLP)	475
9121	An HCI quality attributes taxonomy for an impact analysis to interactive systems design and improvement	Fernando Pincirolo (UCH)	485
9122	Framework para la creación y ejecución de pruebas automatizadas sobre servicios REST	Maximiliano Mascheroni (UNNE), Emanuel Irrazábal (UNNE)	495
9173	Propuesta de Esquemas de Perfiles para la Recuperación de Datos Científicos para un Sistema de Recuperación de Información del Área de Ciencias de la Computación	M. Rey (UNaM), Horacio Kuna (UNaM), E. Martini (UNaM), Ana Valeria Canteros (UNaM), Alejandro Lucas Cantero (UNaM), A. Rambo (UNaM), Claudio Omar Biale (UNaM), Zamudio (UNaM)	505
9197	Modelado y ejecución de procesos de negocio con conceptos de Green IT	Anahí Rodríguez (UNLP), Patricia Bazán (UNLP), Viviana Ambrosi (UNLP), Javier Diaz (UNLP)	515
9207	Quality Evaluation in Agile Process: A First Approach	Noelia Pinto (UTN-FRRe), Cesar Acuña (UTN-FRRe), Liliana Raquel Cuenca Pletch (UTN-FRRe)	525
9223	Metodología para el desarrollo de software en proyectos de I+D en el nivel universitario basada en Scrum	Luciano Straccia (UTN-FRBA), Pablo Pytel (UTN-FRBA), María Florencia Pollo Cattaneo (UTN-FRBA)	535
9230	Implantación de Sistemas: Estudio Comparativo e Identificación de Vacancias en Metodologías Usuales	Marisa Panizzi (UNLA), Alejandro Hossian (UNLA), Ramon Garcia Martinez (UNLA)	546



ID	Trabajo	Autores	
9232	Propuesta de Artefactos para el Subproceso de Gestión del Modelo de Proceso de Proyectos de Explotación de Información (G-MoProPEI)	Sebastian Martins (UNLA), Patricia Pesado (UNLP), Ramon Garcia Martinez (UNLA)	556
9238	Proposal for Optimizing the Design of the National University of La Plata Service Cloud	Nahuel Cuesta Luengo (UNLP), Claudia M. Banchoff Tzancoff (UNLP), Claudia Queiruga (UNLP), Christian Rodríguez (UNLP)	566
9240	Flip-Flop. Aplicación de Buenas Prácticas a partir de la Gamificación	Facundo Rodríguez (UNLP), Franco Vannasaeng (UNLP), Silvia Esponda (UNLP), Ariel Pasini (UNLP), Patricia Pesado (UNLP)	575
9252	Búsqueda de Funcionalidades basada en Expansión de Consultas para LPS	Maximiliano Arias (UNCOMA), Agustina Buccella (UNCOMA), Alejandra Cechich (UNCOMA)	585
9265	Definición de una Herramienta de Calificación y Priorización de Nuevos Requerimientos de Software por medio de la Valoración de los Clientes	Sara Ester De Federico (UTN-FRRo), Javier Gago (UTN-FRRo), Mariela Avogradini (UTN-FRRo), Lucía Melfi (UTN-FRRo), Silvio Gonnet (UTN)	595
9271	Metodologías para el desarrollo de software en PYMES	Marcelo Alberto Colombani (UNER), Martín Mauricio Pérez (UNER), Marcelo A. Falappa (UNS)	605
9294	Evolución de F/OSS: estudio de caso múltiple sobre el crecimiento	Jorge Ramirez (UNSa), Carina Jimena Reyes (UNSa), María Laura Massé Palermo (UNSa), Nilsa Sarmiento (UNSa)	616
9311	Taxonomía de requisitos para aplicaciones web	Silvia Sanchez Zuaín (UCSE), Elena Durán (UNSE)	626
9313	Una metodología para evaluar la madurez de servicios universitarios	Ariel Pasini (UNLP), Elsa Estevez (UNU), Patricia Pesado (UNLP)	636
9318	Computer Aided Verification of Relational Models	Claudia Necco (UNSL), José N. Oliveira (UMinho), Joost Visser (SIG), Roberto Uzal (UNSL)	647

Modelo de Proceso para la Evaluación y Selección de Soluciones Software de Gestión Logística

Ana María de Keravenant^{1,2}, Pablo Pytel^{1,2}, & María Florencia Pollo-Cattaneo^{1,2}

¹ Programa de Maestría en Ingeniería en Sistemas de Información. Facultad Regional Buenos Aires. Universidad Tecnológica Nacional. Argentina

² Grupo de Estudio en Metodologías de Ingeniería en Software (GEMIS). Universidad Tecnológica Nacional. Facultad Regional Buenos Aires. Argentina.
anadekeravenant@yahoo.com.ar, ppytel@gmail.com, flo.pollo@gmail.com

Resumen. Frecuentemente las empresas invierten gran cantidad de tiempo, esfuerzo y dinero en encontrar soluciones que les permitan reformular sus procesos de negocio para mantenerse a la vanguardia del progreso y alcanzar mayor rentabilidad económica ante las constantes e imprevisibles fluctuaciones del mercado. La oferta de nuevas tecnologías y herramientas disponibles para tal fin es sumamente amplia, y la tarea de escoger la alternativa que mejor se adapte a las necesidades de cada empresa envuelve un gran desafío. Si bien la tendencia del mercado en estos últimos años se ha visto inclinada hacia la implementación de Sistemas de Gestión ERP, la elección de la solución más adecuada no es una tarea sencilla y en la actualidad se continúan detectando múltiples casos de insatisfacción en su implantación. Es en este contexto, en el cual se propone la definición de un procedimiento para evaluar y seleccionar un Sistema de Gestión acotando el ámbito de implantación al de las empresas ocupadas en la Gestión Logística, más conocidas como Operadoras Logísticas.

Palabras Claves: Ingeniería de Software, Evaluación y Selección Software, ERP, BoB, Operadores Logísticos.

1. Introducción

Si bien la Logística existe desde el inicio de la civilización humana tiene su origen formal en el ámbito de la Ingeniería Militar. La publicación más reconocida que hace mención a la logística es la obra "*El Arte de la Guerra*" de Sun Tzu escrito por Sun Wu General de la dinastía China Chu, que fue escrita hace más de 2000 años [1]. Durante la Primera y Segunda Guerra Mundial la logística tomó especial relevancia cuando se la empleó para movilizar recursos bélicos a puntos estratégicos. De hecho, la invasión a Europa durante la Segunda Guerra es considerada una de las operaciones logísticas mejor planeadas de la historia [2]. Actualmente, su importancia es reconocida, como lo indican los autores Ferrel, Hirt, Adriaenséns, Flores y Ramos [3], dado que es "una función operativa importante que comprende todas las actividades necesarias para la obtención y administración de materias primas y componentes, así como el manejo de los productos terminados, su empaque y su distribución a los clientes". Esto significa que gestionar la logística de una empresa en forma eficiente

permite reducir considerablemente sus costos, mejorar las ventas y la operatividad de los procesos productivos, así como incrementar su competitividad en el mercado e, incluso, mejorar su imagen hacia el cliente presentando numerosas ventajas [4].

En este escenario, es posible detectar la importancia que posee utilizar soluciones software apropiadas para que las Operadoras Logísticas puedan gestionar sus costos eficientemente y así lograr una posible y futura reducción de los mismos. No obstante, y a pesar de las ventajas que otorgan los Sistemas de Gestión, muchas de sus implementaciones han fallado por factores como [5]: inadecuada definición de los lineamientos del proyecto y de los requerimientos funcionales, demasiada familiarización con un producto, expectativas poco realistas, ignorancia de las ventajas difíciles de cuantificar, falta de compromiso, falta de preparación, no comprensión de las ventajas de la automatización, falta de comunicación, y resistencia al cambio, entre otras.

A estas dificultades se le incorpora el hecho que la oferta de nuevas tecnologías es inmensa y no todas ellas aplican de igual manera a un mismo entorno. Por una parte, es posible identificar soluciones “*enlatadas*” o “*generalistas*”, conocidas como ‘Sistemas de Gestión Empresarial’ o, en inglés ‘Enterprise Resource Planning’ (ERP), que agrupan las principales funcionalidades de cada una de las áreas de una empresa [6]. También existen otras soluciones “*especializadas*” que se limitan a brindar la mejor solución de negocio para un grupo reducido de procesos, por lo que son denominadas ‘Lo Mejor del Mercado’ [7], más conocidos como ‘Best of Breed’ o BoB. Por consiguiente, al momento de seleccionar la solución a implementar, se detecta una importante bifurcación inicial que también debe ser considerada y que dependerá del alcance que se le desee dar a la solución a escoger.

En este contexto, en este trabajo se propone el desarrollo de un Modelo de Proceso que permita guiar al Líder de proyecto (o responsable afín) en la difícil tarea de seleccionar la solución software más adecuada y funcional para una Operadora Logística. Para ello, en este trabajo, primero se lleva a cabo un análisis de los métodos existentes para la evaluación de Sistemas de Gestión (sección 2). Luego, se presenta el modelo de proceso propuesto (sección 3) junto con un examen de su aplicación en un caso de estudio (sección 4). Finalmente, se indican las conclusiones y futuras líneas de trabajo (sección 5).

2. Métodos disponibles para la evaluación de Sistemas de Gestión

No siempre la implementación de un Sistema de Gestión es exitosa. Continuamente las empresas invierten gran cantidad de tiempo, esfuerzo y dinero en implementar soluciones software que les permitan mantenerse a la vanguardia del progreso sin perder competitividad [8]. A pesar de ser algo frecuente, escoger la solución más idónea para los procesos de una empresa y que mayor provecho pueda sacar de ellos, no es tarea sencilla [9].

A partir de la recopilación bibliográfica realizada es posible concluir que existen múltiples procedimientos o metodologías que pueden ser empleados para la evaluación y selección de un Sistema de Gestión. A continuación, se presenta a modo de resumen, una breve descripción de los mismos:

- **SHERPA** [10]: Este método, que permite evaluar Sistemas ERP u otro tipo de Sistema de Gestión existente, prioriza diferentes criterios definidos en función de la experiencia y a partir de una serie de refinamientos sucesivos recomienda la solución más adecuada para la empresa en donde se desea implementar.
- **MSSE** [11]: La metodología MSSE se basa en la metodología SHERPA pero incorpora cuestionarios para cada módulo ERP a fin de determinar los requisitos funcionales y generales que deben ser evaluados en cada área de negocio. Permite también seleccionar el equipo de consultoría.
- **Selección multicriterio mediante AHP y ANP** [12]: Con este método se identifican una serie de criterios y atributos relacionados con las alternativas de solución a implantar, para a partir de ellos armar una jerarquía (AHP) y/o una red de atributos (ANP) a evaluar, a fin de conocer cual es la solución mas apropiada a escoger.
- **A Selection Model for ERP System by Applying Fuzzy AHP** [13]: En este modelo se evalúan 32 criterios derivados de los requisitos de productos y de gestión. Se emplean para la evaluación las características de calidad ISO 9126.
- **A six-stage model of the buying process for ERP software** [14]: Identifica seis procesos distintivos que se aplican en forma iterativa para la evaluación de los proveedores y de las alternativas ERP tanto desde el punto de vista funcional como del técnico.
- **A framework for evaluating ERP projects** [15]: Propone un marco para evaluar la viabilidad de los proyectos ERP combinando el aprendizaje participativo (Técnica Nominal de Grupo - TNG) y los procesos de toma de decisiones (Proceso de Jerarquía Analítica - AHP).
- **A framework for the ex-ante evaluation of ERP software** [16]: Define un marco de cuatro fases que tiene en cuenta la estimación de los costos necesarios para la adquisición, implementación y mantenimiento del Sistema.
- **Proceso de selección de un ERP en una empresa manufacturera mediana** [17]: Los autores dividen la implementación de un ERP en dos fases, la fase de Selección y la de Implementación. La primera de ellas y que es la de interés para este estudio, se subdivide a su vez en diez etapas que mediante su implantación secuencial favorece la selección de la solución ERP.

A pesar de que en los métodos identificados se definen una serie de pasos a seguir para realizar la evaluación y selección de un Sistema de Gestión, no se especifica qué información debe ser relevada según el entorno en donde se implementará [18]. Si a esto, se le suma que la oferta de herramientas disponibles es inmensa y no todas ellas se pueden aplicar de igual manera a un mismo entorno, entonces la selección se convierte en un proceso largo y complejo de llevar a cabo [19]. Un proyecto de implementación puede fracasar si el entorno sobre el cual actuará no ha sido comprendido de forma correcta [20], y esto es evidente dado que cualquier software se modela a partir de la definición de los procesos de la empresa en donde se utilizará. La implantación de una solución de gestión inadecuada para un determinado entorno, puede impactar severamente en los costos, tiempos y sobre todo en la imagen que tiene la empresa ante sus clientes [21]. Esto queda reflejado en los resultados de encuestas realizadas en [22] donde se puede observar la tasa de proyectos con

demoras de costos y tiempo en 2014 han aumentado con respecto al año anterior en un 1% y 3% respectivamente.

Asimismo, del análisis realizado se observa que todos los métodos existentes han sido definidos para ser aplicados a cualquier tipo de ambiente de negocio manteniéndose independientes a éste. Es decir que, por ejemplo, para una empresa autopartista se propone el mismo procedimiento de selección de su Sistema de Gestión que para una empresa envasadora (siendo que sus procesos de negocio son completamente distintos así como el producto final obtenido de ellos). Al trasladar esta problemática al entorno logístico, y más específicamente al de las Operadoras Logísticas, se encuentra incompatible aplicar dichas metodologías. Una Operadora Logística se halla abocada a la gestión de los procesos tercerizados de empresas con diferentes ramos de negocio por lo cual debe adaptarse a los procesos de suministro de sus clientes, donde cada proceso tiene características particulares dependiendo del ramo de negocio y de la empresa. Por lo tanto, la metodología que lleve a cabo la tarea de evaluación y selección de su Sistema de Gestión debe tener la flexibilidad adecuada para darle soporte. Partiendo de esta hipótesis, se considera apropiado el planteamiento de un procedimiento que identifique y analice las características particulares de este entorno y proponga a partir de ello el Sistema de Gestión que mejor se adapte a él.

Al mismo tiempo, también es necesario diferenciar aquellos Sistemas de Gestión que son del tipo ERP de los que son los “Mejores del Mercado” (BoB), a partir del alcance y las limitaciones de cada uno de ellos sobre los procesos logísticos definidos. Esto último se propone sobre la base de que un Sistema de Gestión del tipo ERP no tendrá el mismo alcance y profundidad en sus funcionalidades que uno del tipo BoB, desarrollado específicamente para un área o sector de la empresa [23]. Por lo cual, se considera apropiado que el procedimiento a utilizar no solo permita diferenciar la solución a escoger por su adaptación a los procesos de la empresa, sino también por el tipo de herramienta y su alcance sobre el negocio. En conclusión, el Modelo de Proceso propuesto estará orientado hacia el entorno logístico y contemplará solamente la información que es importante relevar sobre este ambiente y sobre la tecnología en que las herramientas disponibles se encuentran implementadas.

3. Modelo de Proceso propuesto

A partir del problema identificado en la sección anterior, se propone la construcción de un modelo de procedimiento para la evaluación y selección de la solución de negocio a implantar en una Operadora Logística que desea reemplazar su Sistema de Gestión para mejorar así sus oportunidades de negocio.

El modelo de proceso está formado por tres fases que permiten identificar los requerimientos de la organización (fase I) y las características de las herramientas existentes en el mercado (fase II), para así evaluar y seleccionar la mejor alternativa de solución (fase III). Debido a la amplitud del Modelo de Proceso, a continuación sólo se describen brevemente cada una de sus fases junto a sus actividades y tareas. Su descripción detallada se encuentra disponible en el reporte técnico [24] en la cual se detallan todas las plantillas de los entregables y la explicación de las técnicas utilizadas en cada actividad y tarea.

○ **Fase I: Establecer las Bases del Proyecto**

En esta fase se realizan las actividades necesarias para definir las bases del proyecto, recolectando la información que será utilizada en las siguientes fases. Para ello, se llevan a cabo las siguientes actividades cuyas tareas son resumidas en la Tabla 1.

• **Actividad 1- Definir lineamientos del proyecto**

Habitualmente las Operadoras Logísticas deciden embarcarse en proyectos complejos de migración de sus Sistemas de Gestión para obtener más oportunidades de negocio, ofrecer servicios de mayor calidad y lograr posicionarse mejor de cara al mercado. Para ello, en esta actividad primero se define el objetivo y alcance así como los requerimientos generales del proyecto (tarea 1.1). Luego, se identifican los recursos que estarán involucrados en su realización (tarea 1.2) para los cuales se debe definir el plan de trabajo incluyendo los tiempos y costos estimados para la implementación junto con las procedimientos, metodologías o estándares internos a utilizar (tarea 1.3).

• **Actividad 2 – Relevar estado actual**

En esta actividad se lleva a cabo la recopilación de las características generales que presenta la empresa (tarea 2.1) así como de los procesos operativos involucrados en el proyecto y del Sistema de Gestión que es utilizado actualmente (tarea 2.2). Esto se realiza usando un Cuestionario Web suministrado en [25], para guiar a las personas involucradas en esta tarea. Toda esta información es documentada en la tabla de ‘Características Logísticas’ para ser utilizadas en las siguientes fases. Dicha tabla (junto con los otros entregables) se encuentra definida en el anexo A de [24].

• **Actividad 3 - Definir estado deseado**

En la última actividad de esta fase, se identifican las funcionalidades adicionales que la solución a implantar debería incluir para proporcionarle una ventaja competitiva a la Operadora Logística (tarea 3.1).

Tabla 1: Resumen de técnicas aplicadas en la Fase I.

TAREA	PRODUCTOS ENTRADA		TÉCNICA	PRODUCTOS SALIDA	
	Entrada	Entregable		Salida	Entregable
1.1. Definir requerimientos del proyecto	Necesidad de cambio u oportunidad de negocio	-	<i>Reunión de inicio de proyecto</i>	Lanzamiento formal del proyecto	Acta de Proyecto
1.2. Definir recursos involucrados	Necesidad de definir los recursos involucrados	Acta de Proyecto	<i>Evaluaciones conocimiento & Entrevistas Cualitativas</i>	Recursos involucrados en el proyecto	Listado de recursos
1.3. Definir Plan de Trabajo	Necesidad de planificar el nuevo proyecto	Acta inicio / Listado de recursos	<i>Lluvia de ideas</i>	Planificación de actividades	Plan de trabajo
2.1. Identificar características generales	Necesidad de nivelación de conocimiento	Evaluaciones de conocimiento	<i>Reunión Informal & Cuestionario Web</i>	Características de la empresa	Cuestionario web / Descripción de la empresa
2.2. Identificar procesos operativos	Necesidad de nivelación de conocimiento	Cuestionario Web / Descripción de la empresa	<i>Reunión formal</i>	Procesos operativos	Características Logísticas / Especificación funcional
3.1. Identificar oportunidades de negocio	Necesidad de abarcar mayor mercado	Características Logísticas / Especificación	<i>Lluvia de ideas & Entrevista</i>	Oportunidades funcionales	Nuevas Oportunidades funcionales

○ **Fase II: Identificar las Alternativas de Solución**

Una vez definida la Operadora Logística, se procede a identificar el universo de Sistemas de Gestión (ERP y/o BoB) disponibles en el mercado. Para ello, se llevan a cabo las siguientes actividades cuyas tareas son resumidas en la Tabla 2.

• **Actividad 1 –Identificar soluciones disponibles**

Para conocer cuáles son los Sistemas de Gestión que existen actualmente en el mercado, se recomienda asistir a exposiciones o seminarios sobre dicha temática registrando toda la información recolectada (tarea 1.1). Asimismo, en forma complementaria, se deben contactar a los representantes comerciales de las soluciones identificadas para solicitarles información adicional (tarea 1.2).

• **Actividad 2 - Evaluar antecedentes**

Teniendo en cuenta el universo de alternativas obtenido de la actividad anterior, se procede a recopilar información sobre sus antecedentes contactando a otras empresas que las hayan implementado para solicitarles información detallada sobre este proceso y poder, de esta manera, elaborar posteriormente un repositorio de mejores prácticas (tarea 2.1). También se debe consultar a expertos en logística para que evalúen cada una de las alternativas (tarea 2.2). Toda esta información recolectada será consolidada en un repositorio de mejores prácticas (tarea 2.3).

• **Actividad 3: Efectuar selección preliminar**

A partir del universo de alternativas obtenido hasta el momento, en la tarea 3.1 se efectúa una selección preliminar aplicando un conjunto de Reglas de Decisión definidas ad-hoc teniendo en cuenta las características de las Operadores Logísticas relevadas en la fase anterior. Como resultado de esta pre-selección se busca obtener, preferentemente, no más de 4 soluciones finalistas que serán evaluadas con mayor detalle en la fase siguiente.

Tabla 2: Resumen de técnicas aplicadas en la Fase II.

TAREA	PRODUCTOS ENTRADA		TÉCNICA	PRODUCTOS SALIDA	
	Entrada	Entregable		Salida	Entregable
1.1. Realizar exploración de mercado	Necesidad de explorar el universo de soluciones	Características logísticas / Especificación funcional	<i>Relevamiento del Mercado</i>	Soluciones factibles para el ramo logístico	Planilla de pre-selección de soluciones
1.2. Obtener asesoramiento de representantes comerciales	Soluciones factibles para el ramo logístico	Planilla de pre-selección de soluciones	<i>Prototipado de la Solución de negocio</i>	Universo acotado de soluciones	Informe Comercial
2.1. Investigar implementaciones previas	Universo acotado de soluciones	Informe Comercial	<i>Consulta a Empresas</i>	Universo de soluciones acotado	Evaluaciones de referencias
2.2. Realizar consulta a expertos	Universo acotado de soluciones	Evaluaciones de referencias	<i>Consulta a Expertos</i>	Información de expertos	Informe de experto / Evaluación de conocimiento
2.3. Generar repositorio de mejores practicas	Universo acotado de soluciones	Evaluaciones de referencias/ Informe de experto	<i>Análisis Documental</i>	Mejores practicas	Documento de mejores practicas
3.1. Realizar selección preliminar	Universo de Información relevada	Especif. Req. / Informe experto / Documento de mejores practicas	<i>Reglas de Decisión</i>	Soluciones clasificadas	Listado de soluciones Pre-seleccionadas

Tabla 3: Resumen de técnicas aplicadas en la Fase III.

TAREA	PRODUCTOS ENTRADA		TÉCNICA	PRODUCTOS SALIDA	
	Entrada	Entregable		Salida	Entregable
1.1. Ampliar información de costos y tiempos	Soluciones clasificadas	Listado de soluciones Pre-seleccionadas	<i>Entrevista formal</i>	Información comercial adicional	Discriminación de costos y tiempos
1.2. Ampliar información de criticidad funcional	Información comercial adicional	Discriminación de costos y tiempos	<i>Clasificación características</i>	Funcionalidades priorizadas	Listado de funcionalidades priorizadas
2.1. Identificar grado de inclusión funcional en Costos de Software	Funcionalidades priorizadas	Listado de funcionalidades priorizadas	<i>Clasificación porcentual de Costos</i>	Costos discriminados	Discriminación de costos por priorización funcional
2.2. Identificar grado de inclusión funcional en Tiempos de Implementación	Funcionalidades priorizadas	Listado de funcionalidades priorizadas	<i>Clasificación porcentual de Tiempos</i>	Tiempos discriminados	Discriminación de tiempos por priorización funcional
3.1. Evaluar alternativas según características económico - temporales	Costos y tiempos discriminados	Discriminación de costos y tiempos por priorización funcional	<i>Técnica de Decisión Multicriterio</i>	Alternativa (o subconjunto de alternativas) a implantar	Listado de alternativas calificadas
3.2. Evaluar alternativas según tipo de herramienta de gestión	Subconjunto de alternativa a implantar	Listado de alternativas calificadas	<i>Reglas de Decisión</i>	Herramienta seleccionada	Alternativa final a implantar

○ **Fase III: Evaluar y seleccionar la solución de negocio a implantar**

Esta última fase es la más importante de todo el procedimiento ya que determina, como resultado, el Sistema de Gestión a implantar aplicando la información recopilada en las instancias previas. A continuación se describen sus actividades (las tareas se indican en la Tabla 3).

• **Actividad 1 – Definir datos para la Selección final**

En esta actividad se definen los datos detallados a ser utilizados en las siguientes actividades teniendo en cuenta las alternativas pre-seleccionadas en la fase anterior. Para ello, se debe solicitar información adicional a los representantes comerciales de las alternativas pre-seleccionadas, sobre el detalle de los tiempos y costos de implementación por módulo y funcionalidad (tarea 1.1). En paralelo, se proceden a clasificar las funcionalidades asociadas a los procesos operativos de la empresa en “resignables” y “no resignables” (tarea 1.2).

• **Actividad 2 – Preparar datos para la Selección Final**

Teniendo en cuenta la clasificación de las funcionalidades realizada, se procede a clasificar la información recolectada de cada representante comercial. Esto significa dividir en grupos de funcionalidades “resignables” y “no resignables” los costos de software (tarea 2.1) y los tiempos de implementación (tarea 2.2).

• **Actividad 3 – Seleccionar el Sistema de Gestión**

En la primera tarea de esta actividad (tarea 3.1) se realiza la selección final de la solución a implantar a partir de las alternativas pre-seleccionadas. Para ello, se aplica una adaptación de la Técnica de Decisión Multicriterio [26] que permite asignar una calificación objetiva a cada alternativa teniendo en cuenta la información recolectada

y las necesidades de la Operadora Logística. Aquella alternativa que obtenga la calificación final más alta, se considerará como la más viable de implementar. Sin embargo, puede suceder que dos o más alternativas obtuviesen una calificación similar, por lo que se presenta una evaluación adicional por tipo de herramienta (ERP o BoB) para escoger aquella más conveniente. Ésto se realiza en la tarea 3.2 aplicando otro conjunto de reglas de decisión definidas ad-hoc que evalúan un conjunto de criterios determinados de acuerdo a la Operadora Logística y el Proyecto.

4. Discusión sobre la aplicación del Modelo de Proceso propuesto

Para ilustrar los beneficios de la solución propuesta se considera el caso de la empresa FoxMeyer [27] la cual, siendo la cuarta distribuidora de productos farmacéuticos de Norteamérica, en el año 1995 decide adquirir e implementar un sistema ERP para mejorar su productividad y competencia en el mercado. Sin embargo, el resultado del proyecto ha sido desastroso con pérdidas de más de 100 millones de dólares, llevando la empresa a la bancarrota. Entre las principales causas detectadas para este fracaso se pueden observar muchas asociadas a los factores identificados en [5], que serán examinadas a continuación.

Uno de los principales problemas detectados está asociado a la falta de participación de los miembros del equipo de trabajo (sean empleados o gerentes internos, o consultores externos). Para evitar esto, el modelo de proceso propuesto alienta y persigue la comunicación entre todos los integrantes del equipo durante todas las fases del proyecto. De esta forma, los integrantes no son sólo entrevistados en una fase inicial para luego ser dejados de lado, sino que participan en cada toma de decisiones aportando sus conocimientos objetivos (hechos y datos sobre el funcionamiento actual de la empresa) y subjetivos (opiniones muy valiosas sobre posibles puntos de mejora). Asimismo, se puede notar que el modelo no sólo contempla al personal interno de la empresa sino que también prevé las contribuciones de consultores externos en procesos y/o técnicos para evaluar las características de la Operadora Logística (durante la fase I del proyecto) y las posibles alternativas a implementar (en las fases II y III). Así, se buscarían equilibrar los riesgos en el caso de que el personal interno no se encontrara debidamente capacitado en este tipo de proyectos para tomar las decisiones más acertadas. Esta falta de experiencia también es asistida por la propuesta al incluir procedimientos detallados que indican, paso a paso, cómo se debe llevar a cabo cada tarea.

Por otra parte, otro problema del caso contemplado, ha sido que los usuarios finales del sistema no fueron involucrados en el proyecto por lo que se ‘perdieron’ muchos de los requerimientos, generando diversos problemas al comenzar la utilización del sistema, entre ellos, se destaca la incapacidad del nuevo sistema para manejar el volumen de transacciones requerido. En este sentido, la propuesta realiza varias tareas de identificación de los requerimientos, especialmente durante la fase I, contemplando tanto la situación actual de la empresa (en la actividad 2) como la situación futura deseable (actividad 3) luego de haber determinado los objetivos generales del proyecto y su alcance (actividad 1). Dichos requerimientos no son definidos en forma genérica con plantillas complicadas de completar como suele suceder con los métodos existentes. En la propuesta del presente trabajo no sólo se

determina cuál es la información a recolectar vinculada a los procesos y particulares de una Operadora Logística (como se pueden visualizar en las preguntas del Cuestionario Web), sino que también se incluyen varias plantillas específicas (entre las que se destaca la tabla de “Características Logísticas”) que permiten documentar dicha información. Del mismo modo, se incluye un conjunto de plantillas para documentar la información sobre los Sistemas de Gestión disponibles en el mercado (actividades 1 y 2 de la fase II). Con toda esta información formalizada, se procede a las actividades correspondientes a la pre-selección de las alternativas de solución (actividad 3 de la fase II) y la selección final (actividades de la fase III). Para ello, los requerimientos se ‘transforman’ en “criterios de evaluación” a través de reglas y fórmulas definidas ad-hoc.

Finalmente, se puede notar que hasta el momento no se ha mencionado la diferenciación entre los tipos de herramientas de gestión (ERP o BoB). Si bien pareciera que es trivial decidir cuál es el mejor tipo de sistema que se debe implementar en base a la necesidad del ambiente, ésta no es una tarea tan sencilla de realizar como parece. En el caso de FoxMeyer, ésto nunca fue evaluado dado que la empresa tenía el prejuicio de que un sistema ERP era la solución más adecuada ya que brindaría soporte a los diferentes procesos que conforman la operatoria diaria de sus clientes. Sin embargo, luego, se descubrió que no era así debido a diversas contrariedades que se suscitaron debido al carácter “generalista” del ERP (ya que no llegaba a abarcar todo el universo de funcionalidades solicitadas). Dado que el ambiente de esta distribuidora farmacéutica no era estático, sino que se debía adaptar a los requerimientos de sus clientes, de haber sido aplicado el modelo propuesto, en la actividad 3 de la fase III se hubiera detectado que la mejor opción era seleccionar un BoB desarrollado específicamente, o una combinación de módulos de un ERP con desarrollos adicionales para contemplar las necesidades extraordinarias del contexto.

5. Conclusiones

En el presente trabajo se propone un Modelo de Proceso para la evaluación y selección del Sistema de Gestión más adecuado y funcional para una Operadora Logística. Dicho modelo incluye la identificación de los principales factores asociados a la empresa y el proyecto, considerando las particularidades de su entorno y el alcance que se le desee dar a dicha solución. Asimismo, se consideran las particularidades del tipo de sistema (ERP o BoB), sobre los aspectos relativos a su flexibilidad para satisfacer los requerimientos del ambiente. De esta manera se satisface una necesidad importante de las Empresas Logísticas para mejorar significativamente la eficacia y eficiencia de su operatoria.

Como futura línea de trabajo se prevé aplicar el modelo propuesto en proyectos de significativos correspondientes a Empresas Logísticas de tamaño pequeña o mediana que han resultado en forma éxito o en fracaso. Los resultados así obtenidos enriquecerán al modelo para su adopción por las Pequeñas y Medianas Empresas.

Referencias

1. Tzu, S. & Clavell, J. (2013). *The art of war*. Hachette UK.

2. Ballou, R. (2004). *Logística: Administración de la cadena de suministro*. Pearson Educación.
3. Ferrell, O; Hirt, G; Ramos Garza, L; Rodriguez, M; Cardenas, M. (2004). *Introducción a los negocios en un mundo cambiante*. McGraw-Hill.
4. Martínez, V. (2012). *Estudio en Implementación de un Sistema de Gestión de almacén y logística en una PYME Española*. Universidad Politécnica de Cartagena, España.
5. Ligus, R. G. (2004). *The 12 cardinal sins of ERP implementation*. Rockford Consulting. Feb, 1.
6. Menguzzato Boulard, Martina. (2009). *La Dirección de Empresas ante los retos del siglo XXI*. Universidad de Valencia, España.
7. Engle, P. (2008). *Best of breed - Industrial Engineer*. Academic OneFile.
8. Castells, P; Pasola, J. (2004). *Tecnología e Innovación en la Empresa* (Vol. 148). Univ. Politèc. de Catalunya.
9. Grabski, S; Leech, S; Lu, B. (2001). *Riesgos y controles en la implantación de sistemas ERP*. Partida doble, 11(128), 40-51.
10. Pastor, J. A; Franch, X; Sistach, F. (2002). *Methodological ERP acquisition: the SHERPA experience*. The guide to IT service management, 1.
11. Chiesa, F. (2004). *Metodología para selección de sistemas ERP*. Reportes técnicos en ingeniería del software, 6(1), 17-37.
12. Toncovich, A; Moreno, J; Corral, R. (2007). *Selección multicriterio de un sistema ERP mediante las metodologías AHP y ANP*. Disponible en <http://www.cnc-logistica.org/congreso-cnc/documentos/80.pdf>
13. Lien, C; Chan, H. (2007). *A selection model for ERP system by applying fuzzy AHP approach*. International Journal of the computer, the internet and management, 15(3), 58-72.
14. Verville, J; Halington, A. (2003). *A six-stage model of the buying process for ERP software*. Industrial Marketing Management, 32(7), 585-594.
15. Teltumbde, A. (2000). *A framework for evaluating ERP projects*. International journal of production research, 38(17), 4507-4520
16. Stefanou, C. (2001). *A framework for the ex-ante evaluation of ERP software*. European Journal of Information Systems, 10(4), 204-215.
17. Montagna, J; Leone, H. (2000). *Proceso de Selección de un ERP en una Empresa Manufacturera Mediana*. In First World Conference on Productions and Operation Management, Sevilla, España, 1-11.
18. Muñiz, L. (2000). *ERP, Guía práctica para la selección e implantación*. Edición Gestión.
19. Kolz, R. (2000). *Real-time plant / ERP integration Systems, Man, and Cybernetics*. From EBSCO Business Source Premier. Vol. 3. 2092-2097.
20. Ochoa, M; Britos, P; García-Martínez, R. (2006). *Una Prototipo de Entendimiento del Negocio para Metodologías de Desarrollo de Sistemas*. CACIC XII, San Luis, Argentina.
21. Correa-Espinal, A; Gomez-Montoya, R. (2009). *Tecnologías de la Información en la Cadena de Suministro*. Dyna, 76(157), 37-48.
22. Panorama (2015) *2015 ERP Report, A Panorama Consulting Solutions Research Report*. Disponible en <http://go.panorama-consulting.com/rs/panoramaconsulting/images/2015%20ERP%20Report.pdf>
23. Gelinas, U.; Dull, R. (2010). *Accounting Information Systems*. South – Western Cengage Learning.
24. De Keravenant, A.; Pytel, P.; Pollo-Cattaneo, M.F. (2016) *Descripción detallada de propuesta de Modelo de Proceso para la Evaluación y Selección de Soluciones Software de Gestión Logística*. Reporte Técnico GEMIS-TD-2015-03-TR-2016-06. Disponible en <http://tinyurl.com/h6kc82z>
25. De Keravenant, A. (2016). *Cuestionario Web propuesto para identificar el estado actual de la Operador Logística*. Disponible en <http://tinyurl.com/cuestionarioSitActual>
26. Barredo, J. (1996). *Evaluación multicriterio y Sistemas de Información Geográfica en la Ordenación del Territorio*. Ra-ma. Madrid.
27. Jesitus, J. (1997). *Broken Promises? FoxMeyer's Project was a Disaster. Was the Company Too aggressive or was it Misled?* Industry Week, 31-37.

Communication Issues in Agile Software Development

Salvador Esquivel¹

¹ Facultad de Informática,
Universidad Nacional de La Plata.
salvadoresquivel@hotmail.com

Abstract. Scrum framework is a growing trend in software industry to companies that are looking for development agile ways. In their early days, this methodology required that the working team members were established in a unique room, because it is necessary to have great communication and working together. However, it is a common increasingly practice that the teams are in geographically dispersed places, which means that it is necessary to adapt and/or look for the way that this methodology is suitable in these contexts. There are many experiences in the industry where we can see communication issues due to this kind work. In this paper we will briefly explain the Scrum framework definition, the Global Software Development (GSD) context, and the practices used in different case studies to solve the issues when applying Scrum in GSD.

Keywords: Scrum, Global software Development, Communication issues.

1 Introduction

Scrum framework is a growing trend in software industry to companies that are looking for development agile ways. In their early days, this methodology required that the working team members were established in a unique room, working face to face [1], because it is necessary to have great communication and working together.

However, it is a common increasingly practice that companies applies GSD in their software factories, and still require to continue getting the benefits achieved by agile methodologies, such as reducing the " time to market" and greater flexibility in projects, so they need to look for the way that this methodology is suitable in these contexts.

There are many experiences in the industry where we can see communication issues due to this kind work, and the different alternative solutions that were implemented, leading to analyze the results obtained with this way of increasingly expanded work worldwide.

The scope of the document includes the review of those case studies that describe the problems of communication and the different results to solve them while using

Scrum framework within the context of "Global Software Development" (GSD), and will focus in obtaining the different experiences in the industry in this context.

The remainder of this paper is organized as follows:

Chapter 2: the Scrum framework, characteristics and concepts: roles, events and deliverables.

Chapter 3: the different aspects of GSD.

Chapter 4: the different possibilities in which the Scrum methodology is used in projects with GSD structure are described, and the results of the analysis of the documentation for these cases studies are obtained.

Chapter 5: preliminary conclusions as a result of the description of the previous chapter are formulated and research or work that emerges as consequence of this study is mentioned.

1.1 Research method

In a first step, an Internet research was done taking into account the resulting information from the words "Scrum", "Agile" and "Global Software Development" using www.google.com search engine.

For each of all the papers that were obtained, all their references were recursively searched using <https://scholar.google.com/> search engine.

Then, those articles where Scrum was not in the body of them were discarded.

The remained articles were reviewed, and classified considering the purpose of them in:

"Case studies"

"Description of detected issues"

"Framework for investigation – Existing Literature Review"

From these articles, those that referred specifically to show communication issues in the case experience were selected, reaching the amount of 17 papers that are used to review the experience of the industry to the issue we are evaluating.

2 Scrum

According to the Scrum Guide [2], Scrum is a "framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

Scrum is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques. Scrum makes clear the relative efficacy of your product management and development practices so that you can improve."

It was promoted as a way of reducing time to market, increasing productivity, improving quality and gaining cost effectiveness and efficiency, and has gained significant popularity because of a promise to handle requirements volatility throughout the development life cycle, promotion of extensive collaboration between customers and developers, and support for frequent delivery of a product [3].

The following roles are defined in Scrum:

The *Product Owner* (PO): PO is the customer representative in the Development team. He is responsible for defining the priorities for implementing the various defined requirements, responsible for these definitions, and answers the different questions about them to the team. He is responsible for approving the product increases shown in the sprint review meeting.

The *Scrum Master* (SM): SM is responsible for ensuring the proper implementation of Scrum processes. He is there to serve to resolve all the obstacles that may arise under the project. This role is the closest to the project manager one in cascade method, but with a completely different profile, not taking any decisions or assignments. This role change is one of the most difficult to take when transitioning from cascade development scheme to Scrum one.

Team: It is responsible for the development of the increment of the product. It is self-organized; the team in the planning meeting is who defines responsibilities, estimates and which the outcome of the sprint will be, following the priorities defined by the PO. It refines the requirement definitions with the SM and PO in the Product Backlog Refinement meetings. During execution of the sprint, it meets daily to synchronize efforts and raises all the impediments to SM. It is necessary to team members to have an active participation in the project and become owners of it to be successful, and this is a complex requirement to achieve, considering the natural introvert characteristics of most professionals in software engineering, where they often wait to have their tasks assigned by the manager to perform them.

The defined events in Scrum are:

Sprint Planning: It is divided in two steps: In the first one, the Team analyzes and selects the requirements that will be developed in the sprint. The Team negotiates with the PO which items will be committed to be delivered by the end of the sprint; the “Sprint Goal” is defined. In the second step, the Team produces the list of the necessary tasks to develop the selected requirements and the members self- assigns them. This way, the “sprint backlog” is defined as a result of this process, which is the base to be used in the sprint daily meetings.

Daily meeting: It is a meeting to synchronize efforts that lasts at least fifteen minutes, where each member of the Team responds to the following questions:

¿What did I do from the last meeting?

¿What am I going to do until the next meeting?

¿Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?

Sprint Review: At the end of the sprint, the Team shows the developed software increment. It could have a preliminary presentation, which is valid when the attendants need to have some kind of introduction to the feature, to understand what is going to be shown.

Then, the participants give suggestions and improvements that the PO takes into account to add or not in the pending product backlog to prioritize in future sprints.

Finally, the PO approves or rejects what the Team has developed.

Sprint Retrospective: The Team analyzes the way of working and the different issues in order to increase the productivity.

Product Backlog Refinement: The Team and the PO analyze the product backlog items that will be developed in future sprints, with the objective of having them clearly defined by the time the sprint planning meeting.

The Scrum artifacts are:

Product Backlog: It is a prioritized list of features defined by the PO. It is reviewed in the Product Backlog Refinement meeting and it is used as input in the sprint planning meeting, where, depending of the priority and the needed volume of work, the Team defines a part of this list as the objective of the sprint.

Sprint Backlog: It is the subset of the product backlog that the Team commits to develop in the sprint, along with all the needed tasks to manage. It is created in the sprint planning meeting, and the Team is responsible of it. It is reviewed in the daily meetings, and it is possible to add new tasks as soon as they are discovered, and also to eliminate those that are no longer necessary.

Sprint Increment: The different software components developed during the sprint that are potentially ready to deliver and that are shown in the sprint review meeting in a demo.

3 Global Software Development

According to E. Carmel [4], “the list of features that distinguish global software teams from normal (nonglobal) software teams is short and precise:

- Distance (the distance of developers from each other and from their customers or end-users):

Distance impacts the communication between designer and customer, between two developers, and between development teams and their remote managers. Communication constrains become increasingly significant for software development teams, whether the team’s sites are in the same metropolitan area, within the same country, or cross-ocean.

Distance affects all sorts of coordination and control.

Distance forces most communications into electronic pipelines of various widths and colors. These pipelines are not as rich as face-to-face communication.

- Time zone differences:

These differences exacerbate the communication problem. Almost all the communication is channeled through various asynchronous technologies, such as e-mail or formalized work flow arrangements.

- Local culture (including language, national traditions, customs and norms of behavior).

Perhaps it is the most confusing intra-team feature. It encompasses national and ethnic traditions, customs, norms of culture, as well as language. Cross-culture teams have more potential for productivity as well as more potential for problems, related to that of more homogeneous cultural groups. Problems may stem from mistrust, miscommunications, and lack of cohesion.”

Accordingly, in [5] it is stated that virtual teams :

- Reduce costs by cutting travel expenses and time, creating new "e-economies" of scale, and designing better digital processes.
- Reduce the development cycle evolving from process in series to in parallel, establishing better communications and generating more widespread confidence.

- Increase motivation, allowing more diverse participation, stimulating creativity and encouraging processes and products synergy of new business development
- Promotes learning by capturing knowledge in the natural course of work , and by sharing best practices

Dean Leffingwell [6] states that in scale, all development is distributed development; the opportunity to communicate informally decreases with distance : people located 100 meters or more, has only 5 % chance to talk to each other. By increasing the size of the team, working in the same field it is not a practical reality for most companies. While companies with lots of developers are in a unique city, the size of the organization is such that people are not in the same workspace, the same floor, the same building or the same campus.

In the case of multinational corporations, with teams located in multiple countries, increasing trend of outsourcing, the problem gets more severe.

Related to home office, in [7] it states that the difference from other forms of distributed models , the group is individually and fully distributed , while in other cases small groups are integrated in the same physical place. An advantage of home office is as following: decreasing population of large cities, economizing high cost office space, and giving a chance to the people who are not able to work outside. Nevertheless, the disadvantage of home office includes problems such as burden from less communication, and an exclusion from society. In the software development field, Home Office environments became popular with the growth of Open Source Communities, where people contribute in their free time, from different locations, to build even powerful systems.

4 Scrum in GSD

Most of the success of Scrum is that the team members are located in the same geographic area [8], and the interaction between team members in GSD is becoming difficult. The agile community advocates the importance of the proximity and the relationship between team members [9], which presumes that the implementation of these concepts will be complicated.

The practice of Scrum in these environments is becoming increasingly extensive, even several people argue [9] that the careful use of agile practices in globalized projects and distributed can provide a number of benefits in terms of communication, improved productivity, project management, trust, team motivation , project visibility.

The following models of distributed Scrum teams are proposed:[5]

- Isolated: teams are isolated scattered in different geographies without contact with each other
- Distributed Scrum of Scrum: the teams are isolated and integrated through regular Scrum of Scrum meetings
- Fully distributed Scrum: the teams have their members scattered across geographies.

In this third division we might consider two different aspects:

- A first consideration where, although team members are well dispersed, those who are in a particular place are located in the same office, i.e. attending a common space to work.

- A second consideration where their members, totally or partially, are working remotely from their homes ("home office"), without any other contact except the "virtual", or via the Internet, telephone, etc.

In these contexts, we see the different experiences related to communication problems documented in the case studies of the articles analyzed and the actions taken to solve them:

4.1 Issues

In one of the projects that are described in [10], the meetings should perform using telephone and video conference, forcing to one team to be late evening/night in the office, because there is not enough broadband at members' home.

The lack of face to face communication is a recurring problem in different cases : [10], [11], [12], [9], [13], [14], [15], [1], [16], [17], [21]. Even there is a great difficulty in knowing who is talking as they cannot see the faces [18], [19].

To [8], the reduction of informal contact could lead to lack of awareness of the criticality of the different tasks.

To [15], relationship problems between team members are evident, and they make the mistake of not spreading all the information to the remote Team with the purpose of simplifying, leading to low morale and sense of frustration. It is also detected that when increasing the number of members of the teams, communication becomes more complex, and the setting up of teams if they are not multidisciplinary, is making impossible for them to take ownership of the project.

To [20], video conference is not always available, the conference calls are not good due to the lack of face-to-face communication. The telephone system doesn't work properly.

To [22], even informal communication is difficult, they find different expectations and assumptions related to how to implement Scrum due to the cultural barriers and the language.

In [23], is reported that phone discussions between two sub teams don't work, and takes a lot of time communicate and interact.

To [24], daily meetings become a problem.

In [23] also, they say that conference calls are not understood due to language, so they were replaced by chat and e-mail.

Finally, to [22], they raised expectation and language issues in management meetings.

Table 1. Summary issues:

Issue	# of cases
not enough broadband at members' home	1
Lack of face to face communication	11
Not knowing who is talking	2
Lack of awareness of critically of tasks	1
Not spreading all the information to remote team	1
Video conference tool not available	1
Informal communication difficult	1

Conference calls not understood/don't work	1
Language issues	2
Daily meetings become a problem	1
Relationship problems	1
Increasing the number of members	1

4.2 Solutions

The ways to try to solve these issues are diverse:

To [10], it is necessary to make the team travel from one site to another and vice versa, to develop a whole sprint, trying to have local meetings whenever possible; on the other hand they defined short sprints, having review meetings all together, but splitting retrospective ones, and then having coaches meetings.

Similarly,[16] tried to generate personal relationships , moving offshore developers to onshore, and vice versa, temporarily.

To [9], the dynamic of the Scrum events, daily meetings reviews, retrospectives, refining, help or force to reduce this problem, while increasing the visibility, the communication between the stakeholders, the project coordination.

In [13], they try teams to work in the same physical space whenever possible, even to the extreme that if there are no offices, meet in coffee shops, homes or meeting rooms. In any case, it is essential to use technology for web meetings.

In[14], they use videoconference rooms for daily meetings, and developed a digital board (add on of Jira).

In [15], ambassadors were implemented to improve the communication between the different places, generate trust and let transfer knowledge.

The essential communication problem in [1] is in terms of understanding the requirements, so they decided to perform design and requirement workshops.

In [20], they use a tool called web demo, also they use instant messaging.

It is defined to reduce the frequency of daily meetings to three times a week in [21], with the possibility of doing them at home due to the time differences, and trying to have hour overlapping those days.

They decide to implement the co-SM role in [22], as well as an increment of flights between the sites; additionally, they replaced the onshore architect with an offshore one that was moved onshore during the project to improve the communication. For the second case of the same paper, people onshore are more time offshore and they replace the local coordinator with the on site manager.

In [23], the proposal is to delegate SM role in a person from India that travels to Norway for one sprint. They cancel daily meetings. They divide the requirements in modules to reduce the interdependencies. They begin specifying and documenting the product backlog, and they leave the details to the Team to decide.

In [24], they simply decide to use communication and conference tools while having SM controlling and managing the meetings.

And in [22], it was decided to perform an initial training related to Agile language and methodology to managers.

Table 2. Summary solutions

Issue	# of cases
Members travel from one site to another	2
Short sprints	1
Dynamic of Scrum events	1
Make members work in same place	1
Use video conference room or communication tools	2
Ambassadors	1
Reduce frequency of daily meetings	1
Co-SM role	1
Training	1
Delegate SM role in offshore site	1
Reduce interdependencies between sites	1
Design and requirements workshops	1

5 Conclusions

- Scrum framework immediately shows communication problems that are typical in these contexts.
- Scrum provides the procedural tools to detect and solve them, for example through retrospectives meetings, while projects are active.
- The ways of resolution are extremely varied and depend on the different contexts in which projects operate.
- Several drawbacks often try to be solved through the travel of team members between different locations, either "onshore" to "offshore" or vice versa, strongly constrained by the project budget and result is not always possible.
- Need for communication requires the availability of a technological infrastructure to provide web conference services, good sound quality teleconferencing, environments with tools to share information, which do not have or are limited in many opportunities, especially when working with equipment in emerging countries in Asia or Latin America.
- Most cases are referred to an "onshore" group working with another "offshore" team.
- There are no cases where part or all of the members of each team are in the form of "home office", where entirely all informal face to face communication is absent.
- Many papers claim that the framework itself solves or alleviates the inherent GSD problems.

As a result of this study, the author considers the topic of Scrum in the form of total "home office" as an important line of research, commonly called "virtual teams", where there is no personal face to face interaction between development team members.

06 References

1. Paasivaara M, Heikkilä VT, Lassenius C. Experiences in scaling the product owner role in large-scale globally distributed scrum. In: Global Software Engineering (ICGSE), 2012 IEEE Seventh International Conference on. IEEE; p. 174–8. (2012)
2. K. Schwaber and J. Sutherland, “The Scrum Guide - Scrum-Guide-US.pdf.”, <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100>.
3. Hossain E, Babar MA, Verner J. Towards a framework for using agile approaches in global software development. In: Product-Focused Software Process Improvement. Springer; p. 126–40.(2009)
4. Carmel E.: Global Software Teams. Collaborating Across Borders and Time Zones. Prentice Hall PTR; 269 p. (1999)
5. Lipnack J, Stamps J.: Virtual teams: people working across boundaries with technology. 2nd ed. New York: Wiley; 317 p.(2000)
6. Leffingwell D.: Scaling software agility: best practices for large enterprises. Upper Saddle River, NJ: Addison-Wesley; 349 p. (The Agile software development series).(2007)
7. Luz M, Gazineu D, Teófilo M.: Challenges on adopting scrum for distributed teams in home office environments. World Acad Sci Eng Technol.;59:308–11.(2009)
8. Bannerman PL, Hossain E, Jeffery R.: Scrum practice mitigation of global software development coordination challenges: A distinctive advantage? In: System Science (HICSS), 2012 45th Hawaii International Conference on. IEEE; p. 5309–18.(2012)
9. Hossain E, Babar MA, Verner J.: How Can Agile Practices Minimize Global Software Development Co-ordination Risks? In: O’Connor RV, Baddoo N, Gallego JC, Muslera RR, Smolander K, Messnarz R, editors. Software Process Improvement. Springer Berlin Heidelberg; p. 81–92. (Communications in Computer and Information Science). (2009)
10. Drummond B, Unson JF.: Yahoo! Distributed Agile: Notes from the world over. In: Agile, AGILE’08 Conference. IEEE; 2008. p. 315–21.(2008)
11. Cristal M, Wildt D, Prikladnicki R.: Usage of Scrum practices within a global company. In: Global Software Engineering, 2008 ICGSE 2008 IEEE International Conference on. IEEE; p. 222–6.(2008)

12. Jensen B, Zilmer A.: Cross-continent development using Scrum and XP. In: *Extreme Programming and Agile Processes in Software Engineering*. Springer; p. 146–53.(2003)
13. Karsten P, Cannizzo F.: The creation of a distributed agile team. In: *Agile Processes in Software Engineering and Extreme Programming*. Springer; p. 235–9. (2007)
14. Tietz V, Mönch A.: *Facing Fake-to-Fake: Lessons Learned from Distributed Scrum*. (2015)
15. Hogan B.: Lessons learned from an extremely distributed project. In: *Agile Conference, 2006*. IEEE; p. 6 – pp.(2006)
16. Summers M.: Insights into an agile adventure with offshore partners. In: *Agile, 2008 AGILE'08 Conference*. IEEE; p. 333–8.(2008)
17. Kilpala M, Kärkkäinen T.: Distributed scrum when turning into maintenance: A single case study. In: *Second International Conference on Computer Science and Information Technology (COSIT 2015)* [Internet]. [cited 2016 Feb 3]. p. 55–67. Available from: <http://airccj.org/CSCP/vol5/csit53706.pdf> (2015)
18. Paasivaara M, Durasiewicz S, Lassenius C.: Using scrum in a globally distributed project: a case study. *Softw Process Improv Pract*. 13(6):527–44.(2008)
19. Paasivaara M, Durasiewicz S, Lassenius C.: Using scrum in distributed agile development: A multiple case study. In: *Global Software Engineering, 2009 ICGSE 2009 Fourth IEEE International Conference on* [Internet]. IEEE; [cited 2016 Feb 3]. p. 195–204. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5196933 (2009)
20. Cho J.: Distributed Scrum for large-scale and mission-critical projects. *AMCIS 2007 Proc.* ;235.(2007)
21. Vax M, Michaud S.: Distributed Agile: Growing a practice together. In: *Agile, 2008 AGILE'08 Conference* [Internet]. IEEE; [cited 2016 Feb 3]. p. 310–4. Available from: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4599497(2008)
22. Sadun C.: *Scrum and Global Delivery: Pitfalls and Lessons Learned*. In: *Agility Across Time and Space* [Internet]. Springer; [cited 2016 Feb 3]. p. 71–89. Available from: http://link.springer.com/chapter/10.1007/978-3-642-12442-6_5 (2010)
23. Hole S, Moe NB.: A case study of coordination in distributed agile software development. In: *Software process improvement* [Internet]. Springer; [cited 2016 Feb 3]. p. 189–200. Available from: http://link.springer.com/chapter/10.1007/978-3-540-85936-9_17 (2008)
24. Chalegre V gínia C, Santos WB, de Souza LO, Muñoz HJ., Romero de Lemos Meir a S.: Estudo de Caso da Utilização de Scrum no Desenvolvimento Distribuído de Software. In *PUC - RS*; p. 129–36.(2010)

An HCI quality attributes taxonomy for an impact analysis to interactive systems design and improvement

Fernando Pincioli

Research Institute, Faculty of Informatics and Design, Champagnat University .
Godoy Cruz, Argentina
pincirolifernando@uch.edu.ar

Abstract. In the interaction between users and systems, software quality attributes are mainly involved. When designing interfaces for human-computer interaction different alternatives can be considered in order to obtain the highest quality in an interactive system. However, quality attributes have positive and negative contribution relationships among each other, so that a change in one of them can cause a higher improvement than expected or an unwanted degradation of the system. This is the reason why in this paper we propose a taxonomy of non-functional requirements that can be assigned quality properties susceptible to be measured to propose alternatives that achieve a better quality for the systems. Quality that can be obtained by taking into account the contribution relationships among quality attributes, in order to select those alternatives that provide the biggest gain of system quality for the design and improvement of systems and software interfaces.

Keywords: HCI, quality attributes, quality attributes taxonomy, contribution relationships among quality attributes, quality metrics.

1 Introduction

The design and improvement of human-computer interaction (HCI) is a delicate task and requires a lot of effort. In both cases, for the design and the improvement, alternative solutions are analyzed and one of them is chosen to be finally implemented. The overall quality of the interface will have a value that can be measured by different techniques offered in the literature. However, the greatest difficulty is not in the measurement techniques, but on what should be measured, and we believe that this is due to two main factors: the lack of a taxonomy for quality attributes involved in HCI and the lack of analysis of the positive and negative contribution relationships existing among them.

In this work, we offer a taxonomy of quality attributes for HCI and a perspective of how to analyze the impact of the positive and negative contribution relationships among quality attributes that could enhance an alternative solution or could prevent the degradation in the overall quality of an interactive system after the efforts to improve it.

In section 2, we will present how to measure the quality properties of a set of non-functional requirements that are structured as a taxonomic classification. In section 3, we will review how to analyze the positive and negative contributions that occur among non-functional requirements. Finally, in section 4 we will offer a taxonomy of requirements for the design and improvement of interactive systems that can be used to determine its quality measurements and to analyze the cross-impact among them.

2 Taxonomy of quality attributes to measure software systems quality

Taxonomy is the "science that deals with the principles, methods and purposes of classification"¹, normally used to present information classified in a hierarchical and systematic way. Villegas et al. [1] present a list of the main taxonomies used in the disciplines of Software Engineering and HCI. Other authors offer a good overview of the major classifications over time of quality attributes, also called non-functional requirements [2] [3]. In the particular case of quality attributes for quality in use of software products, which are of particular interest in the area of HCI, one of the most widespread classifications is the proposal by ISO/IEC 25010 [4] standard:

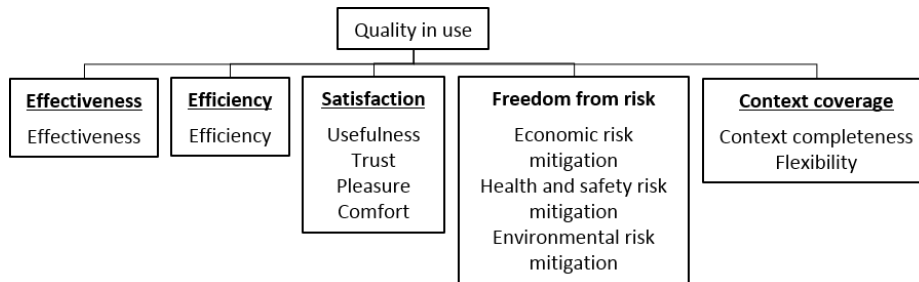
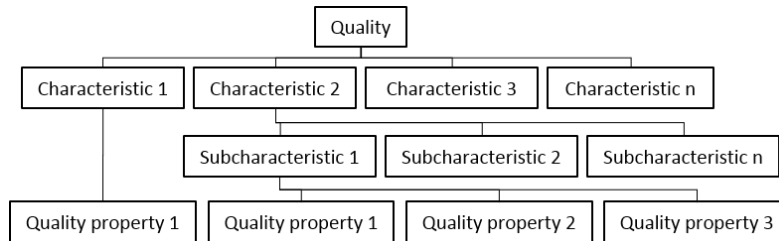


Fig. 1. Quality in use model in the ISO/IEC 25010 standard.

This model depicts a first level of five characteristics and a second level of its main subcharacteristics, which is usually extended with a new level of quality properties, which are the attributes that quality measurements are assigned to when the quality of a software product is objectively measured.



¹ Translated from the Real Academia Espanola's definition.

Fig. 2. Structure used to measure the quality.

Mairiza et al. [5] offer a broad classification of quality attributes with the same hierarchical structure, in which they call "non-functional requirement", "definition" and "attribute" to ISO's "characteristic", "subcharacteristic" and "quality property" respectively.

Taking all things considered, a breakdown structure of quality attributes is performed, where the last level has detailed the properties that will be measured to determine the level of the quality attribute and therefore go on aggregating the results in the higher nodes up to the root node, wherein the overall quality of the system is obtained.

There are different techniques for measuring the properties of the quality attributes of a system that can provide a quantitative assessment for each of them. One of these proposals is GOCAME [6], which stands for Goal-Oriented Context-Aware Measurement and Evaluation. As it can be seen in figure 3, GOCAME is applied to measure the quality of two consecutive versions of a software product. There, the values obtained for the properties of subcharacteristics are combined, then these results are combined among them to get the quality of each characteristic, and finally the same thing is repeated to obtain the overall quality of the software product.

Characteristics and Attributes	JIRA v.1		JIRA v.1.1	
	EI	P/G I	EI	P/G I
1. Actual Usability		53.3%		67.0%
1.1. Effectiveness		73.2%		86.7%
1.1.1. Sub-Task Correctness	86.4%		91.9%	
1.1.2. Sub-Task Completeness	87.9%		95.5%	
1.1.3. Task Successfulness	45.5%		72.7%	
1.2. Efficiency		29.3%		42.8%
1.2.1. Sub-Task Correctness Efficiency	37.4%		44.3%	
1.2.2. Sub-Task Completeness Efficiency	37.5%		47.3%	
1.2.3. Task Successfulness Efficiency	13.1%		36.8%	
1.3. Learnability in use		57.3%		71.6%
1.3.1. Sub-Task Correctness Learnability	78.8%		75.1%	
1.3.2. Sub-Task Completeness Learnability	26.4%		77.3%	
1.3.3. Task Successfulness Learnability	66.7%		62.5%	

Fig. 3. Measurement of Jira's usability in versions 1.0 and 1.1 (taken from [6]).

However, as shown in figure 3, although the usability of the system improved from 53.3% to 67.0% of a Jira's² version to the following one, some subcharacteristics suffered degradation, such as those related to system learnability. Even though it is difficult to ensure, we could think that the increment of the functionality from one version to the next one could have resulted in a greater difficulty to learn them. This is well known as contribution relationships among quality attributes contribution, which can be positive or negative, occurring the latter in the mentioned example.

² Jira is one of the most popular applications in the world for collaborative work, developed by the company Atlassian; <http://www.atlassian.com>

3 Contribution relationships among quality attributes

Quality attributes have positive and negative contribution relationships among each other. This fact, taken into account by the lead authors on requirements, is of enormous importance in establishing solutions to improve the quality of software products [7] [8] [9] [10] [11] [12].

	Availability	Efficiency	Installability	Integrity	Interoperability	Modifiability	Performance	Portability	Reliability	Reusability	Robustness	Safety	Scalability	Security	Usability	Verifiability
Availability																
Efficiency	+			-	-	+	-			-		+			-	
Installability	+							+						+		
Integrity		-			-				-		+		+		-	-
Interoperability	+	-	-			-	+	+		+	-		-			
Modifiability	+	-				-		+	+			+				+
Performance		+			-	-				-		-			-	
Portability		-		+	-	-			+				-	-	+	
Reliability	+	-	+		+	-				+	+		+	+	+	+
Reusability		-	-	+	+	-	+						-		+	
Robustness	+	-	+	+	+	-		+				+	+	+	+	
Safety		-	+	+						+			+		-	-
Scalability	+	+	+			+	+	+		+						
Security	+		+	+		-	-	+		+	+				-	-
Usability		-	+			-	-	+		+	+					-
Verifiability	+	+	+	+	+			+	+	+	+	+	+	+	+	

Fig. 4. Contribution relationships among quality attributes (taken from [7]).

When a need for improvement is detected in any of the quality attributes, it is a mistake to seek improvement solutions only for one attribute regardless of its relationship with the remaining ones. This is due to the fact that the solution implemented to improve one attribute could negatively affect other attributes and degrade them in such a way that a lower overall quality of the product would be obtained.

Thus, when we look for an improvement in the quality of a software product, the combination of positive and negative contributions of the different solutions must be measured in order to choose the one offering the greatest gain in overall quality. We have recently published an article presenting a technique to analyze and to measure the impact of solutions on the overall software quality, to guide the selection of the best alternative in this regard [13].

Our goal is to develop a matrix of contribution relationships to the quality attributes involved in interactive systems, but we found some difficulties on this that we will try to resolve.

4 Taxonomy of quality attributes for HCI

It is difficult to set a taxonomy for quality attributes related to HCI. Different authors, many of them recognized in the field, have different opinions and sometimes even contradictory. One of the difficulties is to set a classification of quality attributes. In the research conducted by Mairiza et al. [5] along the existing literature on quality attributes, they obtained an exhaustive list of them. But at the same time, they found that some authors present as quality attributes what other authors consider properties of quality attributes.

Another difficulty is the diversity of criteria in establishing the quality attributes that influence the acceptability of interactive systems. For example, the ISO/IEC 25010 standard [4] details the quality attributes and their respective subcharacteristic to a system's quality in use. It also indicates which quality properties of software products and computer systems influence on quality in use for primary users of the system. As shown in figure 5, ISO considers that “*functional suitability*”, “*performance efficiency*”, “*usability*”, “*reliability*” and “*security*” of the product have influence on quality in use, while according to that standard “*compatibility*” (besides “*maintainability*” and “*portability*”) does not.

Software product properties	Computer system properties	Product quality characteristic	Influence on quality in use for primary users	Influence on quality in use for maintenance tasks	Information system quality concerns of other stakeholders
☞	☞	Functional suitability	*		
☞	☞	Performance efficiency	*		*
☞	☞	Compatibility		*	
☞	☞	Usability	*		
☞	☞	Reliability	*		*
☞	☞	Security	*		*
☞	☞	Maintainability		*	
☞	☞	Portability		*	

Fig. 5. Product and system quality that influence on quality in use (taken from [4]).

However, if we look at the known Nielsen's classification of acceptability [14], we will find a different opinion.

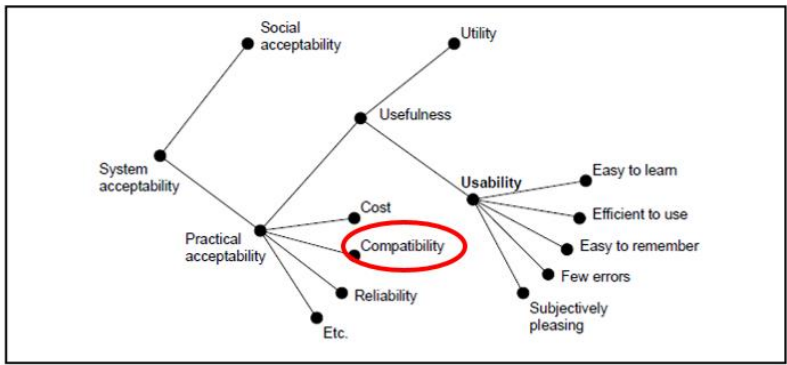


Fig. 6. Nielsen's system acceptability model.

In the classification presented in figure 6, it can be seen that "compatibility" is part of the system acceptability, contrary what is indicated above for quality in use by ISO 25010 standard. For all the aspects previously considered is that we dare to propose a taxonomy of quality attributes for this area. To develop this taxonomy we start from Nielsen's model due to its widespread use and acceptance, but we extend it with a new level of quality subcharacteristics obtained from the following sources:

1. Compilations of the most comprehensive lists of quality attributes presented by different authors, from which we have taken only those attributes that influence the HCI [2] [3] [5] [15].
2. Within the Nielsen's category of "Few errors" we incorporate the "antifragility" quality attribute which is a new concept that considers that the systems must not only be robust, but even strengthened and improved from the impacts they receive [16] [17] [18].
3. We incorporate other HCI specific quality attributes from the material of the PhD course "Design of interactive systems from a user-centered approach", taught by Dr. César Collazos [19].

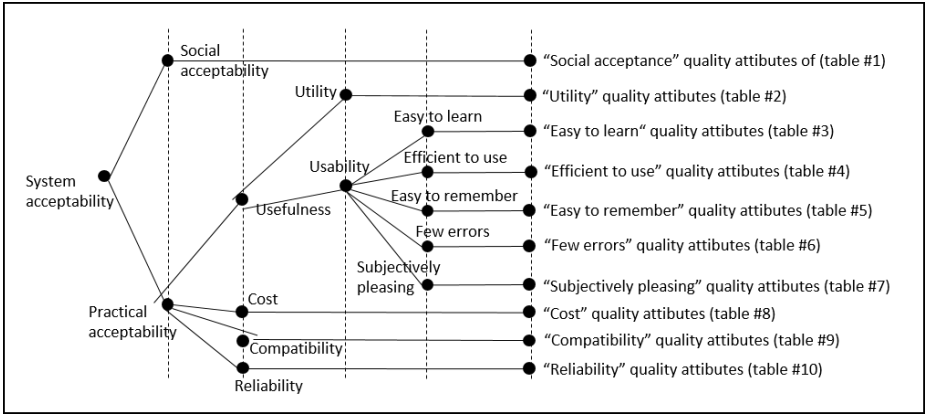


Fig. 7. Extended Nielsen’s acceptability model.

We have extended the Nielsen’s model with a new level of quality attributes, as shown in figure 7. The quality attributes of the new level mentioned in the figure above are detailed in the following explanatory tables.

Table 1. “Social acceptability” quality attributes.

Attributes	
Diffusion	Status

Table 2. “Utility” quality attributes.

Attributes	
Completeness	Functionality
Comprehensiveness	Maturity
Coverage	Service quality
Effectiveness	Suitability

Table 3. “Easy to learn” quality attributes.

Attributes	
Affordability	Natural relationship between controls and their functions
Auto-description	Observability
Available helps	Perception
Communicativeness	Predictability
Comprehensibility	Presentability
Conciseness	Readability
Consistency	Simple design
Content and interaction metaphor	Structuredness
Cultural level	Trainability
Decreased cognitive load	Universality
Degree of technology knowledge	Use of analogies
Easy to use	Use of icons
Expressiveness	Use of standards
Language and communication	Visibility

Table 4. “Easy to use” quality attributes.

Attributes	
Adaptation to different types of environments	Latency
Agility	Layout and organization of controls
Alternatives of use	Layout of the most important information
Availability	Manageability
Comparability	Multiuser architecture
Complexity of interaction	Navigation among windows

Configurability	Navigation inside the windows
Conformance	Operability
Controllability	Physical size of the equipment
Customizability	Reconfigurability
Design for users with cognitive decreases	Remote operation
Design for users with hearing decreases	Repeatability
Design for users with motion decreases	Replicability
Design for users with visual decreases	Response time
Dialogue techniques	Simplicity
Flexibility	Use of system resources
Handling of attention	Use of user resources
Installability	WYSIWYG

Table 5. “Easy to remember” quality attributes.

Attributes	
Experience gain	Mechanisms to help remember where you are
Mechanisms to help remember actions	Memory of use

Table 6. “Few errors” quality attributes.

Attributes	
Access control	Possibility of correcting an error
Accuracy	Possibility of reversing an error
Antifragility	Presentation of correct messages
Correctness	Recoverability
Demonstrability	Restriction indication
Distinction of colors	Stability
Error highlighting	Traceability
Error protection	Uniformity
Fault tolerance	Use of codes besides colors
Feedback of results of actions	Verifiability
Feedback of the actions taken	Visibility of system status
Immunity	Visual structure of information
Integrity	

Table 7. “Subjectively pleasing” quality attributes.

Attributes	
Actualization of technology	Esthetic
Appropriate combination of colors	Fatigue and health
Attractiveness	Formality
Comfort	Gamification
Emotionality	Motivation
Ergonomics	Social context

Table 8. “Cost” quality attributes.

Attributes	
Acquisition cost	Training cost
Cost of consumables	Update cost
Maintenance cost	Upgrade cost

Table 9. “Compatibility” quality attributes.

Attributes	
Coexistence	Mobility
Standardizability	Portability
Generality	Replaceability
Integratability	Transferability
Interoperability	

Table 10. “Reliability” quality attributes.

Attributes	
Anonymity	Privacy
Auditability	Protection
Certainty	Responsibility
Confidentiality	Security
Dependability	Supportability
Non-repudiation	Trustability

Quality attributes presented in the tables above are, then, the basis of quality measurement of interactive systems and of confrontation of their respective contribution relationships in order to determine the best alternative for the design and improvement of these systems.

5 Conclusions and future work

We consider this work as a starting point for the design and improvement of HCI from a taxonomy of the involved quality attributes (tables 1 to 10) that can have applied measurement techniques to obtain a comparable measure of the resulting overall system quality (figure 3) and considering the positive and negative contribution relationships among quality attributes (figure 4) in order to select the design or improvement alternatives providing the best measure of the system quality.

To achieve this there is still much work to be done. The first is to achieve an adequate taxonomy for HCI quality attributes. Different taxonomies and classifications of quality attributes that can be found in the literature are not uniform. In many cases the quality attributes are presented clearly, with sufficient definitions, but other times it is difficult to ensure the concept that aims to convey the author. Moreover, these taxonomies can present different quality attributes groupings, where

sometimes some of them are grouped into each other and in other opportunities they are presented at the same level. This is why we presented our proposal in figure 7.

The second step consists on proposing a matrix of contribution relationships among quality attributes brought from taxonomy obtained in the previous point.

Finally, the definition of the form of measurement and the metrics associated with each quality attribute as the proposal presented in section 2 of this work [6] is pending.

References

1. Villegas, M. et al. "Activity theory as a framework for accommodating cultural factors in HCI studies," *IEEE Lat. Am. Trans.*, vol. 14, no. 2, pp. 844–857, 2016.
2. Afreen, N. et al. "A Taxonomy of Software's Non-functional Requirements". *Proceedings of the 2nd Intl. Conference on Computer and Communication Technologies*, 2015, pp. 47–54.
3. Odeh, Y. and M. Odeh, "A New Classification of Non-Functional Requirements for Service-Oriented Software Engineering," *Nauss.Edu.Sa*, pp. 1–7, 2009.
4. ISO/IEC, "Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models", vol. 2011, 2011.
5. Mairiza, D. et al. "An Investigation into the Notion of Non-Functional Requirements". *Proceedings of the ACM Symp. on Applied Computing*, 2010, pp. 311–317.
6. Olsina, L., P. Lew, A. Dieser, and B. Rivera, "Using web quality models and a strategy for purpose-oriented evaluations," *J. Web Eng.*, vol. 10, no. 4, pp. 316–352, 2011.
7. Wiegers, K. and J. Beatty, *Software Requirements*. Microsoft Press, 2013.
8. Chung, L., B. Nixon, and E. Yu, "Using Non-Functional Requirements to Systematically Select Among Alternatives in Architectural Design". *1st Intl. Work. Archit. Softw. Syst. - Coop. with 17th Int. Conf. Softw. Eng. ICSE 1995*, pp. 31–43, 1995.
9. Chung, L., B. Nixon, E. Yu, and J. Mylopoulos, "The NFR Framework in Action," *Non-Functional Requirements*, pp. 15–45, 2000.
10. Mairiza, D. and D. Zowghi, "Constructing a Catalogue of Conflicts among Non-functional Requirements," *Commun. Comput. Inf. Sci.*, vol. 230, pp. 31–44, 2011.
11. Barbacci, M., et al. "Quality Attributes" *CMU/SEI-95-TR-021*. 1995.
12. Hines, M. and A. Goerner, "Software quality: attributes and modalities". *Trans. Inf. Commun. Technol.*, vol. 11, 1995.
13. Pinciroli, F. "Improving software applications quality by considering the contribution relationship among quality attributes". *3rd Int. Work. Comput. Antifragility Antifragile Eng. (ANTIFRAGILE 2016)*. Elsevier, *Procedia Computer Science*, vol. 83, pp. 970-975, 2016.
14. Nielsen, J. *Usability engineering*. San Francisco: Morgan Kaufmann Publishers Inc., 1993.
15. Masip, L. et al. "User experience specification through quality attributes," *Lect. Notes Comput. Sci.*, vol. 6949 LNCS, no. PART 4, pp. 656–660, 2011.
16. Taleb, N. *Antifragile. Things that gain from disorder*. New York: Random House, 2012.
17. De Florio, V. "Antifragility=Elasticity+Resilience+Machine learning: Models and algorithms for open system fidelity," *Procedia Comput. Sci.*, vol. 32, pp. 834–841, 2014.
18. Jones, K. "Engineering antifragile systems: A change in design philosophy," *Procedia Comput. Sci.*, vol. 32, pp. 870–875, 2014.
19. Collazos, C. "Diseño de sistemas interactivos desde un enfoque centrado en el usuario". *Material of the PhD on Computer Sciences course, Univ. Nacional de San Juan*, May 2016.

Framework para la creación y ejecución de pruebas automatizadas sobre servicios REST

Maximiliano Agustín Mascheroni¹, Emanuel Irrazábal^{1,2},

¹ Universidad Nacional Del Nordeste. Facultad de Ciencias Exactas y Naturales y Agrimensura. Departamento de Informática

² Universidad de la Cuenca del Plata. Facultad de Ingeniería y Tecnología.
{mascheroni, eirrazabal}@exa.unne.edu.ar

Abstract. La arquitectura REST emerge como alternativa al diseño de servicios web, con mayor simpleza que SOAP y los servicios basados en WSDL, haciendo que las pruebas sobre esta arquitectura cobren mayor relevancia. Si bien existen muchas herramientas disponibles, muy pocas pueden integrarse a un proceso de desarrollo continuo de software, donde el tiempo es un factor clave. En este trabajo, partiendo de los principios de esta arquitectura y los fundamentos de pruebas unitarias, se propone un framework para pruebas sobre servicios REST que puede ser integrado a este tipo de entornos. El mismo ha sido implementado en una empresa de desarrollo software multinacional con gran éxito. Se ha comprobado la disminución en un 90% del tiempo para realizar regresiones automatizadas y refactorizaciones en un entorno de integración continua. Finalmente, la utilización de herramientas para ejecutar pruebas automáticas, ha permitido otro tipo de mejoras, como la generación automática de reportes.

Keywords: Pruebas de servicios web, REST, desarrollo continuo de software

1 Introducción

La transferencia de estado representacional (REST - Representational State Transfer) fue ganando adeptos como una alternativa más simple al protocolo de acceso a objetos simples (SOAP - Simple Object Access Protocol) y a los servicios web basados en el lenguaje de descripción de servicios web (WSDL - Web Services Description Language) [1]. La arquitectura REST define un conjunto de principios por los cuales se diseñan servicios web haciendo foco en los recursos del sistema, incluyendo: cómo acceder al estado de dichos recursos y cómo se transfieren por HTTP hacia clientes escritos en diversos lenguajes [2].

La adopción de REST por grandes proveedores de servicios de aplicaciones Web como Google, Yahoo y Facebook o por compañías como Wal-Mart, indican su creciente aceptación [3]. Estas organizaciones consideraron a las tecnologías SOAP e interfaces basadas en WSDL como obsoletas, y decidieron implementar REST como un modelo orientado a los recursos más fácil de usar [4]. En este sentido, la calidad cobra una gran importancia; un servicio web que no ha sido probado puede enviar al

cliente contenido no deseado, malicioso o con errores, en lugar de las estructuras de datos que esperadas [5].

En la actualidad, existe un gran número de herramientas como, por ejemplo, Postman [6], RestClient [7] o Swagger [8], que permiten realizar pruebas sobre servicios REST. Sin embargo, estas herramientas no permiten el desarrollo de pruebas automatizadas ni su integración a servidores de integración continua. Por ello, en este trabajo se propone un framework que utiliza los mismos principios del desarrollo de servicios REST y los combina con los principios de pruebas unitarias. Además de esta sección introductoria, el trabajo se compone en 5 secciones. En la sección 2, se listan los fundamentos y principios de la arquitectura REST. En la sección 3, se describen las herramientas más utilizadas para realizar pruebas de servicios web. En la sección 4 se presenta el framework propuesto y los resultados de su implementación se muestran en la 5. Finalmente, las conclusiones son mencionadas en la sección 6.

2 REST

Un servicio web se define como una aplicación software identificada por un URI cuyas interfaces se pueden definir, describir y descubrir mediante documentos XML [9]. Los servicios web permiten la interoperación de sistemas distribuidos heterogéneos con independencia de las plataformas hardware y software empleadas. Por tanto, puede pensarse en ellos como en una arquitectura, conceptual y tecnológica, haciendo posible que distintos servicios se describan, publiquen, descubran y utilicen a través de sistemas distribuidos, empleando la infraestructura proporcionada por Internet [10].

REST emergió en los últimos años como el modelo predominante para el diseño de servicios. Ha logrado un gran impacto en la web desplazando a SOAP y a las interfaces basadas en WSDL por tener un estilo más simple de usar [2]. Fielding define a REST como “un estilo de arquitectura de software para sistemas hipermedia distribuidos tales como la World Wide Web” [1]. Para otros autores, es un conjunto de principios para el diseño de redes, que es utilizado comúnmente para definir una interfaz de transmisión sobre HTTP de manera análoga a como lo hace SOAP [11].

Una implementación de un servicio REST los siguientes principios [4]:

- **Verbos HTTP:** los más comunes son GET, POST, PUT y DELETE [12].
- **Sin estado:** ni el cliente ni el servidor necesitan recordar ningún estado de las comunicaciones entre mensajes.
- **Identificadores de recursos uniforme (URI):** cada recurso es accedido únicamente a través de su URI.
- **Hipermedios:** permite transferir los formatos HTML, JSON y XML.

3 Pruebas de servicios REST

Según Johnson, la mayoría de las organizaciones han migrado o comenzado a utilizar arquitecturas basadas en servicios REST [13]. Esto ha generado la aparición de un

gran número de herramientas para realizar pruebas sobre estos sistemas, buscando asegurar la calidad de los mismos. A continuación se describen las más utilizadas:

- **Postman** [6]: es un cliente REST que puede integrarse con el navegador Google Chrome. Es una herramienta que permite crear peticiones de manera muy sencilla, para luego poder enviarlas a un servidor y visualizar la respuesta.
- **Rest-Client** [7]: al igual que Postman, es un cliente REST que se integra con el navegador Mozilla Firefox. Permite la construcción de peticiones para probarlas directamente contra un servidor.
- **Swagger** [8]: permite la representación de las APIs REST de un sistema. De este modo, es posible realizar pruebas mediante la generación de peticiones y verificación de las respuestas del servidor a través de su interfaz.
- **SOAP-UI** [14]: es una herramienta multi-plataforma de código abierto para realizar pruebas funcionales sobre servicios REST y principalmente SOAP. Permite crear y ejecutar regresiones automatizadas sobre servicios y también medir los tiempos de respuesta.

Sin embargo, la principal limitación de todas ellas es que requieren de un esfuerzo manual para realizar las verificaciones sobre los datos obtenidos del servidor, para corroborar que sean los esperados. Asimismo, este esfuerzo manual conlleva a que estas herramientas no puedan ser integradas en un entorno de desarrollo continuo de software, donde los elementos principales son las pruebas automatizadas y los servidores de integración continua. Para buscar una solución a estos inconvenientes, surgen propuestas en Git-hub como Restfuse [15] o Rest-assured [16], que aún se encuentran en etapas de desarrollo y solo disponen de versiones beta.

4 Framework Propuesto

Partiendo de las propias herramientas utilizadas para la creación de servicios REST, como, por ejemplo, RestTemplate de Spring [17] y tomando los principios de las pruebas unitarias [18], es posible crear un framework de pruebas para este tipo de servicios. Por ejemplo, los mismos creadores de Spring, lanzan el Spring MVC Test Framework [19], que permite la creación de pruebas unitarias de servicios REST. Al ser pruebas unitarias, tienen la ventaja de integrarse en un servidor de integración continua junto con el código fuente. El problema que presentan estas herramientas es la dependencia: las pruebas se acoplan a la herramienta utilizada, y esto, a su vez, limita su alcance.

El framework propuesto incluye:

1. La creación de una herramienta para interactuar con servicios REST, sin depender de un cliente en particular.
2. El manejo de objetos de transferencia de datos en las peticiones y las respuestas.
3. Un modelo de dos capas para la creación de pruebas.

4.1 Interacción con los servicios REST

En primer lugar, se busca resolver el problema de la dependencia a partir del uso de interfaces. Con la creación de una interfaz es posible brindar métodos para crear las peticiones y utilizar los verbos HTTP, interactuando con los servicios sin importar la implementación. En la Fig. 1, se muestra un ejemplo de una interfaz HTTPClient. Esta interfaz brinda los verbos HTTP con diferentes variaciones y métodos para preparar las peticiones. Las variaciones de cada verbo HTTP sirven para determinar el formato tanto de la petición, como el de la respuesta que se desea obtener. Los formatos permitidos son: un texto plano o un objeto simple con atributos.

```

7 public interface HTTPClient {
8
9     public void setHeader(String field, String value);
10
11     public void setExpectedObjectType(Class<?> type);
12
13     public void addURLQueryParameter(String field, String value);
14
15     public void addBodyParameter(String key, String value);
16
17     public void setCredentialsForBasicAuthentication(String user, String password);
18
19     public HTTPResponse get(String url);
20
21     public HTTPResponse get(String url, Class<?> type);
22
23     public HTTPResponse post(String url);
24
25     public HTTPResponse post(String url, Class<?> type);
26
27     public HTTPResponse post(String url, Object requestBody);
28
29     public HTTPResponse post(String url, Object requestBody, Class<?> type);
30
31     public HTTPResponse post(String url, String requestBody);
32
33     public HTTPResponse post(String url, String requestBody, Class<?> type);
34
35     public HTTPResponse delete(String url);
36
37     public HTTPResponse delete(String url, String item);

```

Fig. 1. Parte de la Interfaz HTTPClient

Cada clase que implementará esta interfaz, tendrá el código para interactuar con los servicios REST de acuerdo al cliente REST utilizado. Habrá tantas implementaciones de la interfaz como clientes REST se quieran utilizar (ver Fig. 2). En el artículo de Bhandari [20], pueden encontrarse una lista de diferentes clientes REST para implementarse con el lenguaje Java. De este modo, cada clase será un envoltorio de un cliente REST. En la Fig. 3 puede verse la implementación de un método GET con sus variaciones, utilizando el cliente RestTemplate de Spring [17].

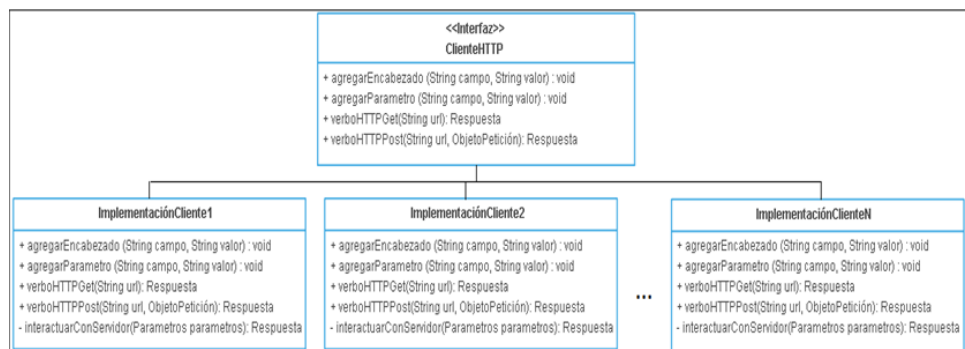


Fig. 2. Diagrama de clases de la implementación de los clientes HTTP

Como puede apreciarse en la Fig. 1 y en la Fig. 3, los métodos que representan verbos HTTP devuelven un objeto `HTTPResponse`. Esta clase es una representación de una respuesta del servidor. Como cada cliente HTTP maneja su propio tipo de objeto “respuesta de servidor”, es necesario hacer una generalización de las mismas para ser utilizada de la misma forma por todas las implementaciones de los clientes. La Clase `HTTPResponse` contiene métodos para obtener de la respuesta: una lista de los encabezados, el código de estado de la petición y el resultado de la interacción con el servidor.

```

1 public class RestTemplateHTTPClient implements HTTPClient {
2
3     //Otras implementaciones
4
5     @Override
6     public HTTPResponse get(String url) {
7         return this.getMethod(url, String.class);
8     }
9
10    @Override
11    public HTTPResponse get(String url, Class<?> type) {
12        return this.getMethod(url, type);
13    }
14
15    private HTTPResponse getMethod(String url, Class<?> type) {
16        //Obtiene la URL con todos los parámetros
17        URI uri = this.getURIWithURLQueryParameters(url);
18        //Crea una entidad-petición RestTemplate usando los encabezados definidos
19        HttpEntity<String> entity = new HttpEntity<String>(this.headers);
20        //Llama a la ejecución del método executeHTTAction con los parámetros
21        return this.executeHTTAction(uri, HttpMethod.GET, entity, type);
22    }
23
24    private HTTPResponse executeHTTAction(URI uri, HttpMethod method,
25        HttpEntity<?> requestEntity, Class<?> responseType) {
26        //Ejecuta la acción solicitada contra el servidor
27        ResponseEntity<Object> response = (ResponseEntity<Object>) RestTemplate
28            .exchange(uri, method, requestEntity, responseType);
29        //Obtiene los encabezados de la respuesta del servidor
30        HttpHeaders responseHeaders = new HttpHeaders(this.getHeaders(response));
31
32        //Genera un objeto HTTPResponse con los datos de la respuesta del servidor
33        return this.createResponse(response.getStatusCode().value(), "OK", response.getBody(), responseHeaders);
34    }
35
36 }

```

Fig. 3. Implementación de un método GET, en una clase de implementación de la interfaz.

4.2 Manejo del Objeto de Transferencia de Datos

Como se menciona anteriormente, tanto las peticiones como las respuestas aceptan dos formatos en sus cuerpos: texto plano o cualquier objeto simple. Una de las maneras de hacerlo es mediante el uso del patrón DTO [21], [22]. Para trabajar con este patrón, es necesario modelar cada entidad que participa en una petición o en una respuesta como un objeto simple.

Para la deserialización, es necesario utilizar alguna herramienta que permita el procesamiento de los formatos que soporta HTTP, como por ejemplo Jackson [23].

La petición puede ser cualquier objeto simple, que a través de una serialización será transformado al formato solicitado por el servidor. Por otro lado, la respuesta también puede ser cualquier objeto simple. De esta manera, se utilizan objetos (DTOs), que contienen en su interior los tipos solicitados.

4.3 Modelo para la creación de pruebas

El modelo propuesto, consiste en separar la lógica de interacción con los servicios web de la capa de pruebas. Una de las principales razones es cumplir con el patrón AAA [24], el cual indica que en primer lugar, las precondiciones y datos de entrada deben prepararse al principio, luego operar sobre el componente bajo prueba y por último verificar que se cumplan los resultados esperados. Por otro lado, se busca mejorar el mantenimiento de las pruebas, permitiendo reducir el esfuerzo requerido para conservar el correcto funcionamiento de ellas. Por esta razón se presentan dos capas: capa de servicio y capa de pruebas.

Capa de Servicio

Contiene clases especiales denominadas servicios. Así como el patrón PageObject [25] permite separar la implementación de una página web de las pruebas, de la misma manera, la capa de servicios aísla de las pruebas las siguientes implementaciones:

- La especificación del cliente a utilizar
- La configuración de los parámetros para generar la petición
- La creación de la petición
- La obtención de la respuesta

Cada método en una clase de servicio, representará una operación disponible de ese servicio. De esta manera, para la capa de pruebas, así como en un navegador web, la interacción con el servidor es transparente. En la Fig. 4 se implementa un servicio que tiene una operación para obtener una lista de países según un continente.

```
1 public class CountryService {
2
3     private HttpClient client = new RestTemplateClient();
4
5     private String username = PropertyReader.getProperty("Username");
6     private String password = PropertyReader.getProperty("Password");
7
8     private String countriesURL = PropertyReader.getProperty("url.countries");
9
10
11     public HttpResponseMessage getCountries(String continent) {
12         client.setCredentialsForBasicAuthentication(username, password);
13         client.addURLParameter("continent", continent);
14         return client.get(countriesURL, Country[].class);
15     }
16 }
```

Fig. 4. Ejemplo de la Implementación de una clase de servicio.

Capa de Pruebas

En esta capa estarán las pruebas. Cada prueba deberá tener los siguientes pasos:

1. Creación del servicio que va a ser utilizado.
2. Invocación a la operación del servicio
3. Verificar si el servicio respondió correctamente, mediante su código de estado.

4. Realizar las verificaciones correspondientes a la prueba en sí.

Los pasos 1 y 2, consisten en definir e instanciar la clase del servicio a utilizar, e invocar al método que representa la operación que se desea. El resultado de esto será un objeto `HTTPResponse` que contiene los métodos para acceder a las variables para realizar las verificaciones de los pasos 3 y 4. Las verificaciones y la ejecución de las pruebas puede realizarse utilizando herramientas de ejecución de pruebas automáticas como JUnit [26] o TestNG [27]. Esto puede verse en la Fig. 5.

```
1  @Test(description="Esta prueba verifica que la lista de países no esté vacía")
2
3  public void countryTest(String continent) {
4      //Paso 1
5      CountryService service = new CountryService();
6      //Paso 2
7      HTTPResponse response = service.getCountries(continent);
8      //Paso 3
9      Assert.assertEquals(response.getStatusCode(), 200, "El servidor no respondió correctamente");
10     //Paso 4
11     Country[] countries = (Country[]) response.getBodyResponse();
12     int cant = countries.length;
13     for(int i=0, i<cant; i++) {
14         SoftAssert.assertNotNull(countries[i]);
15     }
16     SoftAssert.assertAll();
17 }
18
19
```

Fig. 5. Ejemplo de una prueba de un servicio web.

5 Resultados obtenidos

En las primeras etapas de pruebas sobre servicios web, esta actividad se realizó manualmente. La empresa constaba de un equipo que constaba de 15 personas realizando pruebas manuales utilizando clientes REST como Postman y Swagger, y el proceso tomaba un total de 3 horas. Por esta razón, se decidió invertir en la utilización de un cliente REST para realizar las pruebas. Esto redujo significativamente el tiempo para realizar las pruebas. Sin embargo, se perdía mucho tiempo en comprender la manera en manipular la herramienta y en aplicar cada actualización de la misma sobre cada una de las pruebas. El tiempo de creación de una prueba promedio era de entre 1 a 2 horas. Las refactorizaciones podrían tomar un 1 día entero de trabajo de una persona. Finalmente, se optó por una solución que consistió en la implementación del framework descrito en este trabajo.

En la Fig. 6 se pueden apreciar los resultados (expresados en minutos) de los 3 enfoques según 4 características:

- Tiempo para crear un caso de prueba
- Tiempo para ejecutar un lote de caso de pruebas
- Tiempo requerido para realizar cambios varios (refactorización)
- Tiempo requerido para generar la documentación

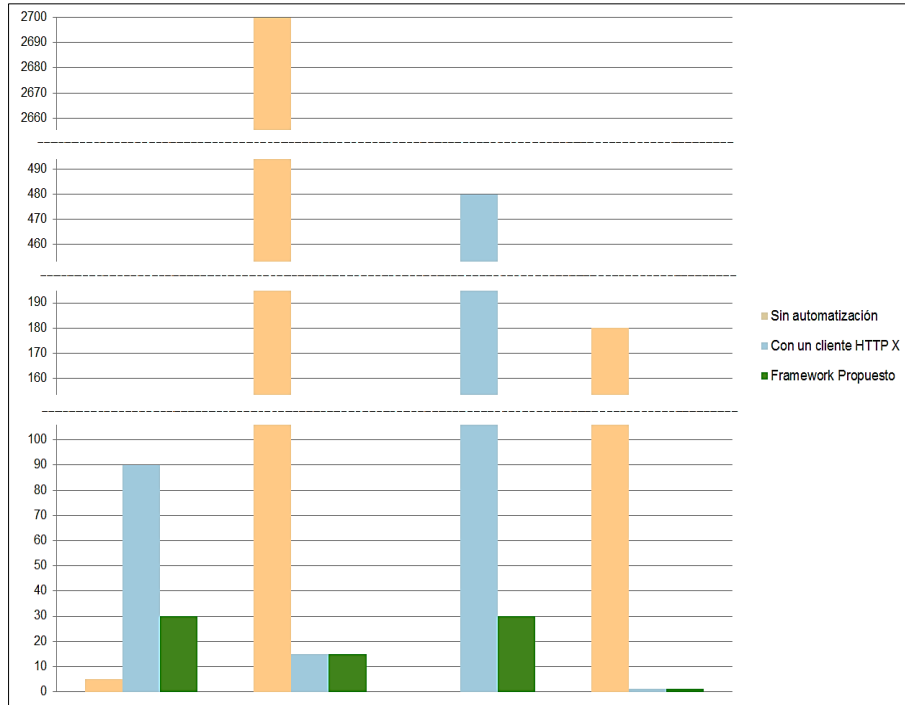


Fig. 6. Comparación de características asociadas a las pruebas según los diferentes enfoques.

El análisis de los resultados mostrados en la Fig. 6, puede verse en la Tabla 1. Se ha comprobado que tanto con el cliente como con el framework, se obtiene una disminución en un 90% del tiempo para realizar regresiones automatizadas con un servidor de integración continua, así como también la generación de reportes es automática. Con el Framework, el tiempo para crear un caso de prueba pasó a ser de 30 minutos. Además, se redujo la cantidad de refactorizaciones, con un tiempo promedio de 30 minutos.

Tabla 1. Comparación de características asociadas a las pruebas según los diferentes enfoques.

Característica	Sin automatización	Con un cliente HTTP "X"	Framework Propuesto
Creación de un caso de prueba	5 min	90 min	30 min
Ejecución de un lote de pruebas	2700 min	15 min	15 min
Refactorización	N/A	480 min	30 min
Documentación	180 min	1 min	1 min

6 Conclusión

Son muchas las empresas que están migrando a las tecnologías de servicios REST y las pruebas sobre ellos cobran relevancia. Si bien existen muchas herramientas, muy pocas pueden integrarse a un proceso de desarrollo continuo de software, donde el tiempo es un factor fundamental.

Mediante la integración de los principios de la arquitectura REST utilizados para el desarrollo de los mismos, y los fundamentos de pruebas unitarias, fue posible la creación de un framework que permite el desarrollo de pruebas automatizadas para servicios REST.

El tiempo de ejecución de un lote de pruebas que llevaba 3 horas, realizado manualmente con el equipo de 15 personas, pasó a ser de 15 minutos en un servidor de integración continua 100% automatizado. Este equipo ahora utiliza esas 3 horas para realizar otro tipo de pruebas complementarias y brindar soporte a otras áreas. El tiempo para crear un caso de prueba pasó a ser de 30 minutos. Además, se redujo la cantidad de refactorizaciones, con un tiempo promedio de 30 minutos.

La generación de reportes también es una tarea totalmente automatizada, a través de las herramientas de ejecución de pruebas automáticas.

Finalmente, los responsables del área de calidad manifiestan que con la implementación del framework, la confiabilidad en las pruebas aumentó considerablemente, ya que algunos errores podían pasar como desapercibidos con las pruebas manuales.

Referencias

1. R. Fielding, "Architectural styles and the design of network-based software architectures," University of California, Irvine, Tesis Doctoral 2000.
2. L. Deseta. (2008, Noviembre) Introducción a los servicios web RESTful. Dos Ideas (En Ideas Ágiles).
3. J. Sharma and M. Singh, "Web Services Oriented Architecture for DPI based Network Forensics Grid," International Journal of Energy, Information and Communications, vol. 6, no. 3, pp. 19-28, 2015.
4. L. Richardson and S. Ruby, RESTful Web Services. California, USA: O'Really Media, Inc., 2008.
5. L. Richardson and S. Ruby, "Writing Web Service Clients," in RESTful Web Services. California, USA: O'Really Media, Inc., 2008, ch. 2, p. 45.
6. Postman. [Online]. <https://www.getpostman.com/>
7. C. Zhou. RESTClient, a debugger for RESTful web services. [Online]. <https://addons.mozilla.org/es/firefox/addon/restclient/>
8. Swagger. APIS.Guru: A wikipedia for web APIs. [Online]. <http://swagger.io/>
9. World Wide Web Consortium (W3C). (2004, Febrero) Web Services. [Online]. <https://www.w3.org/TR/ws-arch/>
10. J. J. Domínguez Jiménez, A. Estero Botaro, I. Medina Bulo, M. Palomo Duarte, and F. Palomo Lozano, "El reto de los servicios Web para el software libre," in Proceedings of the FLOSS International Conference, 2007, pp. 117 - 132.
11. C. Morales Machuca, "Estado del arte: Servicios Web," Universidad Nacional de Colombia, Tesis de Maestría 2010.

- 12.R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T. Fielding, Hypertext transfer protocol--HTTP/1.1, 1999.
- 13.R. Johnson, "REST and Web Services: In Theory and in Practice," Research Gate, Diciembre 2010.
- 14.SoapUI. SMARTBEAR. [Online]. <https://www.soapui.org/>
- 15.Restfuse. [Online]. <https://github.com/eclipsesource/restfuse>
- 16.Rest-assured. [Online]. <https://github.com/rest-assured/rest-assured>
- 17.Pivotal. Spring. [Online]. <https://spring.io/>
- 18.P. Tahchiev, F. Leme, V. Massol, and G. Gregory, JUnit in action, 2nd ed., ACM Digital Library, Ed.: Manning Publications Co, 2010.
- 19.Spring. Spring MVC Test Framework. [Online]
<http://docs.spring.io/spring/docs/3.2.x/spring-framework-reference/htmlsingle/#spring-mvc-test-framework>
- 20.S. Bhandari. (2011, Enero) List of tools for Rest Web Services in Java. DZone. [Online].
<https://dzone.com/articles/list-tools-rest-web-services>
- 21.M. Fowler. (2010) Data Transfer Object. [Online].
<http://martinfowler.com/eaCatalog/dataTransferObject.html>
- 22.M. Gulden and S. Kugele, "A concept for generating simplified restful interfaces," in Proceedings of the 22nd international conference on World Wide Web companion, International World Wide Web Conferences Steering Committee., 2013, pp. 1391-1398.
- 23.D. Peng, L. D. Cao, and W. J. Xu, "Using JSON for data exchanging in web service applications," Journal of Computational Information Systems, vol. 7, no. 16, pp. 5883-5890, 2011.
- 24.Microsoft Developer Network. (2016) Unit Test Basics - AAA Pattern. [Online].
<https://msdn.microsoft.com/en-us/library/hh694602.aspx>
- 25.M. Fowler. (2013, Septiembre) PageObject. [Online].
<http://martinfowler.com/bliki/PageObject.html>
- 26.JUnit. (2002) JUnit. [Online]. <http://junit.org/junit4/>
- 27.TestNG. (2004) TestNG.org. [Online]. <http://testng.org>

Propuesta de Esquemas de Perfiles para la Recuperación de Datos Científicos para un Sistema de Recuperación de Información del Área de Ciencias de la Computación

Rey, M.¹, Kuna, H.¹, Martini, E.¹, Canteros, A.¹, Cantero, A.¹, Rambo, A.¹, Biale, C.¹

¹Departamento de Informática. Facultad de Ciencias Exactas, Químicas y Naturales.
Universidad Nacional de Misiones

hdkuna@unam.edu.ar; martinrey@fceqyn.unam.edu.ar

Resumen. Un Sistema de Recuperación de Información requiere de diversos componentes que definen su funcionalidad y comportamiento. En el caso de un meta-buscador para la recuperación de datos científicos del área de ciencias de la computación, un esquema que defina la forma en la que van a ser almacenados tales datos se considera como un elemento necesario para su evolución. Se han desarrollado perfiles unificados para el almacenamiento de los datos de las entidades involucradas en la gestión de datos científicos, generados a partir de la acción de publicar un documento científico. Tales perfiles, se consideran como el punto de partida para la generación de nuevos componentes para el meta-buscador que haciendo uso de información propia puedan producir información de relevancia para el usuario de la herramienta.

Palabras clave: datos científicos, meta-datos, meta-buscador, recuperación de información.

1 Introducción

Los datos científicos pueden describirse como aquellos que son generados a partir de la acción de publicar un documento científico, en esta acción la publicación queda ligada a su autor o autores, así como también al medio que haya sido seleccionado para su divulgación, entendiendo al mismo como la revista científica o congreso de una disciplina en particular, elemento que agrega un dato más al considerar el campo de estudio en el cual se publica el documento. Posteriormente, si el artículo es citado por otra publicación se generan datos de esta conexión, ligando documentos, autores y fuentes de publicación. Considerando la registración de estas acciones a lo largo del tiempo, se podría generar un historial de un autor, de sus publicaciones y de los campos de conocimiento en los que suele trabajar. Inclusive se podría estimar la relevancia de sus trabajos y también de revistas o eventos científicos, al obtener información histórica de artículos que hayan sido publicados en los mismos.

La recuperación de datos científicos como actividad dentro de la disciplina de recuperación de información ha cobrado mayor interés en el último tiempo [1, 2]. El impacto de internet y sus tecnologías relacionadas ha llevado a la generación de grandes conjuntos de datos derivados de las acciones antes mencionadas [3, 4]. Asimismo, estos datos propiciaron el desarrollo de herramientas para su gestión, mantenimiento,

publicación y procesamiento. Es así como diferentes editoriales y asociaciones con reconocimiento de parte de la comunidad científica han publicado repositorios de datos científicos a través de sitios web que constituyen herramientas de consulta para el usuario-investigador. Complementariamente, han surgido numerosas herramientas de este tipo con un mayor o menor número de características, con variaciones en el origen de los datos, su organización y el procesamiento que realizan sobre ellos [5, 6].

Es en este contexto, que se ha planteado como alternativa un meta-buscador [7] que opera sobre el área de ciencias de la computación, accede a diversas fuentes para recuperar datos científicos y los ordena utilizando un algoritmo que estima su impacto en la comunidad científica [8, 9]. En esta herramienta, la búsqueda de integración de nuevas funcionalidades y de mejora tanto de rendimiento como de eficiencia exige la definición de una estructura homogénea para el almacenamiento de los datos científicos recuperados. Además de la consideración de cuestiones ligadas a la tecnología a emplear, en un entorno operativo cada vez más relacionado con Big Data.

Es por esto que el problema a resolver en la presente línea de trabajo es la generación de tal estructura; ya que es el punto de partida necesario para el desarrollo de procesos que, haciendo uso de los datos, generen información relevante para la consulta de un usuario-investigador, siendo este un objetivo a resolver en un futuro cercano.

2 Antecedentes

2.1 Un meta-buscador para datos científicos de las ciencias de la computación

El Sistema de Recuperación de Información (SRI) antes mencionado constituye un software que debe evolucionar [10]. En este sentido, se considera que un paso siguiente en su desarrollo comprende al almacenamiento de los datos de las entidades que son recuperadas a partir de cada búsqueda. De esta manera permitirá generar soluciones complementarias que, sin perder de vista el aspecto de *meta*-buscador, puedan servir para implementar servicios que sean de utilidad para los usuarios, y así obtener resultados de mayor relevancia. Por ejemplo: recomendar artículos similares, recomendar expertos para el área temática de la consulta, expandir las consultas del usuario a partir de resultados previos, entre otros.

Por otra parte, contar con tales datos, también implica definir diversas cuestiones ligadas a la implementación como son: un esquema de almacenamiento, tecnología asociada que permita su carga y consulta con facilidad, velocidad y efectividad, y los cambios que sean requeridos en los procesos seguidos por el SRI para la correcta integración de estas operaciones de persistencia. Tales definiciones constituyen el punto de partida para el desarrollo de operaciones que hagan uso de los datos almacenados para generar información relativa a las consultas que sean ejecutadas por los usuarios a fin de mejorar su experiencia con el meta-buscador.

2.2 Actualidad en la gestión de grandes conjuntos de datos científicos

En la actualidad, como se ha mencionado, existen numerosas alternativas a la hora de buscar datos científicos.

Por un lado se encuentran las asociaciones de profesionales y editoriales o empresas relacionadas que publican sus librerías digitales, como son ACM, IEEE y la editorial Elsevier con su producto Scopus [4]. Asimismo, existen diversos repositorios que recuperan y compilan publicaciones de revistas y eventos científicos, por ejemplo: DOAJ y DBLP [6]. A ellos, se suman desarrollos de buscadores que brindan acceso a una inmensa cantidad de fuentes de publicación, artículos y datos relacionados a sus autores, no solo las versiones académicas de los productos tradicionales de Google o Microsoft [11], sino también desarrollos de equipos de investigación u instituciones educativas [5]. Además de otros repositorios o bases documentales que se podrían denominar “particulares” al ser propias de una Universidad, Organización o Centro de Investigación específico, tanto del ámbito nacional como internacional¹.

2.3 Formatos para registración de datos científicos

Los diferentes lineamientos o recomendaciones que imponen las fuentes de publicación a través de las convenciones de formato de artículos que utilizan, generan que una misma entidad, pueda ser registrada de diferentes maneras según el lugar en dónde se publique. Este es un problema conocido en el ambiente académico, y en especial en el de procesamiento de datos científico; ejemplos concretos del mismo son: ambigüedad en el nombre de autores y centros de investigación, confusión en la relación entre las siglas y los nombres de revistas y congresos, dificultades en el tratamiento de nombres con formatos no-occidentales, entre otros [12, 13].

Es por esto, que con el correr del tiempo las fuentes de publicación han optado por la generación y adopción de esquemas para la catalogación de su contenido, en algunos casos siguiendo un estándar del área de bibliotecología como son Dublin Core o MARC [14], y en otros desarrollando esquemas propios. Los buscadores, en general han utilizado esquemas específicos a fin de dar a conocer las mejores prácticas para que el contenido de los repositorios fuera indexado y mejor posicionado por los motores de búsqueda, tal es el caso de herramientas como Google Scholar, Microsoft Academic y CiteSeer [15]. En otras oportunidades, las herramientas utilizan bases de datos (BD) que se generan a partir del procesamiento de publicaciones de fuentes específicas, por lo que los meta-datos disponibles son aquellos impuestos por sus convenciones de publicación internas. Mientras que existen otros casos en los que la definición de los meta-datos a almacenar se definió en base a procesos parcial o totalmente automáticos para recuperación de datos desde otras fuentes, tal es el caso, por ejemplo de DBLP [6], AMiner [5] y otras soluciones similares. Existen otros conjuntos de herramientas que permiten la registración de datos de un autor y sus publicaciones, pero en ellas, cambia el actor principal de la registración, no siendo un tercero, sino que son los mismos autores o investigadores aquellos que construyen los registros. Entre este tipo de

¹ Por ejemplo los disponibles en: *repositoriosdigitales.mincyt.gob.ar* – Accedido: julio-2016.

soluciones, se destacan las que generan redes sociales de investigadores como ResearchGate² y aquellas utilizadas para crear catálogos de recursos³.

En este contexto, no se puede afirmar que exista un esquema consolidado a nivel global para el registro de meta-datos de entidades como son los artículos, autores, fuentes de publicación, áreas temáticas y centros de investigación, siendo que diferentes soluciones para el procesado de datos científicos operan bajo diferentes lineamientos que no llegan a constituir estándares para tales actividades. Por tal motivo, para una herramienta como el SRI [10], se hace necesaria la definición de estructuras necesarias para la transformación y persistencia de tales datos a fin de posibilitar el desarrollo de soluciones complementarias que puedan brindar información de mayor relevancia para un usuario-investigador, siendo éste el objetivo principal del presente trabajo.

3 Generación de perfiles para almacenar datos científicos

3.1 Diseño de los perfiles

A fin de generar la estructura para la gestión interna de los datos científicos se procedió a examinar un conjunto de fuentes como buscadores y BD⁴ de carácter científico, de cada una de ellas se obtuvo un listado de los meta-datos que utilizan para la registración de las diferentes entidades con las que operan. Sobre ese relevamiento inicial, se prosiguió con la revisión de los atributos utilizados por cada fuente, a fin de identificar qué compone para cada una de ellas el “perfil” de cada entidad, por ejemplo: cuáles referencian todas las publicaciones de un autor a través de su perfil, cuáles especifican el área temática a la que se corresponde una revista o evento científico, qué tipo de métricas mantienen sobre cada entidad, entre otros. De esta manera, se obtuvo con una descripción detallada de las características principales de cada fuente que permitió progresar con una instancia posterior de análisis.

A partir del análisis de los meta-datos de cada fuente se prosiguió con la extracción de aquellos atributos comunes y/o de similar significado, para la caracterización de cada entidad. Posteriormente, se seleccionaron aquellos atributos, que pudiendo no estar incluidos en la operación anterior, se consideraron de importancia para los objetivos del SRI, y por lo tanto, se incorporaron a los perfiles a generar. Una vez definidos los perfiles, comenzaron a establecerse las relaciones entre los mismos, a fin de representar las conexiones lógicas que existen entre las entidades a registrar.

Como resultado de estas acciones, se obtuvieron los perfiles de meta-datos (ver figura 1) con los cuales operaría el meta-buscador, de las siguientes entidades:

- Artículos
- Autores
- Fuentes de publicación

² Sitio web: *researchgate.net/* - Accedido: julio-2016.

³ Por ejemplo: Mendeley (*mendeley.com/*) – Zotero (*zotero.org/*) - Accedidos: julio-2016.

⁴ Por ejemplo: ACM Digital Library, IEEE Xplore Digital Library, Google Scholar, Microsoft Academic, DBLP, CiteseerX, DOAJ, entre otros. - Accedidos: julio-2016.

- Áreas temáticas
- Centros de investigación

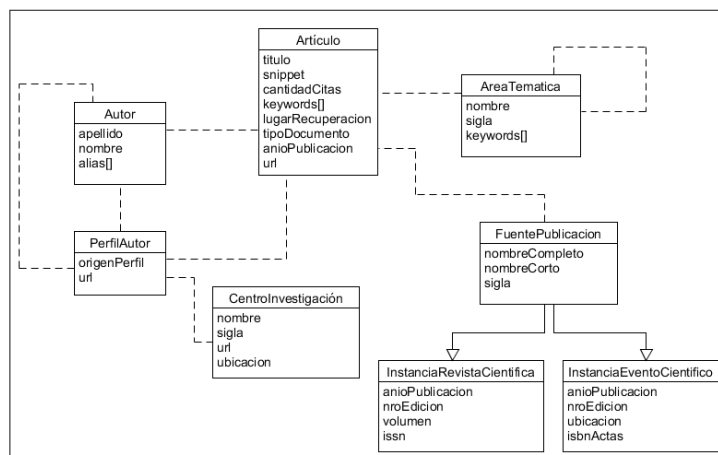


Fig. 1. Modelo conceptual de los perfiles generados para las entidades del SRI.

En las tablas 1, 2 y 3 se pueden apreciar la descripción de los atributos de las entidades principales del modelo: artículo, autor y perfil de autor y fuente de publicación con sus especializaciones para revistas y eventos científicos.

Tabla 1. Descripción de los atributos de la entidad Artículo

Atributo	Descripción
Título	Título del documento
Snippet	Descripción del documento
CantidadCitas	Número de citas según la fuente consultada
Keywords[]	Términos clave asociados
LugarRecuperación	Fuente desde donde se recuperó el artículo
TipoDocumento	Identificación si es artículo, libro o pre-print
AnioPublicacion	Año de publicación del documento

Tabla 2. Descripción de los atributos de las entidades Autor y Perfil Autor

Atributo	Descripción
Apellido	Apellido del autor
Nombre	Nombres del autor
Alias[]	Conjunto de nombres recuperados del autor
OrigenPerfil	Fuente desde donde se recuperó el perfil
URL	URL del perfil

Tabla 3. Descripción de los atributos de la entidad Fuente Publicación e Instancias

Atributo	Descripción
NombreCompleto	Nombre sin abreviaturas de la fuente
NombreCorto	Nombre con abreviaturas
Sigla	Sigla de la fuente
AnioPublicación	Año en el que se publicó la revista o realizó el evento
NroEdición	Número de la edición de la revista / evento
Volumen / Ubicación	Identificación del volumen de la revista / Lugar de realización del evento
ISSN / ISBN	Identificador de la publicación

3.2 Implementación de los perfiles

Una vez finalizado el diseño de los perfiles a emplear para la gestión de los metadatos de las entidades con las que opera el SRI se procedió con su implementación en el back-end del meta-buscador. Para ello se desarrollaron los objetos correspondientes a tales entidades, respetando los perfiles antes descritos e integrando a tales elementos dentro de la estructura del SRI, la cual se debió modificar para integrar nuevos módulos y generalizar el uso de los perfiles a través de todo el proceso de búsqueda y presentación de los resultados. De esta manera, la estructura del meta-buscador que fuera presentada previamente se amplió con los siguientes módulos:

- *Módulo para la gestión de la persistencia:* encargado de la configuración y ejecución de las operaciones cuyo objetivo es la interacción con la BD del SRI para almacenar y recuperar información de los perfiles.
- *Módulo para la recuperación de datos:* encargado del tratamiento y la actualización de los meta-datos de las diversas entidades que se almacenan en la BD del meta-buscador.

Las clases definidas para la representación de los perfiles de las entidades a gestionar se definieron en un ámbito general, con la finalidad de que todos los módulos de la herramienta pudieran hacer uso de los mismos en forma independiente al proceso que tengan por objetivo ejecutar. A nivel de tecnologías empleadas, el cambio con la última versión del SRI se presenta en la persistencia, migrando de un entorno relacional a uno

NoSQL⁵, específicamente uno documental basado en objetos JSON⁶ para el almacenamiento de las consultas ejecutadas y las entidades cuyos perfiles sean recuperados por los procesos del SRI. De esta manera, se integraron los componentes para la gestión de los perfiles al SRI, definiendo como siguiente paso la adaptación de sus procesos internos para hacer uso de los mismos a fin de brindar alguna solución complementaria para el usuario. La estructura del meta-buscador se puede observar en la figura 2.

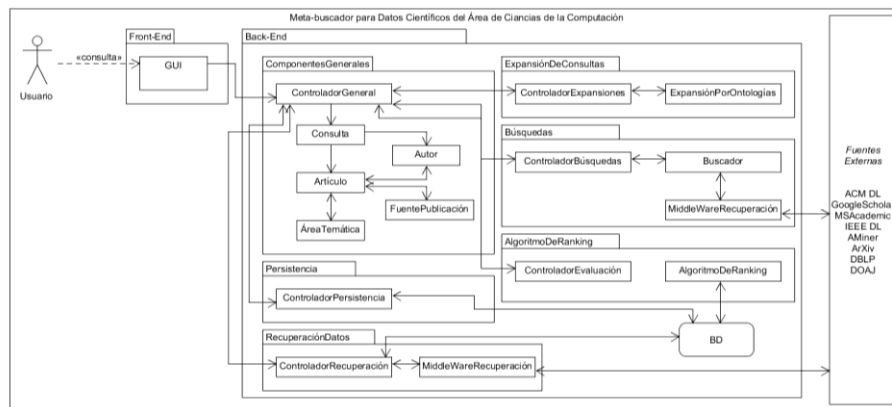


Fig. 2. Estructura del SRI

4 Implementación de los cambios en el SRI

4.1 Procesos del SRI modificados

Una vez integrados los perfiles definidos para la registración de los datos de las entidades con las que opera el meta-buscador, se debió proceder con la modificación de sus procesos operativos a fin de adaptar la solución desarrollada. Se inició con la modificación de los componentes del módulo de búsqueda de resultados, en los que se hace uso de las clases definidas por entidad para construir las colecciones de datos a procesar con cada búsqueda, almacenada en un objeto consulta en la BD. Por el momento, se mantuvo a las publicaciones científicas como único tipo de resultado a recuperar, sin embargo se enriquecieron los datos que se extraen de cada fuente con respecto a las relaciones que existen entre las entidades conforme a los perfiles del SRI.

A fin de optimizar los tiempos de respuesta al usuario, los meta-datos de los autores y las fuentes de publicación no son recuperados al mismo tiempo que se ejecutan las operaciones de búsqueda, sino que únicamente se recuperan los links hacia los perfiles que mantiene cada fuente consultada por el SRI, dejando rastro de las relaciones que existen entre tales elementos para la posterior recuperación de sus datos. De esta

⁵ Acrónimo para Not Only SQL.

⁶ Sigla para JavaScript Object Notation.

manera, la búsqueda se centra en recuperar los artículos necesarios para la aplicación del algoritmo de ranking y su presentación al usuario.

Para la ejecución de las operaciones posteriores a la búsqueda se han desarrollado procesos en los que no interviene el usuario y que tienen por objetivo la captura de los meta-datos de cada entidad que se haya generado en la ejecución del SRI, variando los detalles de transformaciones aplicables sobre los mismos a partir de las particularidades de cada fuente de datos que es consultada al momento de las búsquedas. De esta manera, se propuso un esquema a través del cual el SRI aumenta el contenido de su BD interna sin que esto signifique una merma en el rendimiento de las operaciones de búsqueda que son visibles al usuario-investigador.

4.2 Implementación de soluciones complementarias

Como resultado de las modificaciones introducidas, a partir de cada búsqueda la BD interna del SRI incorpora datos de los artículos que son recuperados, así como también de las entidades que guardan relación con los mismos. Por ejemplo: el perfil de un autor en particular según una o más fuentes, incluyendo sus artículos publicados, las áreas temáticas en las que ha trabajado, las instituciones a las que ha representado y los investigadores con quienes ha desarrollado relaciones de co-autoría. Al contar con este potencial volumen de datos se propuso el desarrollo de soluciones anexas que, integradas al SRI y haciendo uso de su BD, pudieran presentar información de valor agregado al usuario. Se plantearon tres alternativas, que se describen a continuación:

- *Método de recomendación de entidades basado en la relación que las mismas pudieran mantener con la consulta ingresada por el usuario.* Se propuso que a medida que el SRI ejecuta las búsquedas sobre fuentes externas, se presenten al usuario resultados recuperados desde la BD interna del meta-buscador, concretamente perfiles de autores, fuentes de publicación y artículos relevantes para la consulta ingresada por el usuario.
- *Método de filtrado de resultados basado en áreas temáticas.* En este caso, se propuso el desarrollo de un método que, considerando la consulta del usuario determine el área temática dentro de las ciencias de la computación con la que guarda mayor relación para poder aplicar filtros específicos al momento de que el SRI ejecute la recuperación de resultados desde fuentes externas.
- *Método alternativo de expansión de consultas basado en la interpretación de la consulta del usuario.* Se propuso que la consulta ingresada por el usuario pudiera ser expandida utilizando el “historial” de búsquedas del SRI.

5 Experimentación

5.1 Validación de los perfiles integrados al SRI

A fin de validar el funcionamiento del SRI al integrar los perfiles generados, se planteó un conjunto de pruebas que permitiera verificar el resultado de lo planteado en la sección 4.1. Para ello, se definió un conjunto de consultas a ejecutar en temáticas

variadas para evaluar que la captura, adaptación y persistencia de los datos fuera correcta. Con respecto al entorno tecnológico, los cambios con las últimas versiones del SRI se registraron en el backend, principalmente en la BD a emplear, migrando a un motor NoSQL de tipo documental, específicamente MongoDB en su versión 3.0.2.

Se realizaron 10 consultas, con un límite de 30 artículos a recuperar por cada una, ejecutando los procesos de recuperación de datos mencionados en la sección 4.1. Se evaluó la efectividad del SRI en tres aspectos: en el almacenamiento de los links hacia los perfiles, en la recuperación de los datos de la entidad y en el registro de las relaciones entre las entidades recuperadas a partir de un mismo documento. Los resultados obtenidos se pueden observar en la tabla 4.

Tabla 4. Resultados de la validación

Métrica	Valor
Cantidad de perfiles a generar	804
Efectividad de la persistencia	91% (736/804)
Efectividad de la recuperación	79% (581/736)
Efectividad de la generación de relaciones entre entidades	89% (568/640)

Al analizar los resultados obtenidos se puede observar que los métodos han funcionado en la mayoría de las ocasiones de manera correcta, con algunos errores en la recuperación de los perfiles, que deberán refinarse. Sin embargo, como el objetivo de la validación fue testear la funcionalidad e integración de los perfiles y procesos asociados, no se evaluaron a los mismos a partir de métricas propias de recuperación de información. Esta acción será ejecutada una vez implementados los procesos mencionados en la sección 4.2 cuando los datos de la BD del SRI sean explotados para la generación de información de ayuda para el usuario final.

6 Conclusiones y Líneas futuras de investigación

Se ha logrado especificar un conjunto de meta-datos para la registración de datos científicos. Los perfiles generados han sido producto de la unificación de los esquemas de almacenamiento planteados por diversas fuentes de datos, homogeneizando datos y relaciones a registrar. Estos perfiles se han implementado e integrado en un meta-buscador conjuntamente a los procesos necesarios para su generación, la recuperación de sus datos y establecimiento de las relaciones antes mencionadas.

A partir de la experimentación realizada se puede considerar que los mismos constituyen un punto de partida óptimo para el desarrollo de procesos destinados a generen información complementaria para optimizar los procesos del SRI o sean dirigidas al usuario para facilitar sus búsquedas. Tales soluciones son las principales líneas de trabajo a futuro a ejecutar, entre las que se destacan: un proceso de adaptación de consultas basado en algoritmos de extracción de tópicos y asociación por similitud y métodos para recomendación de resultados en forma interna basados en los datos presentes en la BD, entre otros. La implementación de este tipo de procesos se estima que implicará realizar adaptaciones sobre el meta-buscador, principalmente en un

aspecto técnico, requiriendo la utilización de estrategias propias del área de Big Data en cuanto al almacenamiento y acceso a datos.

7 Bibliografía

1. Bose, R., Frew, J.: Lineage retrieval for scientific data processing: a survey. *ACM Computing Surveys*. CSUR. 37, 1–28 (2005).
2. Simmhan, Y.L., Plale, B., Gannon, D.: A Survey of Data Provenance in e-Science. *SIGMOD Rec.* 34, 31–36 (2005).
3. Haustein, S., Peters, I., Bar-Ilan, J., Priem, J., Shema, H., Terliesner, J.: Coverage and adoption of altmetrics sources in the bibliometric community. *Scientometrics*. 101, 1145–1163 (2014).
4. Falagas, M.E., Pitsouni, E.I., Malietzis, G.A., Pappas, G.: Comparison of PubMed, Scopus, Web of Science, and Google Scholar: strengths and weaknesses. *FASEB J.* 22, 338–342 (2008).
5. Tang, J.: AMiner: Mining Deep Knowledge from Big Scholar Data. In: *Proceedings of the 25th International Conference Companion on World Wide Web*. pp. 373–373. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2016).
6. Ley, M.: The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives. In: Laender, A.H.F. and Oliveira, A.L. (eds.) *String Processing and Information Retrieval*. pp. 1–10. Springer Berlin Heidelberg (2002).
7. Kuna, H., Rey, M., Martini, E., Solonezen, L., Podkowa, L.: Desarrollo de un Sistema de Recuperación de Información para Publicaciones Científicas del Área de Ciencias de la Computación. *Rev. Latinoam. Ing. Softw.* 2, 107–114 (2013).
8. Kuna, H., Martini, E., Rey, M.: Evolution of a Ranking Algorithm for Scientific Documents in the Computer Science Area. In: *XX Argentine Congress of Computer Science Selected Papers*. pp. 145–155. EDULP, La Plata, Buenos Aires, Argentina (2015).
9. Rey, M., Kuna, H.D., Martini, E., Podkowa, L., Pautsch, J.G.A., Zamudio, E.: Generación de un método de expansión de consultas basado en ontologías para un sistema de recuperación de información. Presented at the XX Congreso Argentino de Ciencias de la Computación (Buenos Aires, 2014) (2014).
10. Kuna, H., Rey, M., Martini, E., Canteros, A., Cantero, A., Rambo, A., Biale, C., Zamudio, E.: Avances en la Construcción de un Sistema de Recuperación de Información para Información Científica en Ciencias de la Computación. Presented at the XVIII Workshop de Investigadores en Ciencias de la Computación April 14 (2016).
11. Ortega, J.L., Aguillo, I.F.: Microsoft academic search and Google scholar citations: Comparative analysis of author profiles. *J. Assoc. Inf. Sci. Technol.* 65, 1149–1156 (2014).
12. Ruiz-Pérez, R., López-Cózar, E.D., Jiménez-Contreras, E.: Spanish personal name variations in national and international biomedical databases: implications for information retrieval and bibliometric studies. *J. Med. Libr. Assoc.* 90, 411–430 (2002).
13. Garcíarena Ucelay, M.J., Villegas, M.P., Cagnina, L., Errecalde, M.L.: Cross domain author profiling task in spanish language: an experimental study. *J. Comput. Sci. Technol.* 15, no. 2, (2015).
14. Ortiz-Repiso Jiménez, V.: Nuevas perspectivas para la catalogación: Metadatos Versus Marc. *Rev. Esp. Doc. Científica*. 22, 198–219 (1999).
15. Beel, J., Gipp, B., Wilde, E.: Academic Search Engine Optimization (ASEO). *J. Sch. Publ.* 41, 176–190 (2009).

Modelado y ejecución de procesos de negocio con conceptos de Green IT

Anahí Rodríguez¹, Patricia Bazán¹, Viviana Ambrosi^{1,2}, Javier Díaz¹

¹ LINTI – Facultad de Informática – UNLP

² Comisión de Investigaciones Científicas (CIC-BA)

arodriguez@linti.unlp.edu.ar, pbaz@info.unlp.edu.ar, vambrosi@info.unlp.edu.ar,
javier.diaz@linti.unlp.edu.ar

Abstract. La importancia del cuidado del medio ambiente es cada vez más relevante para las organizaciones y gobiernos. La gestión por procesos de las mismas se ve involucrada en este desafío y surgen nuevos indicadores dentro del ciclo de mejora continua, no solamente para optimizar la productividad sino también para considerar la sostenibilidad de las actividades. Green BPM proporciona una forma de trabajar para minimizar los impactos en el medio ambiente. En esta investigación se propone la incorporación de indicadores “verdes” durante el modelado del proceso y también durante la ejecución, registrando sus rastros, de modo de obtener trazas de ejecución que permitan una monitorización específica y enriquecida, para lograr una mejora continua de procesos de negocio “verdes”.

Keywords: Desarrollo Sostenible – Green IT – Green BPM – Modelado de procesos de negocio

1 Introducción

En el marco de la “Conferencia de las Naciones Unidas sobre el Medio Ambiente y el Desarrollo” realizada en Río de Janeiro en 1992 [1], fue creado el *World Business Council for Sustainable Development* (WBCSD) [2] para involucrar a las empresas en cuestiones de sostenibilidad. El industrialista suizo Stephan Schmidheiny, ideólogo del consejo, tuvo la visión de reconocer el indeclinable rol de las empresas en la búsqueda del desarrollo sostenible. Sosteniendo que las empresas podían actuar como un catalizador para el cambio hacia la consecución del desarrollo sostenible y que, al mismo tiempo, necesitaban de él para alcanzar su mayor potencial.

Hoy en día las organizaciones y los gobiernos están más interesados y comprometidos con la ecología y el cuidado del ambiente. No sólo por los compromisos internacionales asumidos post COP 21 en París [3], sino también por las propias regulaciones existentes en algunos países.

Por otro lado en el documento “*Environmental Key Performance Indicators*” [4] se cita que las empresas que miden, gestionan y comunican su desempeño ambiental son

inherentemente mejor colocadas en el mercado. Ellas entienden cómo mejorar sus procesos, reducir sus costos, cumplir con los requisitos reglamentarios y las expectativas de los *stakeholders* y aprovechar las nuevas oportunidades de mercado.

Surgen así nuevos retos que llevan a redefinir las formas de trabajo para minimizar impactos en el ambiente, y donde las organizaciones y el sector de TI (Tecnología de la Información) juegan un papel muy importante para implementar y verificar métodos, políticas y procedimientos que ayuden a la conservación del medio ambiente y que además realicen sus tareas de manera responsable y sostenible teniendo en cuenta las TIC verdes [5].

Existen muchas definiciones para Green IT, como por ejemplo Jonh Lamb en [6] define Green IT como *"El estudio y la práctica de la utilización de los recursos informáticos de manera eficiente, pero esto no lo hace sostenibles"*. Mientras que Weber y Wallace en [7] la definen como *"La reducción del impacto ambiental del funcionamiento del departamento de TI"*, pero es una visión reducida del problema. En cambio J. Díaz et al [8] lo definen como *"uso eficiente de los recursos computacionales minimizando el impacto ambiental, maximizando su viabilidad económica y propiciando una mejora del contexto social"*. Pero podríamos dar una definición más ampliada indicando que Green IT es *"uso eficiente de los recursos computacionales tanto desde el punto de vista del hardware como de software, minimizando el impacto ambiental, maximizando su viabilidad económica y propiciando una mejora del contexto social y cultural"*.

De estas definiciones se desprende que Green IT trata de considerar la protección del medio ambiente durante todo el ciclo de vida de TI. Existen algunas prácticas como la eficiencia en los servidores de datos o la gestión de residuos electrónicos, siendo éstas sólo algunas de las posibles acciones de Green IT [10]. Dichas prácticas se pueden evaluar con métricas para hardware y software que determinen el impacto negativo y permitan realizar acciones para revertir o mejorar las actividades o procesos en las organizaciones [11] [5].

Los procesos de negocio no están ajenos a esta realidad, por lo que las organizaciones deben trabajar teniendo en cuenta esta problemática inminente y los requerimientos asumidos por los gobiernos y empresas a nivel internacional.

Las organizaciones que adhieren a un paradigma de gestión por procesos (BPM – *Business Process Management*), se introducen en un ciclo para identificar, modelar, desplegar y monitorizar procesos de negocio en búsqueda de la mejora continua. Este ciclo se cierra con la monitorización de indicadores que dan cuenta del rendimiento de los procesos y que suelen definirse de manera clásica y universal, como por ejemplo, tiempo de ejecución entre ejecución de actividades de un proceso, cantidad de tareas detenidas, en progreso o fuera de término. En este sentido, sería razonable pensar que estas mismas organizaciones comprometidas con el medio ambiente, estarían interesadas en medir cuan sostenibles son sus procesos, por ejemplo, en función de los recursos que consumen, teniendo en cuenta que en muchos casos son recursos naturales no renovables.

Surge así el concepto de proceso "verde" que requiere que se definan indicadores de mejoras vinculada a aspectos ecológicos y además demanda mejorar las notaciones de modelado y obtener la trazabilidad necesaria para monitorizar los procesos según los indicadores definidos.

Este nuevo concepto de proceso de negocio, amerita un replanteo acerca de la forma en que se modelan, ejecutan y monitorizan. Los analistas de procesos, como los participantes y los expertos en TI, deben considerar la forma de cómo incorporar estos indicadores “verdes”, tanto en tiempo de diseño y en la ejecución del proceso y la consecuente definición de nuevas trazas y criterios de monitorización.

Este trabajo propone un mecanismo para incorporar estos indicadores en tiempo de diseño, y también en ejecución, y define cómo deben ser tenidos en cuenta para monitoreo. El mecanismo se aplica a un caso de estudio desplegado en un motor de procesos (BPMS – *Business Process Management System*) concreto, como lo es *Bonita BPM* [19].

El artículo se organiza de la siguiente manera: en la Sección 2 se muestra el estado del arte y antecedentes acerca de la aplicación de Green IT a BPM. En la Sección 3 se analizan los requisitos que se consideran necesarios para el modelado y ejecución de procesos verdes, con la incorporación de los indicadores “verdes” propuestos. En la Sección 4 se aplica a un caso de estudio. Finalmente se arriban a algunas conclusiones y trabajos futuros.

2 Antecedentes en Green IT aplicado a BPM

Green BPM es la suma de todas las actividades de gestión soportadas por los sistemas de información que ayudan a monitorizar y reducir el impacto ambiental de los procesos de negocios en sus etapas de diseño, mejora, implementación u operación, así como guiar hacia un cambio cultural dentro del ciclo de vida del proceso [12], y ese encuentra en las primeras fases de investigación. La bibliografía es muy reciente en cuanto a cómo medir y monitorizar un Green BPM [13].

Para abordar un Green BPM podemos tener en cuenta los siguientes aspectos [12]:

- Actitud Verde: describe la actitud de la empresa y los empleados para obtener resultados sostenibles, viendo la responsabilidad y cuan interesados están en la adopción de medidas de mejora para lograr un proceso más sostenible. Este es el factor fundamental para tener un Green BPM a largo plazo. Tomando conciencia del impacto ecológico negativo del proceso de negocio haciéndolo más sostenible para tener un impacto menos negativo.
- Estrategia Verde: la gestión del Green BPM debe estar relacionada con la estrategia comercial para el ciclo de vida del BPM, para poder establecer mejoras ecológicas. Se planifica la estrategia y así genera un marco de actuación sostenible. Esto es importante para la adquisición de una estrategia holística para todo el ciclo de vida.
- Gobernabilidad Verde: esta etapa junto con la estratégica son parte de las tareas de gestión. Define la estructura de la gestión, los roles, autoridad para toma de decisiones, responsabilidades, etc. y el control de acciones verdes. También se organizan las personas involucradas en el proceso. La asignación de presupuestos y otros recursos son importantes para Green BPM. Por otra parte, la gobernabilidad verde es responsable de la definición de las metas y de medir el progreso hacia ellos.
- Modelado verde: se considera el impacto ambiental en el modelo del proceso. Se establecen valores y factores claves para cada unidad de la empresa. Con

estos valores podemos medir valores claves ecológicos en los procesos de negocio, sub-procesos, actividades, recursos. Pudiendo asignar valores a cada proceso de los recursos utilizados, la producción de residuos, o emisión de CO².

- **Monitoreo verde:** es de gran utilidad tener medidas para monitorear el impacto ambiental. El monitoreo verde utiliza indicadores ecológicos clave de rendimiento (KPI - *Key Performance Indicators*) para cada actividad de los procesos de negocio. Con estos valores se pueden medir sus impactos ambientales. Los KPI pueden ser de ayuda a una organización para cuantificar las TIC y evaluar el desempeño en esta área. Según la *International Telecommunication Union* (ITU) el tratamiento y gestión de los indicadores clave de rendimiento medioambiental no es nuevo. Algunas organizaciones ya han identificado una serie de KPI ambientales comunes entre ellos. En el documento “General specifications and KPIs” [17] la ITU ayuda a identificar qué indicadores clave de rendimiento podrían ser relevantes para una organización. Hay cinco grupos principales de KPI ambientales que normalmente se clasifican por instituciones del sector TIC y no TIC: la energía, las emisiones de gases de efecto invernadero, residuos, agua, y otros. Los indicadores claves ambientales (*Environmental KPI*) son de tipo directo e indirecto [4] y también encontramos de tipo estático y dinámico. El dinámico es el que se genera del uso real de un dispositivo, por ejemplo el consumo de energía de una impresora; mientras que los estáticos son independientes del uso, por ejemplo servidores, en el cual su uso es constante. Para el caso de los KPI dinámicos se utilizan redes de sensores para medir automáticamente. El monitoreo verde mide los niveles de KPIs ambientales de la organización, teniendo en cuenta valores internos y externos.

- **Optimización verde:** luego de obtener las mediciones anteriormente citadas se deberá realizar un análisis para evaluar posibles cambios en el proceso de negocio [16], si es que es necesario, optimizando con estas mejoras el impacto climático. Como primera medida no es necesario el cambio de todo el proceso, sino pueden ser implementados por etapas, como por ejemplo decidir utilizar más el formato digital que los documentos impresos.

3 Modelado y Ejecución de Procesos Verdes

Según el análisis del estado del arte en torno a Green BPM se desprende que existen definiciones precisas en torno a este concepto que requieren de un marco tecnológico que las implemente.

Actualmente, las organizaciones han alcanzado cierto grado de madurez en cuanto a la adopción del paradigma BPM, incluso utilizando herramientas específicas para su despliegue (BPMS). En este sentido, la ejecución de procesos de negocio es un desafío ya resuelto y hay una gran variedad de alternativas tal como se analiza en [18].

El modelado y ejecución de procesos “verdes” puede llevarse a cabo utilizando las soluciones tecnológicas existentes pero teniendo en cuenta algunas características “verdes”. Si bien se puede implementar en cualquier BPMS, proponemos desplegarlo en *Bonita BPM* [19], por sus características *open source* y por la gran comunidad que posee en torno a extensiones funcionales de su versión comunitaria.

3.1 Antecedentes para modelar y ejecutar procesos verdes

Las extensiones a la notación BPM y la definición de características funcionales para ejecución y monitoreo de procesos dinámicos son dos antecedentes importantes para encaminarse hacia Green BPM y que se presentan en [14] y [15].

En [14] se propone extender el modelado clásico de BPM agregando notaciones que indican aspectos ecológicos, como por ejemplo, añadiendo una notación para las actividades que hacen uso de combustible, actividades que utilizan papel, indicadores para mostrar la cantidad de gases que se emiten en cada *pool* o *lane*. Estos indicadores de aspectos ecológicos podrían dar un gran valor agregado a los resultados esperados del proceso de negocio y el monitoreo del mismo tendría que tener en cuenta estos rastros de ejecución.

Para aportar dinamismo a la ejecución del proceso de negocio en [15] se especificaron características funcionales que debe tener un motor de ejecución de procesos para incorporar la interacción entre participantes y enriquecer los rastros de ejecución de sus actividades, como por ejemplo, incorporación de notificaciones, comentarios etiquetados, definición de tareas ad-hoc sin modelar y notificaciones a los responsables de los procesos.

El dinamismo brindado por la mejora propuesta en [15] puede ser utilizado para incluir en "tiempo real" los valores de diversos aspectos ecológicos (por ejemplo: emisión de gases, consumo de electricidad, de papel, etc.). Con este enfoque los rastros de ejecución ya no serán los clásicos sino que son propios de la ejecución del proceso. Los datos asociados a cada actividad pueden obtenerse en tiempo real o no, dependiendo de la naturaleza de la actividad. Esto nos lleva a cambiar el enfoque de monitorización estática a una dinámica.

A través de la monitorización dinámica se pueden emitir alertas que sirvan para analizar el uso de las mejoras propuestas en los patrones definidos en [16] para obtener un Green BPM partiendo de un BPM clásico. Por ejemplo, aunque en el modelado del proceso inicial no se haya tenido en cuenta el indicador de consumo de energía, si se detecta dinámicamente que un servidor de la organización tiene un consumo de energía excesivo, se puede aplicar el "Patrón 3: Cambio de Recursos" el cual propone reemplazar el servidor propio por un servidor en la nube, o un patrón más adecuado en la planificación de la adquisición y en la generación de residuos electrónicos que pueda ser provocado por su descarte.

3.2 Incorporación de indicadores "verdes" y su despliegue en Bonita BPM

En función de lo presentado en la sección anterior, nuestra propuesta se sustenta en una adaptación de la notación planteada en [14] que se basa en la definición de nuevos tipos de actividades que involucren el uso de recursos cuantificados según criterios "verdes". Estos nuevos indicadores se definen para los casos que se cuente con valores que lo midan y, para cada uno de ellos, existe un flujo que lleve los valores de los indicadores a las actividades de cierto tipo.

Además, se requiere también incluir el mecanismo para obtener el dinamismo, tanto de indicadores no previstos en tiempo de modelado. En este caso el indicador

puede tomarse en tiempo de ejecución mediante una lectura por interacción humana o por ingreso desde otra fuente (como por ejemplo un sensor).

El método propuesto alcanza dos aspectos: 1- incorporación de nuevos indicadores para las actividades en *Bonita BPM* y la definición de cómo medirlos y 2- definición de un mecanismo para incorporar, en ejecución, estos nuevos indicadores mediante actividades automáticas.

3.2.1 Indicadores Verdes

El método propuesto contempla definir nuevos indicadores cuyos valores se almacenan en variables de procesos y pueden ingresarse manualmente a través de un input perteneciente a un formulario, o por la lectura de un sensor, el cual jugaría el papel de entrada de datos a un conector externo de *Bonita BPM*.

La sintáxis que se define para tales indicadores es *_NombreIndicador* y donde la semántica determinará la forma de calcular la medición del indicador.

Consideremos la definición de un indicador que mida el consumo de papel. Para ello utilizamos dos variables de proceso: 1 - *_ConsumoPapelP*, contabiliza el consumo de papel por cada instancia de proceso y 2 - *_ConsumoPapelT*, contabiliza el consumo de papel por cada actividad del proceso.

Las estadísticas que pueden obtenerse al incorporar estos indicadores como variables de proceso son, por ejemplo: 1- cantidad total de papel utilizada en todas las instancias del proceso ejecutadas, 2- cantidad total de papel utilizada por una actividad en particular o 3 - promedio de papel utilizado en una instancia del proceso.

El cálculo de cada indicador se realiza con consultas sobre parte de las tablas de catálogo de *Bonita BPM*, cuyo esquema de relaciones se muestra en la Figura 1.

La tabla *Process_instance* almacena los datos referentes al proceso en las distintas instancias ejecutadas, mientras que *Data_instance* almacena las variables de un proceso en las distintas instancias ejecutadas. Finalmente, *Flownode_instance* almacena los componentes del proceso en las distintas instancias ejecutadas.

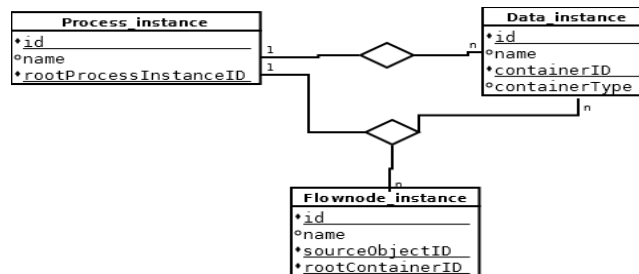


Figura 1. Esquematización de las tablas utilizadas.

La definición de estos nuevos indicadores en *Bonita BPM* definen nuevas estadísticas que se suman a las clásicas de todo BPMS, como lo son: el tiempo de ejecución entre actividades, la cantidad de actividades por participante, el tiempo total de ejecución de un proceso, las demoras en las actividades de un proceso, entre otras. A modo de ejemplo, se muestra la consulta para calcular la cantidad de papel utilizada en todas las instancias de proceso se define la siguiente consulta.

```

Select Sum (data_instance.intvalue)
from data_instance, process_instance
where data_instance.name = _consumoPapel and
(data_instance.containerID =
process_instance.rootProcessInstanceID)

```

3.2.2 Actividades automáticas

Los indicadores “verdes” antes mencionados, y representados como variables de proceso, pueden definirse tanto en tiempo de diseño como en tiempo de ejecución. En el primer caso, se modela con los criterios de modelado clásicos de BPM y solo se debe considerar nombrarlos con la nomenclatura fijada en el punto anterior.

Para incorporar los indicadores “verdes” en tiempo de ejecución, se define en tiempo de diseño una actividad automática (definida como tal por *Bonita BPM*) por cada actividad del proceso; esta actividad ejecutará un *web service*. El mismo contendrá el código necesario para generar un formulario dinámico que permita ingresar el indicador verde. A su vez, la actividad modelada debe incluir un chequeo que determine la ejecución o no de la actividad automática.

Dentro del modelo de proceso se incluirá un *lane* que lo llamaremos ROBOT el cual contendrá la actividad automática.

4 Caso de Estudio

El caso de estudio utilizado presentado en este trabajo, es el seguimiento y control de las tesinas de grado de las carreras de Licenciatura en Informática y Licenciatura en Sistemas de la Facultad de Informática de la Universidad Nacional de La Plata. La presentación de la propuesta de tesina la realizan 1 o 2 alumnos que cuentan con el aval del o de los director(es). Ante la presentación de la propuesta de tesina, se procede a validar los datos académicos. Luego se establece un Tribunal o Jurado, el cual es notificado, para que revise la propuesta. Cada profesor notificado puede aceptar o rechazar la propuesta. Finalizado el trabajo de tesina, los alumnos presentan el informe final. Acordada la conformación del tribunal, el mismo evaluará el informe y determinará una fecha de exposición o bien formulará pedidos de revisión. En ambos casos se notifica al/los alumno/s y al/los director/es. Luego de acordada la fecha de exposición, se lleva a cabo la misma, el Jurado emite su opinión, redacta un acta y fija la calificación. El proceso finaliza con la graduación del alumno.

Este proceso consta de varios subprocesos, uno de los cuales, “Validación de Datos Académicos y Alta de la Propuesta” y cuyo modelo se visualiza en la Figura 2, será el elegido para modelar y ejecutar incorporando indicadores “verdes” con el método propuesto en la Sección 3.

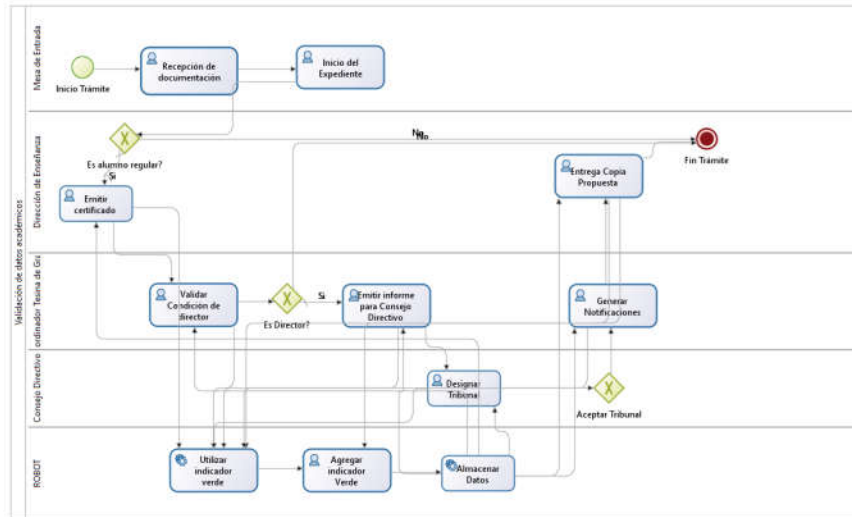


Figura 2. Validación de Datos Académicos y Alta de la Propuesta

En la Figura 2 se muestra el flujo de trabajo desde que se inicia el expediente con la documentación necesaria para el trámite, la designación del tribunal y generación de notificaciones. Según el método presentado se incorpora un *lane* ROBOT que incluye las actividades: 1- **Utilizar Indicador Verde** (Conector al *Web Service* que genera un formulario dinámico para el ingreso del indicador), 2 - **Agregar Indicador Verde** (Despliega el formulario generado por el *Web Service*) y 3 – **Almacenar Datos** (Almacena en las tablas respectivas los valores ingresados en la tarea anterior)

Como mencionamos anteriormente todas las actividades deben tener un chequeo que determine la ejecución o no de la actividad automática. Este chequeo lo determinará el participante del proceso, si desea contabilizar la cantidad de papel consumido por la actividad. Como se muestra en la Figura 3:

Emitir Informe Para Consejo Directivo

Escribir informe del Consejo Adjuntar Informe Ningún archivo seleccionado

Consumo Papel

Figura 3. Chequeo para la ejecución de la actividad automática.

El chequeo sobre el campo del formulario “Consumo Papel”, activa la ejecución del *lane* ROBOT que ejecuta el *Web Service* donde se genera el formulario dinámico

que permite la lectura del indicador. La Figura 4 muestra la ejecución de la actividad “Agregar indicador Verde”, donde se ingresa el valor en este caso de la cantidad de papel consumida. La tarea que se conecta a la base de datos se encarga de almacenar los datos ingresados, para las consultas que se quieran realizar posteriormente.

Agregar Indicador Verde

Ingresar Consumo de Papel

ENVIAR

Figura 4. Ingreso del indicador seleccionado

En este ejemplo se muestra un caso de estudio desarrollado usando el método definido en la Sección 3, que incluye: 1- Definición de indicadores dinámicos utilizando variables de procesos, 2- Incorporación de actividades automáticas para la lectura de los indicadores, 3 – Definición de nuevas estadísticas del proceso mediante el cálculo de nuevas trazas de ejecución generadas por los indicadores “verdes”.

5 Conclusiones y Trabajos Futuros

El trabajo presentado propone un método para modelado y ejecución de procesos de negocio bajo la óptica de Green BPM: conjunto de sistemas de información que ayudan a monitorizar y reducir el impacto ambiental de los procesos de negocios en sus etapas de diseño, mejora, implementación u operación.

Para aplicar el método se seleccionó la herramienta *Bonita BPM* por sus características *open source* y por la gran comunidad que posee en torno a extensiones funcionales de su versión comunitaria. Utilizando esta herramienta, el método permite diseñar, ejecutar y monitorizar un proceso de negocio incorporando datos verdes para su posterior análisis y revisión de estadísticas.

Los reportes que se pueden obtener de las nuevas trazas de ejecución incorporando aspectos “verdes” pueden ayudar a detectar posibles impactos negativos al medio ambiente luego de la ejecución del proceso y así poder lograr una mejora continua de procesos de negocio “verdes”.

Los trabajos futuros de esta investigación consiste en aplicar este mecanismo sobre otros BMPS que admitan la invocación a *Web Services* para la generación de nuevos indicadores y registren en sus tablas de catálogo la información necesaria para obtener trazas enriquecidas y nuevas estadísticas.

Referencias

1. La Conferencia de las Naciones Unidas sobre el Medio Ambiente y el Desarrollo, Declaración de Río sobre el Medio Ambiente y el Desarrollo, Junio de 1992, <http://www.un.org/spanish/esa/sustdev/agenda21/riodeclaration.htm>
2. World Business Council for Sustainable Development (WBCSD) <http://www.wbcsd.org/>
3. Conferencia sobre el Cambio Climático de París se conoce oficialmente como la 21a Conferencia de las Partes (COP) de la Convención Marco de Naciones Unidas sobre Cambio Climático (CMNUCC) <http://www.un.org/sustainabledevelopment/es/Cop21/>
4. Environmental Key Performance Indicators. Reporting Guidelines for UK Business. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/69281/pb11321-envkpi-guidelines-060121.pdf
5. Javier Muñoz Giner, Yuresky Rojas Rincón, “Nuevas tendencias en tecnologías verdes -Green IT para la Gestión en Organizaciones”, II Congreso Iberoamericano SOCOTE - Soporte al Conocimiento con la Tecnología- y VII Congreso SOCOTE Universidad Politécnica de Valencia, Noviembre 2010
6. Jonh Lamb, “The greening of IT: how companies can make a difference for the environment”, Capítulo 1. IBM Press – 0137150830/9780137150830 – Mayo 2009
7. Lawrence Webber, Michael Wallace, “Green Tech: How to Plan and Implement Sustainable IT Solutions”, Julio 2009.
8. Javier Díaz, Viviana Ambrosi, Néstor Castro, Claudia Banchoff Tzancoff, Marcelo Raimundo, “Porque incluir Green IT en la currícula de Informática”
9. Fatima Zahra HANNE, “GREEN-IT: Why Developing Countries Should Care?”, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, Julio 2011.
10. <http://green-software-engineering.de/en/publications.html>
11. Nicky Opitz, Henning Krüp, Lutz M. Kolbe, “Green Business Process Management – A Definition and Research Framework”, 47th Hawaii International Conference on System Science, 2014
12. Nicky Opitz, Henning Krüp, Lutz Maria Kolbe, “Environmentally sustainable business process management – developing a green bpm readiness model”.
13. Maximilian Röglinger, Jens Pöppelbuß, Jörg Becker, “Maturity Models in Business Process Management”, “Business Process Management Journal 18” (2012).
14. Jan Recker, “Green, Greener, BPM?”, <http://www.bptrends.com/publicationfiles/07-05-2011-COL-Class%20Notes--Green%20Greener%20BPM-Recker.pdf>
15. Patricia Bazán, Roxana Giandini, Jose Martinez Garro, Javier Díaz, “Mejora de la monitorización y ejecución de procesos de negocio con integración y socialización”, Latin American Computing Conference (CLEI), 2015
16. Alexander Nowak, Frank Leymann, Daniel Schleicher, David Schumm, Sebastian Wagner, “Green Business Process Patterns”, Proceedings of the 18th Conference on Pattern Languages of Programs, 2011
17. ITU General specifications and KPIs https://www.itu.int/dms_pub/itu-t/oth/4B/04/T4B0400000B0009PDFE.pdf
18. Delgado, A., Calegari, D., Milanese, P., Falcon, R., & García, E. (2015, May). A systematic approach for evaluating BPM systems: case studies on open source and proprietary tools. In IFIP International Conference on Open Source Systems (pp. 81-90). Springer International Publishing.
19. <http://es.bonitasoft.com/>

Quality Evaluation in Agile Process: A First Approach

Noelia Pinto¹, César J. Acuña¹, Liliana Cuenca Pletsch¹

¹Grupo de Investigación en Ingeniería y Calidad del Software (GICS), Facultad Regional Resistencia, Universidad Tecnológica Nacional, French 414, Resistencia, Chaco, Argentina
{ns.pinto, csr.acn, lilianacp}@gmail.com

Abstract. In recent years, it has been given much importance to the use of models and quality standards on software development processes. Because these are those that facilitate continuous improvement and enable companies to provide higher quality products to its customers by increasing their competitive level.

Today, software development is based on agile processes that allow production characterized by its changing requirements and the need for continuous customer deliveries environments. Thus it is imperative to provide companies with tools for assessing the quality of these cycles agile processes.

QuAM is presented in this article, an approach to design a model of quality, integrated and flexible, that assesses the quality development cycles based on the principles and practices of the agile approach.

Keywords: Agility, Quality Software, Software Engineering, Agile methodologies.

1 Introduction

The final product quality, low cost and timely deliveries become key elements for the benefit of domestic sales and international projection of the Software Industry. In this sense, and in order to increase the quality and capability of their processes and, consequently, the quality of its products and services, software process improvement [1] becomes the differentiating element that companies in the sector need to increase their competitiveness.

In the case of Argentina, Software Industry consists mainly of PYMES, companies that represent 80% of the sector, according to the latest report of the Permanent Observatory of Software Industry and Information Services (OPSSI) [2]. Thus, taking into account this reality, it is important to note that several authors [3] [4] [5] agree on the difficulty that means for PYMES to implement programs of Software Process Improvement (SPI) mainly due to the lack of monitoring action plans and implementation due

to the high cost involved. Thus, the parameters of development time and cost of solutions will directly affect the work done, resulting the quality as the first variable adjustment available.

However it is not correct to consider the study or the impact of those elements that associate the quality of the final product but is also necessary to adjust the parameters associated with the processes that have facilitated obtaining it. Depending on this, there are numerous methodological proposals that guide the software development cycle and affecting different dimensions of the process. The more traditional methodologies are especially focused on a rigorous definition of roles, the activities involved, artifacts to be produced, and the tools and notations that will be used [6].

But these approaches are not the most suitable for many projects related to current scenarios where the system environment is changing and where it is demanded to drastically reduce development time without neglecting high levels of quality. So agile methodologies pursue principles such as incremental delivery of new functionality to the client, which are prioritized according to business value added (so the software product evolves in different deliveries), favoring the continuous improvement and emphasis on close collaboration between the development team and business experts [7].

Previously it has been presented [8] a study on PYMES of Software companies in the NEA, in which the adoption of agile methodologies such as life cycle in their development processes was analyzed. From there the objective of this work is clear, that is to present QuAM, one first approach to the design of a model that allows quality assessment of agile processes, contemplating two perspectives: Process and Product. The article is structured as follows: in section 2 the state of the art is described through the presentation of related work. Then, in section 3, the characteristics of the proposed model by the design quality of experience for the same validation are presented. And finally, conclusions and future work are to be developed from this line of work are exposed.

2 Related Work

There are several models in the literature to assess the quality of software, quality trying break into a category of simpler characteristics and from two perspectives: the product and the process.

Among quality models that enable evaluation of the software product, the Model Mc Call, created by Jim Mc Call in 1977 [9] stands out. This defines 3 perspectives (Operability Product, Product Review and Transition Product) for the analysis of the quality of software, together with associated factors and criteria. The proposed metrics are questions that apply a numerical weighting to a particular software product attribute. After obtaining the values for all metrics specific criteria, the average of these is the value for that criterion. Another model worth mentioning is FURPS [10], developed by Hewlett-Packard in 1987, in which a set of quality factors of software are described: Functionality (Functionality) Usability (Usability), Reliability (Reliability), Performance (Performance) and capacity support (Supportability). These elements can be used to establish quality metrics for all software process activities.

Among the international quality standards associated with the most relevant software is ISO / IEC 9126 [11], based on a hierarchical model with three levels: Features, Sub-characteristics and Metrics. The first level has six characteristics: Functionality, Reliability, Efficiency, Maintainability, Portability and Ease of Use. These characteristics (factors) are composed in turn by sub-characteristics (sub- factors) related to external quality, and sub-characteristics related to the internal quality.

There are also quality models that evaluate processes for obtaining software product. One of them, based on agile methodologies, is the AGIS (combination of AGILE and ISO) [12] which establishes a mechanism for measuring the degree of agility of software development processes. The ISO model supplemented with 10 dimensions; this configuration is oriented to measure the degree of implementation of the values of the agile manifesto [13] in the areas of engineering knowledge. AGIS aims to meet two needs: one focuses on companies, since this model can achieve differentiation from other companies that have only certified quality through ISO 9001: 2008. Furthermore, AGIS provides a report with suggestions for improvement based on the assessment of the size proposed evaluate.

Another model, similar to the above, is the AGIT (Agile Software Development) [14] which suggests that the best performance is achieved when the goals of all stakeholders are met. This requires an approach considering the views of different stakeholders, for which appropriate indicators are defined for each. AGIT considers four different views for stakeholders: the IT Manager is the actor concerned with traditional aspects of performance considering SW development time, cost and quality; the second actor is represented by team members whose goal is "job satisfaction"; the Scrum Master whose main goal is the "efficient resolution of impediments".

Finally, the main objective seeking customers, the fourth stakeholder is its own satisfaction. This model suggests evaluate the quality of development processes considering the views of the different stakeholders involved, describing the indicators that are appropriate to each of these profiles.

On the other hand, it is also available the COBIT Model(Control Objectives for Information and Related Technology) [15], a tool that represents a particular collection of documents which can be classified, generally accepted as the best practices for the management, control and IT security. In COBIT, these domains is called: Plan and Organization (PO); Acquisition and implementation (AI); Delivery and Support (DS); Monitoring and Evaluation (ME). Through these four domains, COBIT has identified 34 IT processes. Through these four domains, COBIT has identified 34 IT processes and for each of these domains it defines goals and metrics to determine and measure their results and performances, based on the principles of balanced scorecard business (BSC).

Finally, among the models that apply to processes of software development it is important to emphasize the CMMI (Capability Maturity Model Integration) [16], a model for the improvement and evaluation of processes for development, maintenance and operation of software systems. CMMI has four disciplines to choose from: Systems Engineering (SE), Software Engineering (SW), Processes and Products Development (IPPD) and Distribution (SS). The model itself has two representations. One of them is

the staged representation in which it is centered a set of process areas, which are organized by maturity levels (1-5), while in the continuous representation, each process area are classified in terms of capacity levels (0-5). CMMI and agile methods have also been compared in several studies [17] [18] [19], for example, Paulk [20] suggests that the use of stories XP, at the customer's premises and continuous integration can comply with the requirements management objectives CMMI-SW. Moreover, in his study, Turner and Jain [21] determined that several of the components of CMMI and agile methods were in conflict, most of them related to organizational processes.

Among the presented models, it is observed that there is no proposal which allows quality assessment of agile processes themselves. Therefore, it is presented below the QuAM design, an approach which aims to provide a method of evaluation to determine the quality of software development processes based on Agile and its resulting products.

3 QuAM: Quality Evaluation Model of Agile processes

3.1 Design of the proposal

QuAM defines a scheme of components to set up a quality assessment model that provides an objective measure of the quality of the agile process implemented in any given project, allowing to obtain the agile profile associated with it. In this first instance, the model structure is presented taking into account only the dimension to the process level. Thus, the proposal that has been called Quam Level 1 provides a metric tree ($M_i, i=1...4$) composed by measurable attributes (A_i) through a series of criteria with associated measures. Then, according to the scope of this article, the dimension of the quality model presented here will evaluate the following components based on:

- Metrics 1 - Election of Life Cycle: The life cycle of a software project defines the order of the process activities. Quam will consider better Iterative life cycles and incremental over others. Focus will be placed on the implementation of the process, not in the documentation generated. The attributes and criteria to be evaluated are presented in Table 1.

Table 1. Attributes and Metrics Criteria: Election of Life Cycle

<i>Positive Attributes</i>		<i>Negative Attributes</i>	
<i>A1.1 - Value to Iterative and Incremental Life Cycle</i>		<i>A1.2 - Value to Waterfall Life Cycle</i>	
Not complete iterations are performed, but new features are added to the product	0	The project is divided into strictly sequential steps.	-2

Table 1 (cont.). Attributes and Metrics Criteria: Election of Life Cycle

In each iteration, the product is revised and improved through refactoring.	1	Phases are executed simultaneously.	-1
At each iteration, not only the functionality are improved, but also new are added to the product.	2	At the end of each phase, it is possible to make a backtracking and improve the defined in the previous stage.	0

- Metrics 2 - Assessment of Team: The human component of the project to assess must have adequate skills to the agile philosophy, and the company must have the means to achieve it. For QuAM will be important to evaluate the flow of communication between team members and the ability to face the same agile practices. The attributes and associated criteria to this metric are presented in Table 2.

Table 2. Attributes and criteria associated with Metric: Assessment of Team

<i>Positive Attributes</i>		<i>Negative Attributes</i>	
<i>A2.1 - Value to team meetings</i>		<i>A2.2 - Value of compliance schedule</i>	
No meetings are held in all iterations.	0	The schedule is adapted according to the changes and needs that arise throughout the project.	-3
In each iteration a meeting is done virtually at least.	1	Control milestones are set out in the schedule and changes can be defined on the scheduled dates.	-1
In each iteration, at least one meeting is held with the physical presence of the entire team.	3	The schedule established by stages is strict and does not allow changes.	0
<i>A2.3 - Value to the definition of roles.</i>		<i>A2.4 - Value to the process by over the team.</i>	
No roles are defined for individuals.	0	Activities, deliverables and development and management tools are defined for the project.	-3
A clear definition of roles is performed on individuals team.	1	Activities and project deliverables are defined.	-2
A clear definition of roles and responsibilities is done between team members.	2	Activities for each iteration are defined in the project.	-1

Table 2 (cont.). Attributes and criteria associated with Metric: Assessment of Team

A clear definition of roles, responsibilities and interaction between team members are made.	3	Activities for the project are defined but not at the level of each iteration	0
--	---	---	---

- Metric 3 - Production capacity of deliverables: QuAM will evaluate the frequency with which the project produces deliverables versions from the product to the customer. In this component, it will take into account the compliance with the lead time and the validity of each deliverable, favoring those projects whose validation has been automated. The change management process will be also measured the change management process will be also measured about the product and the verification process implementation and validation of them. In Table 3, the attributes and criteria that are considered for this metric are included.

Table 3. Attributes and criteria associated with Metric: Production capacity of deliverables

<i>Positive Attributes</i>		<i>Negative Attributes</i>	
<i>A3.1 - Value of the use of change management tools.</i>		<i>A3.2 - Value to requirements management and requisites.</i>	
There is a single project in the change management tool with a single workflow shared by all team members.	0	The Document of Software Requirements Specification (SRS) is updated simultaneously with the software	-3
There is a unique project in the change management tool but not all team members have their workflow (branch).	1	SRS is updated only if new requirements are added to Software	-1
There is a unique project in the change management tool used, and workflows (branches) are administered by each team member involved	3	SRS can not be upgraded, and must be strictly enforced.	0
<i>A3.3 - The value to functional product.</i>		<i>A3.4 - Value to documentation.</i>	
Generate deliverable upon project completion without making testing.	0	It requires detailed documentation at project start.	-3
Generate deliverable with manual testing after each iteration.	1	It requires only documentation needed at the beginning of each iteration.	-1

Table 3 (cont.). Attributes and criteria associated with Metric: Production capacity of deliverables

Generate deliverable with testing automated and integrated with other functions after each iteration.	3	No documentation is required to begin implementing the functionality included in one iteration.	0
---	---	---	---

- Metrics 4 - Customer communication: The quality model proposed will propitiate the incorporation of the client, as an active member in all stages of the project. Thus, this metric will assess implementation of regular communication mechanisms between the client and the team.

Table 4. Attributes and criteria associated with Metric: Customer Communication

<i>Positive Attributes</i>		<i>Negative Attributes</i>	
<i>A4.1 - Assess collaboration with the customer.</i>		<i>A4.2 - Assess contract negotiation.</i>	
Customer collaborates to team demand.	0	There is detailed recruitment at the beginning and no changes accepted.	-3
Customer is part of the team, answers queries and plans iterations.	1	The contract requires consider changes during the project.	-1
Customer is part of the team. He responds consultations, planning iterations, and collaborates on writing and testing requirements	3	The contract exists but does not affect the level project development process.	0

It is worth mentioning that for the design of this proposal, and taking into account defined in [22], positive attributes were considered (those who try to emphasize), and negative attributes (those who try to belittle). Thus, the positive attribute is measured on a scale from 0 to 3, and the negative attribute on a scale from -3 to 0. Thus, each metric could obtain a measure of -3, in the case that both attributes take the worst value (-3 for negative and the positive attribute 0), and 3, in the case that both attributes take the best value (0 negative attribute and 3 positive attribute). If a zero or near zero value is obtained, it means that the measurement values are not significantly above the positive negative.

Therefore, and taking into account the details of the associated criteria for the final value of each metric, it must consider both the corresponding measure to positive as the associated to the negative and the sum of its values, shown in (1):

$$M_i = M(A_{i.1}) + M(A_{i.2}) \quad i=1..4 \quad (1)$$

For example, the metric 1 (M1) - Election of the life cycle is measured by adding the measure of the value that the process gives the cycle of iterative and incremental life (A1.1) over the life cycle cascade (A1. 2).

3.2 Validation: Experience design

It is also necessary, design the process of validation of information that QuAM provides with the tools necessary to do so. To do this, in principle, the PYMES companies of the NEA Software Industry are convened, to assist in the validation of QuAM with real production environments to detect successes or issues to be improved in the model definition.

One of the instruments to be used will be an online survey with closed questions referred to the SW development processes of these companies. The same, it was designed and implemented through Google Forms, to facilitate dissemination among participants of experience and maximum reliability in the process of gathering information. The target population is made up of a group of 15 NEA companies, characterized by work on development projects web applications.

Currently the validation process has started with 25% of companies invited to take part of it. And it is expected that once the process of data collection is completed, an analysis of the obtained will be conducted to generate partial reports to determine the level of quality associated with agile business processes. The process will not end there, but the information obtained must be filed with the involved to achieve the feedback to determine if the concluded from the use of the proposed model is approximately or not, the reality perceived by the companies.

4 Conclusions and Future Work

The main contribution of this work is the preliminary definition of the components that form part of a quality model that helps to assess the quality of agile processes in PYMES dedicated to software development. There are several papers in the literature with the aim of improve the quality of the development process, submit proposals to adapt norms and standards to the philosophy of agile methodology. However, they are not specifically focused on the evaluation of the results obtained by processes under the agile philosophy. Thus, in principle, the QuAM presentation as a new approach to quality model will allow to start the cycle of quality assessment in real software projects guided through streamlined processes.

As future work, it is intended to obtain results of the validation experience presented in this article. And based on that, start defining a framework, including guidelines and practical guides, whose objective is the automation of measuring the quality of software projects based on agile processes.

5 Acknowledgements

This work was done under the accredited research project "Assessment Framework for Software Quality" EIUTIRE0002205TC of UTN. It should also be noted that the article is part of the activities planned in the Project on Technology and Social (PDTS) Development presented, "Contribution to the competitiveness of Software development NEA companies", IP253, evaluated and approved by the National Council of Scientific and Technical Research of Argentina (CONICET).

It is necessary to acknowledge the assistance of researchers Engineer Nicolás Tortosa, Blas Cabas Geat and Maximiliano Ulibarrie.

6 References

1. Navarro, J. M., & Garzías, J. (2010). Experiencia en la implantación de CMMI-DEV v1. 2 en una micropyme con metodologías ágiles y software libre. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software, 6(1), 6-15.
2. Reporte anual sobre el Sector de Software y Servicios Informáticos de la República Argentina. OPSSI, Abril 2016. Disponible en <http://www.cessi.org.ar/descarga-institucionales-2007/documento2-130347cd83ae771a9f3db3da5407269a>
3. Mas A., Amengual E. (2005). "Las mejoras de los procesos de Software en las pequeñas y medianas empresas (pymes). Un nuevo modelo y su aplicación a un caso real". Revista Española de Innovación, Calidad e Ingeniería del Software, Vol.1, No. 2
4. Pasini, A. C., Esponda, S., Bertone, R. A., & Pesado, P. (2008). "Aseguramiento de Calidad en PYMES que desarrollan software." XIV Congreso Argentino de Ciencias de la Computación.
5. Pflegger, S. (2002) "Ingeniería de Software. Teoría y Práctica." Pearson Education.
6. Letelier, P., Penadés, P. (2006) "Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)" Técnica Administrativa, Buenos Aires. ISSN 1666-1680
7. Alliance, A. (2001). "Agile manifesto". Disponible en <http://www.agilemanifesto.org>
8. Rujana, M., Romero Franco, N., Tortosa, N., Tomaselli, G., & Pinto, N. (2016, May). Análisis sobre adopción de metodologías ágiles en los equipos de desarrollo en pymes del NEA. In *XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina)*.
9. Córdoba, J., Cachero, C., Calero, C., Genero, M., & Marhuenda, Y. (2007, October). Modelo de Calidad para Portales Bancarios. In *XXXIII Conferencia Latinoamericana de Informática (CLEI'07)*.
10. Behkamal, B., Kahani, M., & Akbari, M. K. (2009). Customizing ISO 9126 quality model for evaluation of B2B applications. *Information and software technology*, 51(3), 599-609.
11. ISO/IEC 9126: "Software Engineering - Product quality", International Organization for Standardization, 2000.
12. Matalonga, S., & Rivedieu, G. (2015). AGIS: hacia una herramienta basada en ISO9001 para la medición de procesos ágiles. *Computación y Sistemas*, 19(1), 163-175. Disponible en <http://www.agilemanifesto.org/iso/es/> Último acceso 06/2016
13. International Organization for Standardization. (2000). ISO 9001: 2008: Quality Management Systems-Requirements. International Organization for Standardization.
14. Cohen, D., Lindvall, M., & Costa, P. (2003). Agile software development. DACS SOAR Report, 11.

15. Paulk, M. C. (2001). Extreme programming from a CMM perspective. *Software, IEEE*, 18(6), 19-26.
16. Piattini, Oktaba, Orozco, "COMPETISOFT. Mejora de procesos software para pequeñas y medianas empresas", Editorial Ra-Ma, Año 2008.
17. D. Kane and S. Ornburn, "Agile Development: Weed or Wildflower?" *CrossTalk, The Journal of Defense Software Engineering*, <http://www.stsc.hill.af.mil/crosstalk/2002/10/kane.html>, 2002. (1.3.2006)
18. J. Nawrocki, W. Bartosz, and A. Wojciechowski, "Toward Maturity Model for eXtreme Programming," In proceedings of the 27th Euromicro Conference, pp. 233-239, 2001.
19. M. C. Paulk, "Extreme Programming from a CMM Perspective," *Software*, vol. 18, issue 6, pp. 19-26, 2001
20. R. Turner and A. Jain, "Agile Meets CMMI: Culture Clash or Common Cause," In proceedings of the Second XP Universe and First Agile Universe Conference on Extreme Programming and Agile Methods - XP/Agile Universe, pp. 153-165, 2002.
21. Turner, R., & Jain, A. (2002, August). Agile meets CMMI: Culture clash or common cause?. In *Conference on Extreme Programming and Agile Methods* (pp. 153-165). Springer Berlin Heidelberg.
22. Mendes Calo, K., Estevez, E. C., & Fillottrani, P. R. (2009). Un framework para evaluación de metodologías ágiles. In *XV Congreso Argentino de Ciencias de la Computación*.

Metodología para el desarrollo de software en proyectos de I+D en el nivel universitario basada en Scrum

Luciano Straccia, Pablo Pytel, María Florencia Pollo-Cattaneo

Grupo de Estudio en Metodologías de Ingeniería de Software (GEMIS). Universidad Tecnológica Nacional. Facultad Regional Buenos Aires. Argentina.

lstraccia@frba.utn.edu.ar, ppytel@gmail.com, flo.pollo@gmail.com

Resumen. La investigación en el ámbito de las ciencias informáticas es, en muchas ocasiones, inseparable del desarrollo, constituyendo cada proyecto en un proyecto I+D (investigación y desarrollo). El desarrollo de software en el marco de un proyecto I+D debe ser una actividad llevada a cabo con estrategias y metodologías apropiadas que permitan dar respuestas a las necesidades del proyecto y que se adapten a las características institucionales y de los recursos humanos con los que se cuenta, partiendo de experiencias y herramientas brindadas por la ingeniería del software. A partir de la experiencia inicial en el desarrollo de software dentro su proyecto el grupo GEMIS, de la Universidad Tecnológica Nacional, Facultad Regional Buenos Aires (UTN-FRBA), se propuso partir de una metodología ágil y, revalorizando algunos aspectos como las definiciones funcionales y la arquitectura, definir una metodología acorde a las necesidades de proyectos de I+D en el ámbito universitario.

Palabras Claves: Desarrollo de Software, Metodología, Scrum.

1 Introducción

Dentro del ámbito de la Facultad Regional Buenos Aires de la Universidad Tecnológica Nacional (UTN-FRBA) se ha conformado el Grupo GEMIS (Grupo de Estudios de Metodología para Ingeniería de Software y Sistemas de Información), un equipo de personas con interés en la investigación vinculada a Ingeniería en Sistemas de Información y en Tecnología Aplicada a la Educación. Así se ha puesto en marcha en Mayo del 2015 el Proyecto de Investigación y Desarrollo (PID) denominado “Intervenciones tecnológicas en dispositivos didácticos con herramientas de tecnología informática” cuyo objetivo es describir y analizar el uso de la tecnología informática en las intervenciones didácticas de los profesores de las asignaturas de la carrera de Ingeniería en Sistemas de Información (ISI) de la UTN-FRBA y desarrollar nuevos artefactos tecnológicos que favorezcan la mejora en las intervenciones didácticas y una metodología de implementación.

La construcción de soluciones de Sistemas de Información y Tecnología Informática (SI/TI) para el ámbito educativo no está exento de problemas habituales

en el desarrollo de software. Sin embargo, en el marco de un PID, estos problemas tradicionales se conjugan con otros conflictos propios de la investigación en el ámbito universitario. GEMIS ha identificado algunas problemáticas específicas en su propio proyecto que requieren la definición de una metodología de desarrollo precisa y acorde a sus necesidades.

Este trabajo presenta las características del PID y los recursos humanos asociados al mismo (sección 2), realiza una descripción de la organización inicial del trabajo del equipo (sección 3), recorre las bases teóricas y metodológicas en las cuales se sostiene la propuesta de este trabajo (sección 4), presenta una metodología para el desarrollo de software en equipos de I+D (sección 5) y una evaluación de la experiencia de su implementación (sección 6). Finalmente, se indican las conclusiones obtenidas y futuras líneas de trabajo (sección 7).

2 Características del proyecto y los recursos humanos

Los proyectos de investigación y desarrollo requieren de recursos humanos que permitan continuarlos a lo largo del tiempo. La conformación de un equipo de investigadores y desarrolladores en el ámbito universitario tiene diversas fuentes de recursos posibles: docentes, alumnos y graduados. Pocos de estos recursos tienen una dedicación exclusiva o semiexclusiva al proyecto y aquellos que pudieran poseerlo se encuentran más abocados a la gestión del mismo (directores y codirectores) que al desarrollo tecnológico en sí mismo; además otros investigadores con una dedicación horaria superior a la media de los integrantes se abocan a aspectos propios de investigación que, por tratarse de trabajos asociados a la vida académica universitaria y su impacto en la didáctica, no tienen una vinculación directa con el desarrollo, excepto que son generadores de requerimientos y necesidades y acompañan la implementación de los artefactos construidos.

Indagando especialmente en cada uno de las diferentes fuentes de recursos humanos para el proyecto se ha observado además que:

- los alumnos de los primeros niveles de la carrera de ISI no poseen los conocimientos técnicos requeridos para el desarrollo de software con la complejidad que el proyecto amerita y, si lo poseen, se trata de conocimientos obtenidos por fuera del marco institucional de su carrera, como puede ser aprendizaje individual, carreras de grado previas o experiencia laboral;
- los alumnos de los niveles superiores de la carrera poseen dificultades para sostener su participación activa y continua en el proyecto por sus compromisos laborales (ya que se observa que los alumnos de niveles superiores ya poseen participación en la vida laboral en su mayoría) y esto implica, en algunos casos, el abandono de los proyectos en los que se encuentran involucrados o, en el mejor de los casos, en una disminución de su participación;
- para los graduados se observa el mismo inconveniente asociado a los alumnos avanzados.

Además, en particular el PID de GEMIS no aspira a construir un producto software único, sino diferentes aplicaciones software a través de las cuales involucrarse en la mejora de la didáctica, por lo cual no se trata de la construcción de un producto sino de múltiples productos que, integrados, tengan un impacto positivo y permitan cumplir los objetivos. En este contexto, se lleva adelante un Desarrollo de Software basado en Componentes (CBSE, *Component Based Software Engineering*), cuyo concepto se tratará más adelante.

Por otra parte, los requerimientos asociados al desarrollo son identificados o surgen a lo largo del ciclo de vida de la investigación. Los resultados obtenidos a partir de la implementación de un producto construido son fuente de análisis para el propio proyecto de investigación y permiten la definición de nuevos requerimientos. GEMIS se encuentra en un trabajo continuo para el relevamiento de dificultades asociadas al proceso de enseñanza y aprendizaje, lo que constituye un dominio de aplicación dinámico, con requerimientos cambiantes y cuyos *stakeholders* (interesados) no necesariamente conocen acerca del desarrollo de software.

Finalmente, el nivel de remuneración de los desarrolladores de los proyectos de investigación y desarrollo en la Universidad es bajo, tratándose de becas o, en un alto porcentaje, de integrantes de actividades voluntarias.

Además dado que el objetivo de GEMIS (y deseable en todo PID) involucra la formación de sus recursos humanos, se incorporan al proyecto recursos con poco *expertise* o con conocimientos tecnológicos básicos.

3 Organización inicial del equipo de desarrollo

Acorde a la necesidad de realizar en una primera etapa un relevamiento de las situaciones problemáticas existentes y, a partir de ello, generar soluciones que permitan superar estas situaciones, el inicio del proyecto se vio involucrado en una metodología de estilo cascada [1], con una definición lo más avanzada posible de requerimientos y, a partir de ello, la derivación al equipo de programación para la construcción de los respectivos productos que atendieran esos requerimientos.

El uso de esta metodología dio impulso al proyecto, permitió trabajar en los primeros aspectos de la investigación, generar definiciones acordes a lo requerido por los desarrolladores e, inclusive, a partir de clarificar el alcance vigente, la posibilidad de motivar a nuevos recursos a incorporarse al proyecto.

Sin embargo el crecimiento de las problemáticas halladas, el crecimiento de los requerimientos de software, las dificultades con los recursos ya mencionados y el retraso de las fechas de entrega, han generado la necesidad de repensar la metodología a utilizar. La metodología tradicional en cascada ha sido útil y relevante para impulsar el proyecto en sus inicios, dando a partir de su crecimiento paso a la necesidad de redefinición metodológica. A medida que el proyecto fue avanzado la definición de requerimientos formal y tradicional dio lugar al uso de historias de usuario [2] que “son descripciones cortas de una necesidad de un cliente del software que estamos desarrollando. Su utilización es común cuando se aplican marcos de trabajo ágiles” [3] y buscan favorecer la construcción del producto software sin una dedicación temporal excesiva para el análisis de los requerimientos.

El equipo de desarrollo previsto al inicio del proyecto poseía conocimientos en cierto lenguaje de programación (PHP) y los primeros productos fueron desarrollados bajo dicho lenguaje. Sin embargo, nuevos recursos humanos mostraron su interés en participar en el proyecto lo que planteaba la disyuntiva de descartar su participación o incluirlos diversificando los lenguajes de desarrollo. GEMIS ha optado por esta segunda opción ya que el lenguaje de programación (y la tecnología en general) no operan como restricciones del proyecto, excepto por su grado de factibilidad de implementación y de la disponibilidad de un servidor para su despliegue.

4 Bases teóricas y metodológicas de la propuesta

En este capítulo se presentan las bases teóricas y metodológicas en las cuales se sostiene la propuesta del presente trabajo. En una primera parte se describen las características del Desarrollo de software basado en Componentes (sección 4.1), luego se describen generalidades del desarrollo ágil (sección 4.2), la metodología Scrum (sección 4.3) y el papel de la arquitectura de software (sección 4.4). Finalmente se plantea la Gestión de personas como pool de recursos (sección 4.5).

4.1. Desarrollo de Software basado en Componentes

El desarrollo de software basado en componentes (CBSE, *Component Based Software Engineering*) es aquel que está fundamentado en la producción de diversas piezas de software ensambladas de una manera integral que permita el funcionamiento del sistema software como un todo. En este contexto, se define a un sistema como “un conjunto de mecanismos y herramientas que permiten la creación e interconexión de componentes software, junto con una colección de servicios para facilitar las labores de los componentes que residen y se ejecutan en él” [4].

Un componente es una pieza de código preelaborado que encapsula alguna funcionalidad expuesta a través de interfaces estándar. Szyperski, exarquitecto de software de Microsoft, define componente como una “unidad de composición de aplicaciones software que posee un conjunto de requisitos y que ha de poder ser desarrollado, adquirido, incorporado al sistema y compuesto por otros componentes, de forma independiente en tiempo y forma” [5]. Entre las ventajas que pueden hallarse en el uso del CBSE se encuentra la reutilización de software, la simplificación de las pruebas, la simplificación del mantenimiento del sistema y una mayor calidad de los componentes [6].

El aspecto de mejora de la calidad es una variable fundamental que ha llevado a GEMIS a implementar esta estrategia de desarrollo de software, dado que permite la construcción de un componente que luego puede ser reemplazado (o mejorado) por un nuevo componente, el cual podría ser desarrollado por un programador con más *expertise* que el programador original.

4.2. Desarrollo ágil

Poole [7] define el desarrollo ágil “como aquel que, en comparación con el desarrollo tradicional, provee beneficios de mayor flexibilidad, retorno de inversión más alto, realización más rápida del retorno de la inversión, más alta calidad, (y) mayor visibilidad” [8]. El desarrollo ágil no sólo ha otorgado diferentes metodologías (Scrum, XP, ASD, Crystal, entre otras) sino que también ha brindado técnicas y herramientas aplicables independientes de la metodología (algunas de las cuales han sido inicialmente aplicadas en el equipo GEMIS).

El Manifiesto Ágil [9] reúne las motivaciones y principios de un desarrollo ágil, entre los cuales se encuentra la satisfacción del cliente con entrega temprana y continua de software, considerando al software funcionando como la medida principal de progreso; adaptación a requisitos cambiantes; confianza en los individuos (con el entorno y apoyo necesario) y simplicidad en el desarrollo. Por otro lado, a partir de las características de GEMIS cobran especial relevancia algunos otros aspectos expresados en el Manifiesto: valoración de la comunicación (con conversaciones cara a cara); promoción del desarrollo sostenible por medio de los procesos ágiles; mejora en las arquitecturas, surgimiento de requisitos y soluciones a partir de la actividad de equipos auto-organizados; y reflexión periódica del equipo sobre cómo ser más efectivo para, a continuación, ajustar y perfeccionar su comportamiento en consecuencia.

4.3. Scrum

Scrum es una metodología ágil que se basa “en la teoría de control de procesos empírica o empirismo. El empirismo asegura que el conocimiento procede de la experiencia y de la toma de decisiones basándose en lo que se conoce. Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo” [10].

El equipo Scrum se compone de un responsable del producto (dueño del producto, *Product Owner*), el equipo de desarrollo (*Development Team o Scrum Team*) y el *Scrum Master*, que es responsable de asegurar que Scrum es entendido y adoptado correctamente.

Scrum supone los siguientes artefactos: lista del producto (*Product Backlog*), lista del sprint (*Sprint Backlog*) y un incremento. Un *sprint* es cada iteración del proceso de desarrollo. La lista del producto contiene todas las funcionalidades previstas para el producto y la lista del *sprint* contiene el listado de aquellas funcionalidades que serán incluidas en la iteración siguiente (sin considerar ni definir en cuál sprint serán incluidas las restantes). El incremento es “la parte de producto producida en un sprint, y tiene como característica el estar completamente terminada y operativa, en condiciones de ser entregada” [11], independientemente de si finalmente el *Product Owner* decida realizar la entrega al cliente o destinatario del mismo.

Los *sprints* comienzan con una reunión donde se planifica el mismo (*Sprint Planning Meeting*) y finalizan con una revisión del sprint en cuanto al producto (*Sprint Review*) y una retrospectiva (*Sprint Retrospectiva*), que implica una revisión del sprint en cuanto a personas, relaciones, procesos y herramientas [10]. Durante el sprint se llevan a cabo reuniones diarias (*Daily Meeting*) donde se comparte información sobre el estado de la iteración y las dificultades presentes

4.4. El papel de la arquitectura de software

La arquitectura del software es la organización fundamental de un software, formada por sus componentes, las relaciones entre ellos, el contexto en el que se implantarán, y los principios que orientan su diseño y evolución [12]. La bibliografía asociada a Scrum generalmente no hace referencia a las decisiones arquitecturales y las discusiones existentes acerca de cómo incorporar la arquitectura en los proyectos dirigidos por Scrum son confusos. Dado que muchos adeptos a esta metodología provienen del campo de la programación del software, se pueden hallar manifestaciones acerca de la arquitectura como emergente del código fuente. Sin embargo, si bien la arquitectura puede emerger del código (uno los principios del manifiesto ágil se vincula a los diseños emergentes de los equipos de desarrollo), es importante la formalización (que no implica necesariamente documentación ni aspectos generalmente definidos como “burocráticos”) de la arquitectura a fines de poder ser compartida por la totalidad del equipo, especialmente cuando se espera llevar a cabo de un desarrollo basado en componentes.

Mago y Alferez [13] proponen incorporar la arquitectura del software en el proceso de Scrum como un *sprint* 0, en el cual se define la arquitectura que guiará el proyecto; sin embargo esto no es factible utilizarlo cuando se piensa el software con una arquitectura cambiante ya que los autores prevén que el arquitecto de software guíe al equipo en los *sprints* subsiguientes a fines de orientarlos para la adaptación a la arquitectura predefinida, pero no la readaptación o reestructuración arquitectónica.

4.5. Gestión de personas como *pool* de recursos

La dirección de las actividades de desarrollo del proyecto implica gestionar una cartera de proyectos constituyendo una responsabilidad similar a un PMO (*Project Management Office*, Oficina de Gestión de Proyectos) que es definida por el PMBOK® como “una unidad de la organización para centralizar y coordinar la dirección de proyectos a su cargo” [14], siendo una unidad diseñada para dirigir y controlar el desarrollo de un grupo de proyectos informáticos de manera simultánea. Considerando la caracterización de PMO realizada por el Project Management Institute (PMI), GEMIS trabaja con el marco de trabajo correspondiente a una oficina de respaldo, servicios y controles del proyecto que “proporciona los procesos que faciliten un apoyo continuo para las labores de gestión del proyecto, programa o cartera en toda la organización. Aplica la gobernanza, procesos, prácticas y herramientas establecidas por la organización y brinda un apoyo administrativo para las labores del proyecto, programa o cartera dentro de su dominio” [15].

Casey y Peck [16] plantean tres modelos de PMO: tipo estación meteorológica, torre de control y pool de recursos, en las cuales el foco principal de trabajo son la generación de informes de acompañamiento de indicadores, el control de proyectos y gestión del conocimiento y la gerencia y aplicación de recursos, respectivamente [17]. GEMIS adopta, por las características de sus proyectos y sus recursos, una dirección asociada al pool de recursos, también denominada en ocasiones Escuadrón de Combate [17]. Debe tenerse en cuenta que la metodología implantada por GEMIS no considera gerentes de proyectos, por lo cual la asignación por parte de la PMO de recursos a los proyectos implica la asignación directa bajo el modelo de pool de

recursos como un inventario de recursos disponibles. Según el DRAE, un pool es un “grupo de personas entre las que se reparte una tarea determinada” [18].

5 Metodología propuesta

GEMIS ha considerado importante, a partir de las experiencias obtenidas en la organización inicial del equipo de desarrollo (descritas en la sección 3) y las características del proyecto y los recursos humanos existentes (descritos en la sección 2), la definición de una metodología basada en Scrum y adaptada a los requerimientos propios del equipo: desarrollar software basado en componentes, *sprints* que superen los tiempos previstos en las reglas del Scrum (donde generalmente se considera mandatorio una ventana de tiempo máxima de 1 mes) y revalorizar el papel de cierta documentación funcional como mandatoria y una definición cambiante de arquitectura de software.

El aspecto funcional cobra mayor valor por la necesidad de comunicar detalles respecto del dominio a la comunidad académica como parte de las responsabilidades de un grupo de investigación y desarrollo de una universidad. En tanto el valor de los aspectos arquitectónicos ha sido desarrollado previamente al hacer referencia al desarrollo basado en componentes.

Ambos aspectos se han incorporado a los requerimientos del incremento, así el resultado del sprint consiste en un incremento de software, un documento de arquitectura y una breve descripción de la funcionalidad incorporada, realizada bajo la forma de historias de usuario, incorporándose estos productos a los artefactos resultantes (Figura 1)



Figura 1: Artefactos GEMIS

La lista de producto y la lista de sprint, junto con la información general del producto, conforman para GEMIS un documento denominado Visión de Producto y del cual puede hallarse una plantilla en <https://goo.gl/HgHBuP>.

El proceso de Scrum no se ve modificado respecto al proceso original, exceptuando las duraciones de los *sprints* llevando cada sprint a una duración de 3 meses, período definido para la *Scrum Meeting* y la periodicidad de las *Daily Meeting* que dejan de ser diarias y pasan a ser semanales (*Weekly Meeting*).

Los *sprints* de cada uno de los productos se llevan a cabo en fechas idénticas, de manera de permitir utilizar el *Scrum Meeting* para la reasignación de recursos provenientes del pool de recursos con el que se cuenta, posibilitando incorporar a una

nueva iteración de un producto un recurso que estuviera, hasta la iteración anterior, involucrado en otro proyecto.

La asignación de recursos a cada uno de los proyectos se hace a partir de la presentación del estado general de cada uno de los productos y la revisión de productos por crearse o tareas asociadas a investigación tecnológicas (listado proporcionado en una Visión de Proyecto y que es discutido en un *Project Meeting*, coincidente en tiempo y espacio con el *Scrum Meeting* (Figura 2)). De esta manera, en el *Scrum Meeting* se presentan los diferentes productos existentes y su estado de avance, los requerimientos pendientes y los recursos podrá seleccionar en cuál producto desean continuar su participación siendo asignados al mismo previo a la generación de un nuevo *sprint*, constituyendo un *Project Meeting*.

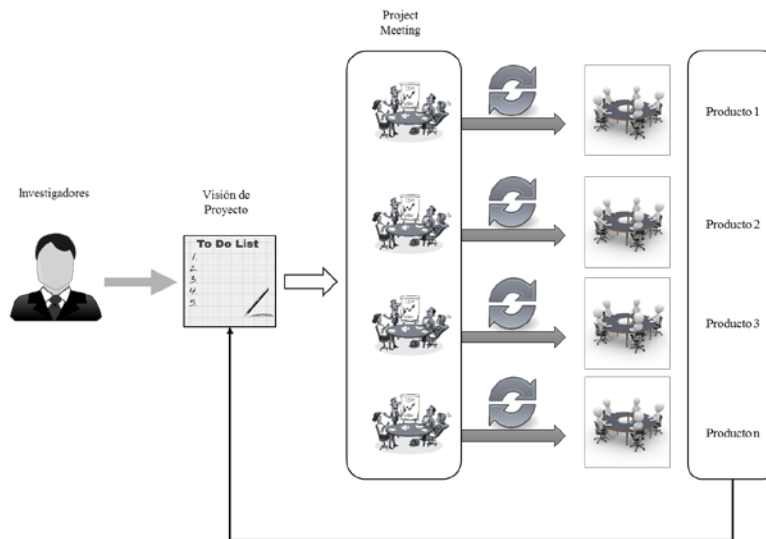


Figura 2: Metodología GEMIS

6 Experiencia de implementación

Desde el inicio del PID se han presentado diversos trabajos que avalan la perspectiva de trabajo del grupo GEMIS [19],[20],[21] y se inició el desarrollo de diversos productos: un sitio web para el análisis de evaluaciones en el ámbito universitario [22], desarrollo de soluciones móviles dirigidas a la evaluación diagnóstica como parte del proceso de enseñanza y a la resolución de problemas [23],[24] y el desarrollo de una aplicación móvil para la autoevaluación por parte del alumno [25].

Si bien los desarrollos previstos han mostrado avances, se tornaba dificultoso constituir versiones estables de las aplicaciones y los tiempos reales siempre superaban, en gran medida, a los tiempos previstos. A partir de la nueva perspectiva de trabajo y la implementación de la propuesta metodológica explicada en este trabajo

se ha logrado configurar un mapa de aplicaciones en las cuales, incluso, a partir de las discusiones generadas en las primeras reuniones de equipo bajo esta nueva metodología, se logró acordar un nuevo componente, denominado Datos Maestros (DM), que prestaría servicios a las diferentes aplicaciones, definiéndose que se realizaría como Servicios REST.

Respecto de cada producto, se ha logrado llegar a una documentación precisa respecto de los alcances esperados, pudiendo constituir todos los artefactos definidos en la metodología y posibilitando la disponibilidad de software funcionando que se encuentra en las primeras etapas de puesta en producción. El producto CW que se inició en el año 2014 tuvo gran deficiencia en la construcción del software ya que, a pesar de presentarse algunos avances, estos no cumplían con las expectativas. A partir de la puesta en marcha de la metodología, se logró finalizar una primera versión del sitio web (primer *sprint*) e incorporar el módulo de reportes (segundo *sprint*) encontrándose actualmente en el desarrollo de su tercer *sprint*. Asimismo, se logró poner en marcha un proyecto de centralización de datos maestros (DM) y generar una primera versión de los productos RR y TD [23],[24], constituir el equipo necesario para el inicio de un nuevo producto que se encuentra transitando su primer *sprint* (producto AS, Asistencia) e integrar el trabajo de GEMIS con el trabajo de una institución terciaria con carreras asociadas al desarrollo de software tal como el Instituto Superior Da Vinci a través del cual se pudo iniciar el producto IA (Intercambio de Archivos), que se encuentra también en el primer *sprint*.

Finalmente, la metodología propuesta en el presente trabajo permitió la incorporación de nuevos recursos humanos al proyecto que en principio no tenían una asignación prevista pero su participación en las *Project Meeting* permitió asignarlos a algunos de los productos con desarrollo vigente.

7 Conclusiones

Los proyectos de I+D en el ámbito universitario asociados al desarrollo de software no son ajenos a las problemáticas habituales existentes en la industria del software. Asimismo incorporan problemáticas propias del quehacer institucional universitario, las particularidades de los recursos humanos y nuevas obligaciones respecto a resultados esperados.

En este trabajo se presenta la adaptación de la metodología Scrum para el desarrollo de software para proyectos de I+D en el ámbito universitario, considerando especialmente la importancia del Desarrollo de Software Basado en Componentes, el papel de la arquitectura de software y la gestión de personas como pool de recursos.

A partir de la puesta en marcha de la metodología se produjeron avances importantes sobre uno de los productos iniciados (CW), se logró poner en marcha un proyecto de centralización de datos maestros (DM), generar una primera versión de nuevos productos (RR y TD), iniciar un nuevo producto (AS) y trabajar integradamente con otras instituciones

Como futuras líneas de trabajo se hace necesario continuar con la aplicación metodológica en los diferentes proyectos y productos existentes y verificar el cumplimiento de las diferentes metas planteadas. Además dadas las características de

los recursos humanos, será necesario realizar una gestión de los recursos no sólo para resolver problemas y necesidades presentes sino también previendo potenciales cambios y necesidades futuras, bajo una perspectiva de gestión del pool de talentos [26].

Referencias

1. Kimball, R., et al. *The Data Warehouse Lifecycle Toolkit*. John Wiley & Sons. (2011)
2. Ambler, S. *User Stories: An Agile Introduction*. (2014)
3. PMOInformatica. *Historias de usuario en 5 pasos*. (2014)
4. Fuentes, L.; Troya, J.M.; Vallecillo, A. *Desarrollo de Software Basado en Componentes*. ETSI Informática. Universidad de Málaga. Málaga, España.
5. Szyperski, C. *Component Software. Beyond Object-Oriented Programming*. Addison-Wesley. (1998)
6. Casal Terreros, J. *MSDN Microsoft. Desarrollo de Software basado en Componentes*.
7. Poole, D. *Do It Yourself Agile*. (2009)
8. Gimson, L. *Metodologías ágiles y desarrollo basado en conocimiento*. Universidad Nacional de La Plata. (2012)
9. *Manifiesto por el Desarrollo Ágil de Software*. Disponible en <http://agilemanifesto.org/iso/es/>
10. Schwaber, K. y Sutherland, J. *La Guía Definitiva de Scrum: Las reglas del Juego*. (2013)
11. *Scrum Manager*. Disponible en www.scrummanager.net
12. IEEE Std. 1471-2000.
13. Mago, E.; Harvey Alferez, G. *El Papel de la Arquitectura de Software en Scrum*. Revista Software Gurú Nro. 30. Noviembre 2010 - Enero 2011. Venezuela. (2011)
14. *PMBOK. Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK®)*. Tercera Edición. Project Management Institute, EE.UU. (2004)
15. PMI. *Informe Pulso de la Profesión: marcos de trabajo de la PMO*. PMI. (2013)
16. Casey, W. y Peck W. *Choosing the right PMO setup*. En *Network*, Febrero (2001).
17. Alonso González, A. *Como implantar una oficina de gestión de proyectos en su organización*. Visión Libros. Madrid. (2008)
18. *DRAE. Diccionario la Real Academia Española*.
19. Straccia, L.; Vegega, C.; Bernal, L.; Deroche, A.; Pytel, P.; Pollo-Cattáneo, M.F. *Intervención tecnológica no convencional en los dispositivos didácticos de la carrera de Ingeniería en Sistemas de Información*. XVII WICC. Universidad Nacional de Salta. (2015)
20. Straccia, L.; Acosta, M.; Pytel, P.; Pollo-Cattáneo, M.F. *Intervenciones tecnológicas no convencionales en dispositivos pedagógicos*. En *Libro de Trabajos IV Jornada de Enseñanza de la Ingeniería*. UTN, Avellaneda. (2014)
21. Straccia, L.; Vegega, C.; Pytel, P.; Pollo-Cattáneo, M.F. *Tecnología Informática para facilitar la labor docente en la formación por competencias en Ingeniería*. XII Congreso Internacional sobre el Enfoque basado en Competencias. Cartagenas de Indias. (2016)
22. Straccia, L.; Deroche, A.; Vegega, C.; Pytel, P.; Pollo-Cattáneo, M.F. *Software para análisis de instrumentos de evaluación en la formación por competencias*. Proceedings XX CACIC. Universidad Nacional de La Matanza. (2014)
23. Straccia, L.; Acosta, M.; Vegega, C.; Pytel, P.; Pollo-Cattáneo, M.F. *Mejora de la didáctica en la enseñanza de la Ingeniería a través del uso de nuevas aplicaciones sobre los dispositivos móviles*. TEYET. Univ Nacional del Noroeste. (2015)

24. Straccia, L.; Marino Aguirre, M.; Acosta, M.; Vegega, C.; Pytel, P.; Pollo-Cattáneo, M.F. El desarrollo de artefactos de tecnología informática como aporte a las intervenciones didácticas. III CONAIISI. UTN. Buenos Aires. (2015)
25. Deroche, A.; Acosta, M.; Vegega, C.; Bernal Tomadoni, L.; Straccia, L.; Pytel, P.; Pollo-Cattáneo, M.F. Diseño de aplicación móvil para asignatura de grado en Ingeniería en Sistemas de Información. III CONAIISI. UTN. Buenos Aires. (2015)
26. MalikehBeheshtifar; Fateme-BegomKamani-Fard. Talent Pool: A Main Factor to Success. En Interdisciplinary Journal of Contemporary Research in Business. Vol. 4. Nro 12. Abril 2013.

Implantación de Sistemas: Estudio Comparativo e Identificación de Vacancias en Metodologías Usuales

Marisa Panizzi^{1,2}, Alejandro Hossian^{1,3}, Ramón García Martínez¹

¹ Grupo de Investigación en Sistemas de Información. Departamento de Desarrollo Productivo Tecnológico. Universidad Nacional de Lanús, 29 de Septiembre 3901, (B1826GLC), Lanús, Buenos Aires, Argentina

² Departamento de Ingeniería en Sistemas de Información. Universidad Tecnológica Nacional .Facultad Regional Buenos Aires, Medrano 951 (C1179AAQ), C.A.B.A, Argentina

³ Grupo de Investigación en Aplicaciones de Sistemas Inteligentes en Ingeniería. Universidad Tecnológica Nacional – Facultad Regional Neuquén, Av. Pedro Rotter S/N Barrio Uno, (8318), Neuquén, Argentina

marisapanizzi@outlook.com, alejandrohossian@yahoo.com.ar, rgm@yahoo.com

Resumen. La implantación de sistemas es una de las fases del proceso de software donde la dimensión de interacción humana es relevante. Sin embargo, la debilidad de custodia de esta dimensión constituye un factor que en muchos casos lleva al fracaso el proyecto. En este trabajo, se reporta sistematización y cotejo de las actividades de implantación en las metodologías: IEEE 1074, Métrica V3, Proceso Unificado de Rational y SCRUM. Se propone un instrumento para su comparación con la fundamentación de las dimensiones de análisis utilizadas, y se da una interpretación preliminar de resultados con énfasis en áreas de vacancia.

Palabras Clave. procesos de software, implantación de sistemas, aspectos humanos, análisis comparativo de procesos.

1 Introducción

La evolución de la ingeniería de software ha permitido obtener un abanico de metodologías o estándares entre ellos, IEEE 1074 [1], Métrica versión 3 [2], Proceso Unificado de Rational [3], Scrum [4][5]. A pesar de esta evolución, autores como Vasconcelos y Werner [6] plantean que se presentan varios problemas relacionados con las descripciones de procesos en los modelos de procesos de software o en los estándares existentes. En [7] se plantea que los problemas existentes en los modelos de procesos de software son: (i) no existe una definición de un modelo que abarca la representación conjunta de los procesos, productos, personas y organización, (ii) no está formalizado en el proceso de software los aspectos como la organización del trabajo, las personas y sus interacciones, y (iii) no hay ningún proceso definido que incluya tanto la parte técnica y humana del proceso en el modelo.

The Stadish Group Report [8] plantea una serie de factores que influyen para el éxito o el fracaso de un proyecto entre los cuales se pueden enumerar el

involucramiento de los usuarios, la imprecisa definición de los requisitos, los recursos especializados, todos vinculados a los aspectos humanos en el desarrollo de software.

Los avances propuestos por la Ingeniería de Requisitos como por ejemplo estándares IEEE-830 [9], 29148-2011 [10] para especificación de requisitos, el Léxico Extendido del Lenguaje [11] han permitido resolver cuestiones vinculadas a los aspectos humanos en el primer eslabón de la cadena de construcción de software.

Otro de los eslabones del proceso de software que los autores sostienen que se evidencia una fuerte interacción humana dada entre los profesionales de sistemas y la comunidad usuaria, es en el proceso de implantación. A este proceso se le resta importancia por tratarse del último eslabón de la cadena de procesos de software. Esta desvalorización del proceso se refleja en que el profesional al que se le asigna el rol de implantador o responsable de la entrega del producto software al cliente, no posee las competencias socio-técnicas necesarias para desempeñarse en dicho proceso.

Este trabajo se desarrolla bajo la hipótesis que los proyectos de software fracasan por los aspectos humanos, la definición de roles, las interacciones entre los diferentes roles, las capacidades de las personas que desempeñan esos roles, entre otros. Los autores plantean como hipótesis de trabajo la dimensión humana y su impacto en el proceso de implantación. De esta hipótesis se deriva la pregunta de investigación: ¿si al realizar un estudio comparativo todas las áreas están cubiertas en el proceso de implantación de un sistema informático?.

En este contexto, este artículo presenta los procesos considerados (Sección 2), se describen y justifican las dimensiones consideradas para el análisis (Sección 3), se presentan los resultados de la comparación y se dan interpretaciones preliminares (Sección 4) y se formulan conclusiones y futuras líneas de trabajo (Sección 5).

2 Procesos considerados

En esta sección se presentan las metodologías que se han contemplado para el estudio comparativo, el estándar IEEE 1074 [1] (sección 2.1), la metodología Métrica versión 3 [2] (sección 2.2), el Proceso Unificado de Rational [3] (sección 2.3) y por último la metodología SCRUM [4][5] (sección 2.4).

2.1 IEEE 1074

El estándar IEEE 1074 [12] proporciona el conjunto de actividades que constituyen los procesos que son obligatorios para el desarrollo y mantenimiento de software. Se encuentra organizado en 17 procesos, que comprenden un total de 65 actividades. Los procesos se dividen en cuatro secciones lógicas o grupos de procesos.

El primer grupo está compuesto por el Proceso de Modelo del Ciclo de Vida del Software que proporciona actividades que se necesitan para identificar los modelos de ciclo de vida software candidatos y para seleccionar aquel modelo que se vaya a utilizar en el proyecto.

El segundo grupo está conformado por el Proceso de Gestión del Proyecto, que propone un conjunto de procesos de iniciación, supervisión y control del proyecto a lo largo de ciclo de vida del software.

El tercer grupo está compuesto por los procesos Orientados al Desarrollo, los Procesos de Pre-Desarrollo, los Procesos de Desarrollo y los Procesos de Post-Desarrollo del software.

El último grupo está compuesto por los Procesos Integrales, son aquellos procesos que se necesitan para completar con éxito las actividades de un proyecto.

El proceso de implantación está tratado en el proceso de instalación del grupo de procesos post-desarrollo perteneciente al grupo de Procesos orientados al Desarrollo.

Este proceso implica el transporte y la instalación de un sistema software desde el entorno de desarrollo al entorno de destino [1].

Las actividades del proceso de instalación propuestas en el estándar son: la distribución del software, la instalación del software, la carga de la base de datos (si el proyecto lo requiere), la aceptación del software en el entorno de operación, la realización de las actualizaciones y finalmente la instalación del software probado.

Los Procesos Integrales que articulan con el proceso de implantación son los procesos de verificación y validación, de gestión de configuración, de desarrollo de la documentación y de formación.

Los Procesos de Gestión del Proyecto relacionados al proceso de implantación son: el proceso de iniciación del proyecto, el proceso de supervisión y control del proyecto y el proceso de gestión de la calidad.

2.2 METRICA Versión 3

La metodología MÉTRICA Versión 3 [2] ofrece a las organizaciones un instrumento útil para la sistematización de las actividades que dan soporte al ciclo de vida del software. Cubre distintos tipos de desarrollo: estructurado y orientado a objetos. A través de interfaces, facilita la realización de los procesos de apoyo u organizativos: Gestión de Proyectos, Gestión de Configuración, Aseguramiento de Calidad y Seguridad.

Los procesos de la estructura principal de la metodología MÉTRICA Versión 3 son: Planificación de Sistemas de Información, Desarrollo de Sistemas de Información y Mantenimiento de Sistemas de Información.

En el marco de la metodología METRICA versión 3, se ha considerado la Fase de Implantación y Aceptación del Sistema (cuya abreviatura es IAS) como proceso que contempla los aspectos del proceso de implantación. Esta fase tiene como objetivo principal la entrega y aceptación del sistema en su totalidad, y la realización de todas las actividades necesarias para el paso a producción del mismo.

Las actividades que componen la Fase de Implantación y Aceptación del Sistema son: Establecimiento del plan de implantación, Formación necesaria para la implantación, Incorporación del sistema al entorno de operación, Carga de datos al entorno de operación, Pruebas de implantación del sistema, Pruebas de aceptación del sistema, Preparación del mantenimiento del sistema, Establecimiento del acuerdo de nivel de servicio, Presentación y aprobación del sistema y Paso a producción. Estas actividades a su vez se descomponen en tareas.

La Interfaz de Gestión de Proyectos propone tres tipos de actividades: Actividades de Inicio del Proyecto, Actividades de Seguimiento y Control y Actividades de Finalización del Proyecto, Cierre y registro de la documentación de gestión.

La Interfaz de Seguridad tiene el objetivo de incorporar en los sistemas de información, mecanismos de seguridad adicionales a los que se proponen en la propia metodología.

La Interfaz de Gestión de la Configuración consiste en la aplicación de procedimientos administrativos y técnicos durante el desarrollo del sistema de información y su posterior mantenimiento.

Las actividades propias de la Interfaz de Calidad están orientadas a verificar la calidad de los productos.

2.3 Proceso Unificado de Rational

Los autores han decidido basar el estudio comparativo en el Proceso Unificado de Rational [3] por considerarlo como un refinamiento del Proceso Unificado, aunque muchas veces en la industria suele emplearse como un mismo concepto.

El Proceso Unificado de Rational es un proceso de desarrollo de software iterativo e incremental [3].

El Proceso Unificado de Rational comprende no solamente las disciplinas de Ingeniería sino que adiciona las disciplinas de soporte. El proceso de ciclo de vida de RUP se divide en cuatro fases: Incepción, Elaboración, Construcción y Transición; a su vez estas fases se dividen en iteraciones. A través de las fases se desarrollan en paralelo nueve workflows o disciplinas: Modelado de Negocios, Requerimientos, Análisis & Diseño, Implementación, Prueba, Despliegue, Gestión de Configuración & Cambio, Gestión del Proyecto y Entorno [3].

Los aspectos relacionados al proceso de implantación son tratados en el Flujo de despliegue propuesto por el Proceso Unificado de Rational cuyo objetivo es producir con éxito distribuciones del producto y distribuirlo a los usuarios. La Fase de Transición es la cuarta fase del ciclo de vida del software, se ocupa de que el software sea puesto en manos de la comunidad de usuarios. [3].

Las actividades propuestas por el Proceso Unificado de Rational para el flujo de despliegue son: probar el software en su entorno de operación, empaquetar el software, distribuir el software, instalar el software, brindar asistencia a los usuarios, brindar capacitación a los usuarios y a los técnicos de mantenimiento, la migración del software existente y carga de datos, probar las versiones beta por los usuarios y la aceptación formal.

El flujo de Despliegue se relaciona con las disciplinas de soporte propuestas por este marco de trabajo ya que requiere de ellas insumos para su ejecución como así también le retribuye productos a las mismas.

2.4 SCRUM

Schwaber Ken y Sutherland Jeff [5] definen a SCRUM como un marco de trabajo para el desarrollo y el mantenimiento de productos complejos dentro del cual las personas pueden afrontar complejos problemas adaptativos, a la vez que entregan productos del máximo valor posible de forma productiva y creativa.

Scrum es un modelo de desarrollo ágil caracterizado por: (i) adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto, (ii) basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto-organizados, que en la calidad de los procesos empleados, (iii) solapamiento de las diferentes fases del desarrollo, en lugar de realizarlas una tras otra en un ciclo secuencial o de cascada.

Este marco de trabajo consiste en los Equipos Scrum y en sus roles (Dueño del Producto, Equipo de Desarrollo, Scrum Master y los Interesados), eventos (el Sprint, la reunión de Planificación, Scrum Diario, la Revisión de Sprint y la Retrospectiva de Sprint), artefactos (Pila de producto, la Pila de Sprint y el Incremento) y reglas asociadas [4].

Esta metodología presenta aspectos vinculados a la gestión del proyecto, los cuales se los visualiza en la planificación, en las técnicas de estimación, en las revisiones de los sprints. Los aspectos vinculados al proceso de implantación no se explicitan en esta metodología.

3 Dimensiones consideradas para el Análisis

En esta sección se plantean las dimensiones que se han considerado para el análisis del proceso de implantación de un sistema informático como así también del proceso de gestión del proyecto que articula con el mismo. Las dimensiones consideradas son fases, actividades y tareas que componen un proceso (sección 3.1), las herramientas propuestas (sección 3.2), las técnicas propuestas (sección 3.3), los artefactos (insumos y productos) (sección 3.4), los roles requeridos (sección 3.5) y las competencias necesarias para desempeñar cada uno de los roles (sección 3.6).

3.1 Fases / Actividades / Tareas

Hossian [13] denomina proceso al conjunto de acciones o actividades sistematizadas que se realizan o tienen lugar con un fin.

Pressman [14] define proceso software como una serie de pasos que incluye actividades, restricciones y recursos para producir un determinado resultado esperado.

MoProSoft [15] define actividad como el conjunto de tareas específicas asignadas para su realización a un rol o más roles. De las definiciones revisadas de proceso de software, se visualizan los términos fases, actividades y tareas, este ha sido el motivo para ser incorporado como una dimensión de análisis.

Para unificar la terminología que empleamos en nuestro análisis, se denominará a la dimensión "Fases/Actividades/Tareas".

3.2 Herramientas propuestas

El Diccionario de la Lengua Española [16] define el término "herramienta" como "...Instrumento, por lo común de hierro o acero, con que trabajan los artesanos...".

Si bien esta definición no refleja totalmente a que denominamos herramienta en el contexto de la Ingeniería de Software, podemos considerar de ella dos componentes, instrumento y que se utiliza para trabajar. En la construcción del producto software se requiere de herramientas, se pueden mencionar las herramientas de modelado, las cuales permiten la creación de los modelos necesarios del sistema que se está estudiando. Con la evolución de la Ingeniería de Software, se ha introducido un amplio abanico de diferentes tipos de programas que se utilizan para ayudar a las actividades del proceso de software, como el análisis de requerimientos, el modelado de sistemas, la depuración y las pruebas, a este grupo de herramientas se las denomina herramientas CASE (Ingeniería de Software Asistida por Computadora) [17].

Los autores consideran para esta dimensión de análisis tanto a las herramientas manuales y automatizadas que se utilizan en el proceso de implantación y en el proceso de gestión del proyecto de software.

3.3 Técnicas Propuestas

Los autores han considerado la definición de técnica propuesta en la metodología METRICA versión 3 [2] como el conjunto de procedimientos que se basan en reglas y notaciones específicas en términos de sintaxis, semántica y gráficos, orientadas a la obtención de productos en el desarrollo de un sistema de información. Se considera necesaria la aplicación de técnicas en la realización de las actividades de un proceso con el propósito de lograr una estandarización de las mismas dentro del proceso analizado.

3.4 Artefactos (Insumos / Productos)

En Acuña et al. [8] plantean que la definición de un proceso software debe especificar no sólo las actividades, sino también las técnicas para realizar las tareas, los actores que ejecutan las actividades, sus roles y los artefactos producidos.

MoProSoft [15] define producto como cualquier elemento que se genera en un proceso.

Los autores consideran al término artefacto o producto a cualquier tipo de información creada, cambiada, utilizada por los roles que participan en el desarrollo del software. Esta dimensión de análisis considera los artefactos productos y los artefactos insumos ya que son necesarios como inputs para la realización de las actividades del proceso.

3.5 Roles Requeridos

MoProSoft [15] define el concepto de rol como responsable por un conjunto de actividades de uno o más procesos. Un rol puede ser asumido por una o más personas de tiempo parcial o completo.

Los autores consideran de importancia la definición de roles para la realización de actividades dentro de un proceso de software así se conoce de antemano las responsabilidades, derechos, aptitudes y habilidades necesarias para el rol requerido.

3.6 Competencias

Desde hace décadas que se intenta resolver el problema de gestionar adecuadamente el capital humano en el desarrollo de software. Boehm [18] postula explícitamente que las relaciones humanas dentro de los proyectos de software son un componente clave (junto con los recursos adecuados y la ingeniería de software) para la consecución de un producto de software con éxito y la realización de un proceso de desarrollo y mantenimiento de software exitoso.

DeMarco y Lister [19], Constantine [20] consideran que los principales problemas de desarrollo de software son humanos y no técnicos.

Por la revisión realizada respecto a los aspectos humanos en el proceso de desarrollo de software, se visualiza que aún es un variable no resuelta en su totalidad, por este motivo es que los autores adhieren a considerar las competencias como una dimensión de análisis.

Tobón [22] plantea que una competencia es una actuación integral para identificar, interpretar, argumentar y resolver problemas del contexto con idoneidad y compromiso ético, articulando el saber ser, el saber hacer y el saber conocer.

En Acuña et al. [8] plantean una clasificación de competencias en función de cómo se movilizan los recursos. Las competencias duras se centran en los aspectos técnicos que se requieren para llevar a cabo una actividad, generalmente se expresan en términos de conocimientos, destrezas y habilidades. Las competencias blandas se centran en los comportamientos, en los rasgos personales entre los cuales se pueden mencionar el liderazgo, la colaboración con los demás, la integridad y la persuasión.

La dimensión competencias contemplará las competencias duras y las competencias blandas.

4 Resultados e Interpretación

En esta sección se plantea una tabla comparativa de las dimensiones de análisis (sección 4.1) y por último la interpretación de los resultados obtenidos (sección 4.2).

4.1. Tabla Comparativa de Dimensiones

En la Tabla 1, se presenta el grado de cumplimiento de las dimensiones de análisis consideradas para el proceso de implantación dentro del proceso de desarrollo de software (ver en la Tabla, Implantación) y para los procesos de gestión de proyectos de software (ver en la Tabla, Gestión de Proyectos).

En esta comunicación, los autores se focalizan en el proceso de implantación como proceso dentro del proceso de desarrollo de software. Ha sido necesario considerar

los aspectos de gestión del proyecto software por las interacciones que se presentan con el proceso de implantación.

Tabla 1. Tabla comparativa de las dimensiones de análisis.

	Metodologías/Procesos							
	IEEE 1074		METRICA Versión 3		Proceso Unificado de Rational		SCRUM	
	Implan-tación	Gestión del Proyecto	Implan-tación	Gestión del Proyecto	Implan-tación	Gestión del Proyecto	Implan-tación	Gestión del Proyecto
Fases/ Actividades/ Tareas	■	■	■	■	■	■		■
Herramientas Propuestas					■	■		■
Técnicas Propuestas			■	■	■			■
Artefactos (Insumos/ Productos)	■	■	■	■	■	■		■
Roles Requeridos			■	■	■	■		■
Competencias								

Si bien en la tabla precedente, se presenta una unificación en la terminología para ambos procesos analizados como así también para las dimensiones de análisis consideradas surge la necesidad de describir este proceso de unificación de terminología.

En el estándar de la IEEE 1074 [1] el proceso de implantación se lo denomina proceso de instalación y el proceso de gestión del proyecto queda resuelto por los procesos de gestión del proyecto y los procesos integrales. El estándar plantea la información que ingresa y la información resultante de los procesos que los componen, en la tabla denominaremos a esta información, artefactos (insumos/productos).

En la metodología Métrica versión 3 [2] la denominación equivalente para el proceso de implantación es Fase de Implantación y Aceptación del usuario. El equivalente al proceso de gestión del proyecto se encuentra resuelto en las interfaces de Gestión de Proyectos. Esta metodología presente el concepto de roles con el término participantes.

En el Proceso Unificado de Rational [3] la denominación equivalente para el proceso de implantación se presenta en el flujo de despliegue y la fase de transición. Respecto al proceso de gestión del proyecto se encuentra resuelto en el Flujo de Gestión del proyecto. El Proceso Unificado de Rational presenta el concepto de trabajador, en nuestro análisis será considerado en la dimensión Roles Requeridos.

En la metodología SCRUM [4][5], no se encuentra un proceso equivalente al de implantación. El proceso de gestión del proyecto está resuelto con el mismo nombre.

4.2. Interpretación de Resultados

Los resultados de análisis a los cuales se ha arribado con la Tabla Comparativa de Dimensiones (sección 4.1.) permiten formular las siguientes conclusiones parciales:

- En relación a la dimensión Fases/Actividades/Tareas, el estándar IEEE 1074 [1], la metodología Métrica versión 3 [2] y el Proceso Unificado de Rational [3] proponen las fases, actividades y tareas tanto para ambos procesos. En la metodología SCRUM [4][5] se definen las actividades para el proceso de gestión del proyecto y se evidencia la ausencia de actividades para el proceso de implantación.
- En la relación a la dimensión Herramientas propuestas, en el estándar IEEE 1074 [1] y en la metodología Métrica versión 3 [2] se denota la ausencia de la propuesta de herramientas para las actividades de ambos procesos. En el Proceso Unificado de Rational [3] se visualiza la propuesta de herramientas para ambos procesos. En la metodología SCRUM [5][6] se proponen herramientas para el proceso de gestión del proyecto de software.
- En relación a la dimensión Técnicas Propuestas, en el estándar IEEE 1074 [1] se visualiza la ausencia para ambos procesos. En la metodología Métrica versión 3 [2] se proponen técnicas para ambos procesos. En el Proceso Unificado de Rational [3] se proponen técnicas para el proceso de implantación. En la metodología SCRUM [5][6] se proponen técnicas para el proceso de gestión del proyecto de software.
- Respecto a la dimensión Artefactos (Insumos/Productos), en el estándar IEEE 1074 [1], en la metodología Métrica versión 3 [2] y en el Proceso Unificado de Rational [3] se identifica la propuesta de artefactos para ambos procesos. En la metodología SCRUM [5][6] se denota la propuesta de artefactos para el proceso de gestión del proyecto de software únicamente.
- En relación a la dimensión Roles Requeridos, en el estándar IEEE 1074 [1] no se proponen roles para ninguno de los procesos estudiados. La metodología Métrica versión 3 [2] y el Proceso Unificado de Rational [3] presentan los roles para ambos procesos. La metodología SCRUM [5] [6] solo propone roles para el proceso de gestión del proyecto de software.
- Respecto a la dimensión Competencias, ninguna de las metodologías consideradas enuncian competencias para los procesos.

5 Conclusiones y Futuras Líneas de Trabajo

Se ha presentado una revisión sistemática del subproceso de implantación de sistemas mediante un estudio comparativo de las metodologías seleccionadas, que ha permitido identificar un conjunto de vacancias del subproceso, en particular se destaca la carencia vinculada a los aspectos humanos; aun existiendo modelos que proponen roles, no se presentan las competencias con las que deben contar las personas para desempeñar dichos roles. Se ha logrado la construcción de un instrumento comparativo que permite estudiar una serie de dimensiones de análisis del proceso de implantación y del proceso de gestión del proyecto con el que interactúa.

Como trabajo futuro se identifica: (a) la necesidad de incorporar más modelos de procesos o metodologías, como por ejemplo: Programación Extrema (XP), Método de desarrollo de sistemas dinámico (DSDM), Desarrollo Rápido de Aplicaciones (RAD),

Proceso Unificado Ágil (AUP); y (b) explorar la incorporación de otras dimensiones en el instrumento comparativo, como por ejemplo, las métricas.

Referencias

1. IEEE 1074, 1997. IEEE Standard for Developing Software Life Cycle Processes. IEEE Std 1074 (Revision of IEEE Std 1074-1995; Replaces IEEE Std 1074.1-1995) (1997).
2. PAe, Métrica versión.3. Portal de Administración Electrónica. Gobierno de España.
3. IBM. Rational Software. Péraire C., Edwards M, Fernandes A., Mancin E. y Carroll K. The IBM Rational Unified Process for Systems (2007).
4. Scrum Manager. Palacio J. Gestión de Proyectos SCRUM Manager (Scrum Manager I y II). <http://www.scrummanager.net>. Página vigente al 17/06/2016 (2015).
5. Schwaber K. y Sutherland J. La Guía Definitiva de Scrum: Las reglas del juego. (2013)
6. F. M. de Vasconcelos Jr. and C. M. L. Werner, "Software development process reuse based on patterns". Proceedings of the Ninth International Conference on Software Engineering and Knowledge Engineering (June 1997) pp. 97- 104 (1997).
7. Acuña S., Juristo. N. Moreno A., Mon A. A software Process Model Handbook for incorporating people's capabilities. United States of America: Springer Science+Business Media, Inc. (2005).
8. Report, The Chaos Manifesto: Think Big, Act Small. Stash Group. (2013).
9. IEEE 830. IEEE Recommended Practice for Software Requirements Specifications - IEEE Std 830-1998. (1998).
10. ISO/IEC/IEEE 29148. Software & Systems Engineering Standards Committee of the IEEE Computer Society, ISO/IEC/IEEE 29148:2011(E), Systems and software engineering — Life cycle processes — Requirements engineering (2011).
11. LEITE JCSP. Engenharia de Requisitos, Notas Tutoriales, Teaching material of the Requirements Engineering course, Computer Science Department of PUC-Rio, Brazil (1994).
12. Piattini Velthius, M., Calvo-Manzano, J., Cervera J. y Fernández, L. Análisis y Diseño de Aplicaciones de Gestión: Una perspectiva de Ingeniería del Software. Alfaomega Grupo Editor (2004).
13. Hossian, A. Modelo de Proceso de Conceptualización de requisitos. Tesis Doctoral en Ciencias Informáticas. Facultad de Informática. Universidad Nacional de La Plata (2012).
14. Pressman R. Ingeniería del software: Un enfoque práctico. Mc Graw Hill. 7ma Edición (2010).
15. MoProSoft. Modelo de procesos para la industria del software. Secretaría de Economía México. Versión 1.3 (2005).
16. RAE. Diccionario de la Lengua Española. Real Academia Española. <http://dle.rae.es/> Página Vigente al 27/06/2016.
17. Sommerville I.. Ingeniería del software. Addison-Wesley. 7ma. Edición (2005).
18. Boehm, Barry. Software Engineering Economics. Prentice Hall, (1981).
19. DeMarco T., Lister T. Peopleware: Productive Projects and Teams.3era. Edición. Addison-Wesley Professional (2013).
20. Constantine L.. "The Peopleware Papers: Notes on the Human Side of Software" (2001).
21. Tobón, S. Formación basada en competencias. 2nd. Ed. Bogotá: ECOE Ediciones (2005).

Propuesta de Artefactos para el Subproceso de Gestión del Modelo de Proceso de Proyectos de Explotación de Información (G-MoProPEI)

Sebastian Martins^{1,2}, Patricia Pesado³, Ramón García-Martínez²

¹ Programa de Doctorado en Ciencias Informáticas. Facultad de Informática. Universidad Nacional de La Plata

² Laboratorio de Investigación y Desarrollo en Ingeniería de Explotación de Información. Grupo de Investigación en Sistemas de Información. Universidad Nacional de Lanús

³ Instituto de Investigaciones en Informática LIDI. Facultad de Informática. UNLP - CIC smartins089@gmail.com, ppesado@lidi.info.unlp.edu.ar, rgm1960@yahoo.com

Resumen. Se ha señalado que los modelos de procesos existentes presentan carencias al momento de abordar un proyecto de explotación de información. Como respuesta a dicha necesidad, se definió un modelo de proceso (MoProPEI) el cual permita sistematizar el desarrollo de un proyecto, considerando aquellos aspectos de gestión y control requeridos para esta disciplina. En este trabajo se presenta una visión detallada del subproceso de gestión con detalle del conjunto de formalismos utilizados para representar artefactos de entradas y salidas para cada una de las actividades. Se ilustran en una prueba de concepto perteneciente al área de educación.

Palabras Clave. Ingeniería de Explotación de Información. Modelo de proceso. Gestión de Proyectos. Artefactos de Gestión. Minería de Datos.

1 Introducción

El termino minería de datos está fuertemente ligado al concepto de base de datos y se remonta a la definición de algoritmos de búsqueda de patrones de conocimiento en grandes bases de datos [1]. Sin embargo, hoy existen líneas de investigación en campos tales como: minería de textos [2], minería de imágenes [3], minería de patrones en flujos de información [4], minería en la web [5], entre otras. En este contexto, en la línea de investigación en la que se enmarca este artículo, se conviene utilizar el término “explotación de información” como referencia genérica a cualquiera de los tipos de minería precitados. Con base en que la Ingeniería de Software ha sido definida en el SWEBOK [6] como: “la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, y el estudio de estos enfoques, es decir, la aplicación de la ingeniería al software”; se conviene en definir a la Ingeniería de Explotación de Información [7] como la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo de proyectos de explotación de información, y el estudio de este enfoque, es decir, la aplicación de la ingeniería a la explotación de información.

Se ha señalado [8] que los modelos de procesos existentes presentan carencias al momento de abordar un proyecto de explotación de información. Por ello, se definió un modelo de proceso (MoProPEI) que mediante la visión de procesos (fases, tareas, técnicas de representación y procedimientos de ejecución de la tarea) permita sistematizar del desarrollo de un proyecto, considerando aquellos aspectos de gestión y control requeridos en todo proceso productivo.

En [9], se ha realizado una comparación con los principales modelos de procesos utilizados en la industria, centrándose en los aspectos de gestión, resaltando las ventajas introducidas por el proceso propuesto.

En este trabajo se presenta una descripción de las problemáticas abordadas (sección 2), se presenta una visión detallada del subproceso de gestión, presentando el conjunto de formalismos utilizados junto con las dependencias funcionales de cada actividad (sección 3), las cuales se ilustran en una prueba de concepto perteneciente al área de educación (sección 4). En la sección 5, se presentan las conclusiones y futuras líneas de trabajo.

2 Descripción del Problema

Los modelos de proceso actuales se centran en las tareas de desarrollo (aquellas vinculadas con el set de datos y los algoritmos) dejando de lado otros aspectos relevantes en el desarrollo de un proyecto, como la planificación, seguimiento y control de las actividades, la visión transversal del mismo, entre otros.

De forma complementaria, los modelos existentes brindan una visión global del proceso, pero carecen de una descripción detallada que permita comprender las vinculaciones entre las actividades y las posibles herramientas/técnicas, complejizando el desarrollo del proceso, realizándolo de manera desorganizada dificultando la trazabilidad, evaluación y reutilización de los proyectos.

Hace ya más de 10 años, se ha señalado la alta tasa de fracasos en proyectos de explotación de información (cifra superior al 60%) [10] y distintas dificultades en el desarrollo de los mismos [11-13], siendo CRISP-DM la principal metodología empleada para el desarrollo de proyectos desde dicha fecha hasta la actualidad [14]. Esto puede deberse a la carencia de artefactos propuestos por los modelos de procesos existentes, debiendo los ingenieros de explotación de información realizar de manera artesanal o poco sistematizada el desarrollo de los proyectos, contando a su vez, con un modelo guía el cual está centralizado en las actividades de desarrollo, omitiendo aspectos vinculados con la gestión del proyecto, en una disciplina cuyos proyectos han ido incrementando en magnitud y complejidad, así como en la conformación de sus grupos interdisciplinarios de trabajo.

3 Subproceso de Gestión y Control en MoProPEI

El proceso propuesto está conformado por 2 subprocesos: Desarrollo, vinculado con las actividades de entendimiento y preparación de los datos y su posterior extracción de conocimiento; y Gestión, orientado al control y la administración del proyecto.

El subproceso de Desarrollo se encarga de todas las tareas asociadas a la obtención de requisitos, la comprensión del negocio y de los problemas del mismo, la identificación de recursos relevantes para el desarrollo del proyecto, particularmente de las fuentes de datos, el análisis, comprensión y preparación de los datos existentes en la organización, la identificación y selección de los procesos, técnicas y herramientas a utilizar, la implementación y su posterior evaluación y producción de los resultados obtenidos, cuyo orden se define con el objetivo de reducir la cantidad de iteraciones entre las etapas del subproceso, y favorecer la ejecución del mismo, logrando una mejor articulación entre las actividades involucradas.

El subproceso de gestión se concibe de forma transversal a las actividades de desarrollo, cuya ejecución de sus tareas no es de forma lineal, sino que se realizan en base al progreso del proyecto. El mismo abarca la administración del proyecto, comprender la situación del cliente, identificar, planificar y controlar los recursos, identificar el modelo de ciclo de vida, controlar la ejecución de las actividades, realizar las mediciones, definir la viabilidad del proyecto, y formalizar el cierre del mismo. El subproceso de Gestión, está conformado por cuatro fases, cada una de las cuales se componen de distintas tareas generales, que identifican un conjunto de actividades con un objetivo específico dentro del proyecto.

Este trabajo presenta las actividades que componen cada fase, indicando sus documentos de entrada y salida, pudiendo identificar las dependencias que ellos poseen.

La fase de **Iniciación** (figura 1), está compuesta por cuatro actividades: *definición de la comunicación*, tiene como elemento de entrada el discurso del cliente, el discurso del líder del proyecto, y las políticas de las organizaciones intervinientes definiendo a partir de ellos el protocolo del proyecto; *exploración de conceptos iniciales*, cuyos inputs son el output de la actividad previa y el discurso del cliente, produciendo como resultados parciales el reporte de recursos del cliente, el plan de adquisición de conocimiento y el conocimiento adquirido; *evaluación de la situación*, cuyos elementos de entrada están conformados por todos los elementos de salida de la actividad previa, y produce los reportes de recursos internos, posibilidad de tercerización, viabilidad, y se identifican proyectos similares que sirvan de guía para el desarrollo del proyecto y riesgos y contingencias del mismo; y *definición del ciclo de vida* que establece el patrón estructural mediante el cual el proyecto será ejecutado (ciclo de vida) a partir de las características del proyecto definidas en la actividad previa.

La fase **Planificación** (figura 2), la cual se compone de tres actividades generales: *planificación de las actividades*, cuyos elementos de entrada son los proyectos guías identificados, el ciclo de vida y el documento de requisitos, definido en el proceso de desarrollo y se acuerda el mapa y calendario de actividades; *planificación de recursos*, que a partir de los proyectos guías, el calendario de actividades y el documento de requisitos, define el plan de tercerización y el reporte de recursos requeridos; y *estimaciones y responsabilidades*, en donde a partir de los elementos de

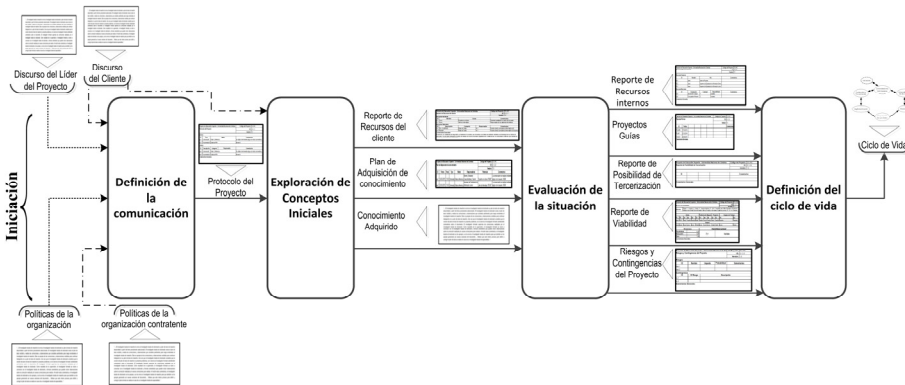


Fig. 1. Fase Inicial

salida de la actividad previa, junto con los proyectos guías, el mapa de actividades y el documento de requisitos, se realiza la estimación de costo y se definen los alcances del proyecto y las obligaciones que las partes intervinientes acuerdan (contrato del proyecto).

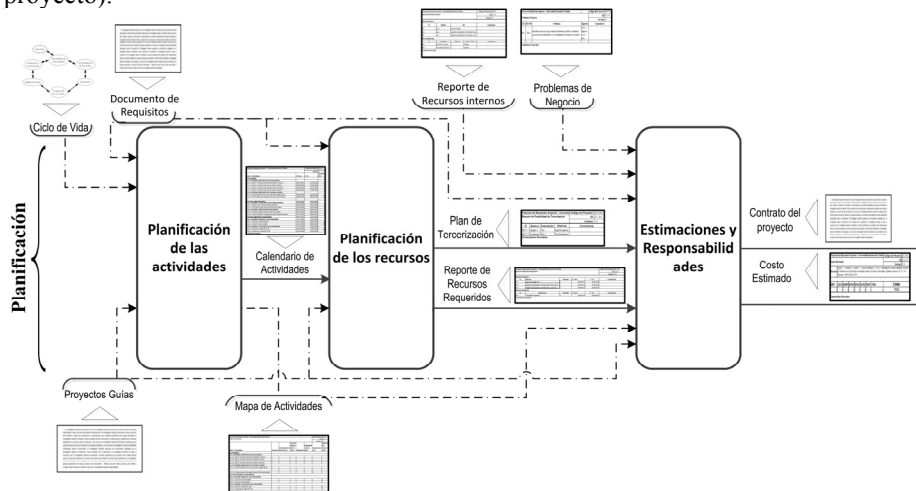


Fig. 2. Subproceso Gestión: Fase Planificación

La fase **Soporte** (figura 3), está conformada por 3 actividades: *gestión del ciclo de vida*, que a partir del calendario de actividades y el modelo de ciclo de vida escogido, se determinan los alcances del ciclo de vida y los elementos pendientes de realizar para próximas iteraciones del proyecto (en caso que hubiese) definidos en los reportes formales de inicio y fin de ciclo, los cuales son utilizados de manera global en el desarrollo del proyecto; *gestión del desarrollo*, en donde se define las responsabilidades de los recursos en el proyecto, teniendo como elementos de ingreso el calendario de actividades, el plan de tercerización (en caso que hubiese), los riesgos y contingencias del proyecto y del problema de negocio y los reportes de recursos requeridos y del progreso de las actividades, generando como productos de salida los

contratos, el reporte de asignación de responsabilidades y la comunicación del progreso del proyecto, utilizados de manera global en el desarrollo del proyecto; y *gestión de la configuración*, que a partir de los avances en el proyecto, se producen ajustes al documento del proyecto y se establece el manejo de versionado del mismo.

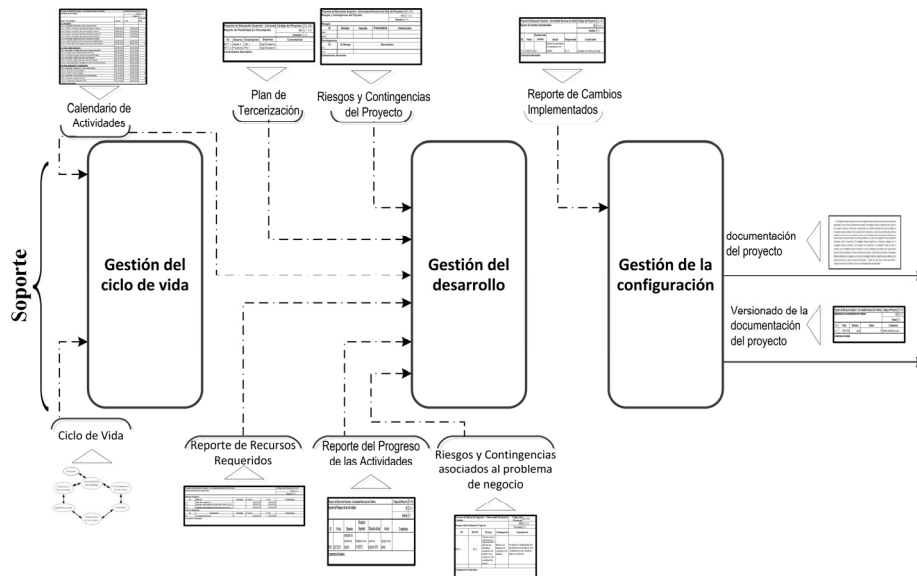


Fig. 3. Subproceso Gestión: Fase Soporte

La fase **Control y Calidad** (figura 4), integrada por 4 actividades: *control de los recursos*, cuyos elementos de entrada son los contratos de recursos y tercerización, el reporte de recursos requeridos y los problemas y objetivos de negocio, generando los controles correspondientes al incorporación de recursos y tercerización de tareas; *mediciones del proyecto*, en donde a partir del listado de métricas y los costos de las actividades del proyecto, se calculan y controlan las variables de interés para el proyecto, generando los reporte de métricas y de costos; *control de las actividades*, en donde se realizan los controles generales del progreso de las actividades, riesgos y calidad del proyecto, a partir de las salidas de la actividad previa junto con los costos y esfuerzo estimado, los riesgos y contingencias identificados, el calendario de actividades y el reporte de responsabilidades del personal, brindando un control detallado del progreso y posibles desvíos en el desarrollo del plan de proyecto; y *gestión del cambio*, donde se realiza la evaluación, implementación y control de los cambios solicitados a través del desarrollo del proyecto dejando constancia de los mismos, sus elementos de entrada son el documento de requisitos, los problemas de negocio y sus riesgos asociados y las solicitudes de cambio, produciendo los reportes de cambios implementados y de control de la integración del cambio.

La fase de **entrega** (figura 5), compuesta por 2 actividades: *formalización externa del cierre del proyecto*, que se realiza una verificación y validación formal con el objetivo de constatar que las necesidades del cliente fueron satisfechas y que las obligaciones tercerizadas fueron correctamente cumplimentadas, generado a partir del discurso del

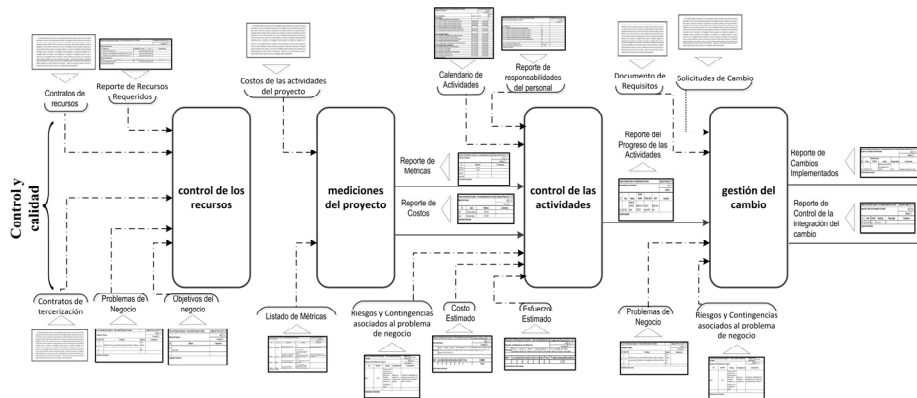


Fig. 4. Subproceso Gestión: Fase Control y Calidad

cliente, el contrato del proyecto, los contratos de recursos y tercerización y la planilla de criterios de éxito del problema de negocio, el documento de aceptación y el reporte de conclusión de contrataciones; *formalización interna del cierre del proyecto*, en donde se generan una serie de reportes para el control y mejora propia del equipo de trabajo, según la experiencia adquirida a lo largo del proyecto. Sus elementos de entrada son los generados en la actividad previa junto con los reportes de calidad del proyecto, riesgos acontecidos (en caso que hubiese), calidad del proyecto, métricas y costos, generando el reporte de sugerencia de mejoras y el documento interno de desarrollo del proyecto.

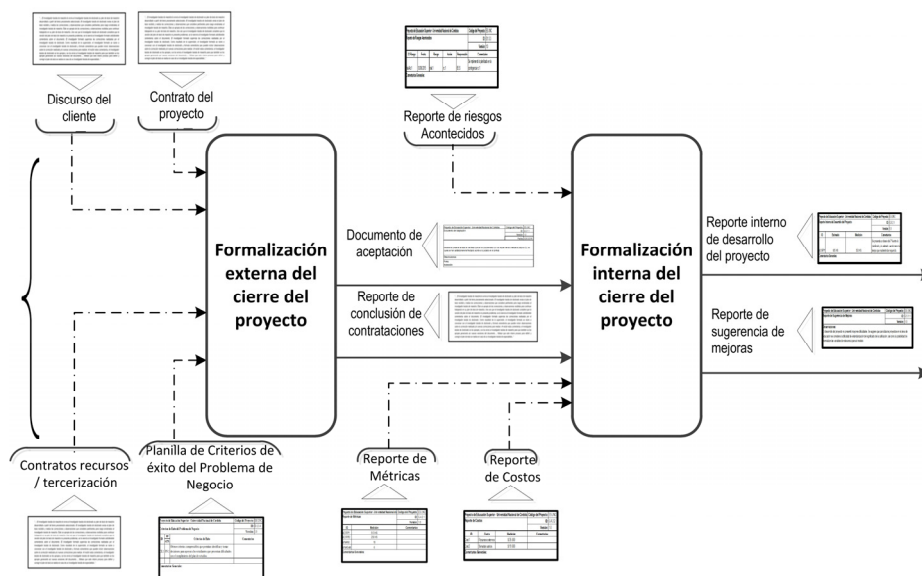


Fig. 5. Subproceso Gestión: Fase Entrega

4 Prueba de Concepto

En esta sección se presenta una prueba de concepto en la cual se introducen algunos formalismos de mayor relevancia obtenidos a partir de la aplicación de MoProPEI en el proyecto [15]. El mismo se encuentra enmarcado en el ámbito de la educación, y sus requerimientos generales obtenidos a partir del cliente pueden ser resumidos en:

“El propósito de esta investigación es contribuir a facilitar la apropiación del conocimiento en Educación Superior en contextos de masividad.

Proveer de información para un adecuado diseño de las políticas públicas en Educación Superior, despierta el interés para contribuir con una mejor apropiación del conocimiento por parte de la sociedad. En esta dirección, una dimensión relevante es la asociada a las características del estudiante, principal actor de este complejo escenario.

Se espera que esta contribución resulte novedosa, tanto en las interpretaciones sobre la información del estudiante que emerjan, como en la construcción del diseño metodológico aplicable en las muy diversas prácticas de este escenario.

La Universidad Nacional de Córdoba dispone de 2 sistemas fundamentalmente en los cuales se resguarda la información del estudiante. El Sistema de Información Universitaria SIU_GUARANI, de gestión académica, el cual contiene información académica y socioeconómica de los estudiantes, y las aulas virtuales desarrolladas sobre la plataforma MOODLE las cuales contienen información de las instancias de acreditación.

Se espera lograr un mejor conocimiento de las características del estudiante, protagonista principal de este escenario, permitiendo proporcionar contribuciones novedosas y valiosas que favorezcan la toma de decisiones en aspectos vinculados con la gestión de la Educación Superior en contextos de masividad.”

De los requerimientos previamente educidos, se identifican objetivos y problemas de negocio a partir de los cuales se orienta el desarrollo de la solución. Entre los elementos generados durante el desarrollo de proyecto, se ilustran de la fase de iniciación: el *Reporte de Recursos del Cliente* (tabla 1), generado en la actividad Exploración de Conceptos Iniciales, en el cual se identifican los recursos del cliente de interés para el desarrollo del proyecto; los *Riesgos y Contingencias del Proyecto* (tabla 2), producido en la actividad evaluación de la situación, identificando aquellos eventos que pueden ocurrir afectando el desarrollo del proyecto y las posibles acciones a realizar; y el *Reporte de Viabilidad* (tabla 3), generado en la actividad evaluación de la situación, determinando la posibilidad de realización con éxito del proyecto (haciendo uso del modelo definido en [16]); y de la fase de planificación: el *Listado de Métricas* (tabla 4), basadas en [17], generado en la actividad de planificación de las actividades, identificando aquellas variables de interés a medir a lo largo del proyecto.

5 Conclusiones

Los autores vienen desarrollando en los últimos años una línea de trabajo en el área de control y gestión para proyectos de explotación de información. Se han propuesto dos subprocesos: de desarrollo que abarca aquellas actividades asociadas con el entendimiento del negocio y de los datos, su preparación, las técnicas de explotación de información y el reporte de los resultados, y otro Gestión enfocado en el control y la administración del proyecto.

Tabla 1. Reporte de Recursos del cliente

Proyecto de Educación Superior - Universidad Nacional de Córdoba				Código del Proyecto:	ES.UNC
Reporte de Recursos del cliente				ID:	G.1.2.1
				Versión:	1.0
Recursos Humanos					
ID	Nombre	Cargo		Comentarios	
rh.1	xxxx	Experto en el área "XXXX"		Docente investigador a cargo del proyecto	
rh.1	xxxx	Jefe de Informática "XXXX"		Responsable de los sistemas informáticos	
Recursos Materiales					
ID	Descripción	Categoría	Responsable	Comentarios	
rm.1	SIU_GUARANI	Base de Datos	rh.1	Fuente donde se registra la información del desempeño del alumno en el transcurso de su carrera	
rm.2	Moodle	Base de Datos	rh.1	Fuente donde se registra la información del alumno respecto de la cursada de una materia específica	
Comentarios Generales:					
Se acordó, por cuestiones de seguridad y privacidad en los datos, que el acceso a los datos se realizaría mediante una descarga de los mismos en texto plano, la cual sería entregada al equipo de trabajo con todos los campos registrados en el sistema que no contuviesen contenido que violase las normas de privacidad de los estudiantes.					

NOTA: Se resalta que por cuestiones de privacidad, el nombre del personal cliente fue omitido.

Tabla 2. Reporte de Riesgos y Contingencias del Proyecto

Proyecto de Educación Superior - Universidad Nacional de Córdoba				Código del Proyecto:	ES.UNC
Riesgos y Contingencias del Proyecto				ID:	G.1.3.3
				Versión:	1.0
Riesgos					
ID	Nombre	Impacto	Probabilidad	Comentarios	
risk.1	Demora en acceso a los datos	3	70%	Por cuestiones de políticas de aseguramiento de la privacidad de los datos personales de los estudiantes	
Contingencias					
ID	ID Riesgo	Descripción			
rc.1	risk.1	Se ajustarán los plazos del proyecto acorde a las demoras			
Comentarios Generales:					

Tabla 3. Reporte de Viabilidad

Proyecto de Educación Superior - Universidad Nacional de Córdoba										Código del Proyecto:	ES.UNC	
Reporte de Viabilidad										ID:	1.1.3.7	
										Versión:	1.0	
Técnica	Pytel, P., Hossian, A., Britos, P., García-Martínez, R. 2015. Feasibility and Effort Estimation Models for Medium and Small Size Information Mining Projects. Information Systems Journal, 47: 01-14. Elsevier. ISSN 0306-4379.											
Datos					Problema de Negocio			Proyecto		Equipo de Trabajo		
P1	P2	A1	A2	A3	E1	P3	A4	A5	E2	E3	P4	E4
todo	todo	todo	mucho	mucho	todo	mucho	poco	mucho	todo	mucho	mucho	regular
Umbral												
poco	poco	poco	poco	poco	nada	poco	poco	poco	nada	nada	poco	nada
Dimensiones					Viabilidad global							
Plausibilidad					8,24			7,1		Valido		
Adecuación					5,66							
Éxito					7,49							
Comentarios Generales:												

Tabla 4. Listado de Métricas

Proyecto de Educación Superior - Universidad Nacional de Córdoba			Código del Proyecto:	ES.UNC
Listado de Métricas			ID:	G.2.1.2
			Versión:	1.0
ID	Tipo	Descripción	Fórmula	Comentarios
M.DRPY	Proyecto	Tiempo total requerido para el desarrollo del proyecto	$\sum trA$ trA = tiempo requerido por actividad	Sumatoria de los tiempos requeridos para cada actividad del proyecto
M.DRPEI	Proyecto	Tiempo medio requerido para el desarrollo de un problema de explotación de información	$(\sum trA.SD) / NPEI$ trA.SD = tiempo requerido por actividad del subproceso de desarrollo NPEI = cantidad de problemas de explotación de información	Solo se considera el tiempo de las actividades pertenecientes al subproceso de desarrollo
M.NANU	Datos	Número total de atributos que no son de utilidad en las tablas	$\sum aNu$ aNu = cantidad de atributos que no son de utilidad	
M.NASxM	Modelo	Número medio de atributos significativos por modelo	$(\sum AtS.M) / NMOD$ AtS.M = cantidad de atributos significativos de un modelo NMOD = cantidad de modelos	
Comentarios Generales:				

Actualmente se está trabajando en los formalismos que encapsulan los artefactos de entrada y salida de cada una de las actividades y la implementación de técnicas, que asociadas a cada una de las actividades permiten transformar el artefacto de entrada en los correspondientes artefactos de salidas. En este trabajo se han desarrollado formalismos para el subproceso de gestión y particularmente se han detallado en cada una de las fases del proyecto cuales son los artefactos de entrada y salida. El próximo paso es desarrollar las técnicas para la transformación de artefactos en el subproceso de gestión y seguidamente elaborar los artefactos y las técnicas para el subproceso de desarrollo de MoProPEI.

Como futuras líneas de trabajo, se prevé ampliar los casos de aplicación del modelo en cantidad y variedad de dominios.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por beca PROMINF-UNLa-2015-2017 del Ministerio de Educación Argentina y por el Proyecto 33A205 de la Secretaria de Ciencia y Tecnología de la Universidad Nacional de Lanús.

Referencias

1. Maimon, O. y Rokach, L. (Eds.). 2005. *Data mining and knowledge discovery handbook*. Springer.
2. Tan, A. 1999. Text mining: The state of the art and the challenges. In *Proc. PAKDD 1999 Workshop on Knowledge Discovery from Advanced Databases*. pp. 65-70.
3. Hsu, W., Lee, M., Zhang, J. 2002. Image mining: Trends and developments. *Journal of Intelligent Information Systems*, 19(1): 7-23.
4. Gaber, M., Zaslavsky, A. Krishnaswamy, S. 2010. *Data stream mining*. En Maimon, O. and Rokach, L. eds. *Data mining and knowledge discovery handbook*. Springer, Pág. 759-787.
5. Kosala, R., Blockeel, H. 2000. Web mining research: A survey. *ACM SIGKDD Explorations Newsletter*, 2(1): 1-15.
6. Abran, A., Moore, J. W., Bourque, P., Dupuis, R., Tripp, L. 2004. *Guide to the Software Engineering Body of Knowledge (2004 version)*. IEEE. ISBN 0-7695-2330-7.
7. García-Martínez, R., Britos, P., Pesado, P., Bertone, R., Pollo-Cattaneo, F., Rodríguez, D., Pytel, P., Vanrell, J. 2011. *Towards an Information Mining Engineering*. En *Software Engineering, Methods, Modeling and Teaching*. Sello Editorial Universidad de Medellín. ISBN 978-958-8692-32-6. Pág. 83-99.
8. Martins, S., Pesado, P., García-Martínez, R. 2014. Propuesta de Proceso de Ingeniería de Explotación de Información Centrado en Control y Gestión del Proyecto. XI Workshop de Bases de Datos y Minería de Datos. *Proceedings XX Congreso Argentino de Ciencias de la Computación*. Universidad Nacional de la Matanza. ISBN 978-987-3806-05-6.
9. Martins, S., Pesado, P., García-Martínez, P. 2016. Information Mining Projects Management Process. *Proceedings 28th International Conference on Software Engineering & Knowledge Engineering*. Pág. 504-509. ISBN 1-891706-39-X.
10. Gondar, J.E. 2005. *Data Mining Methodology*. Data Mining Institute. ISBN: 978-84-96272-21-7.
11. Wirth R., Hipp J. 2000. CRISP-DM: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, Manchester, UK, pp. 29–39.
12. Yang Q., Wu X., 2006. 10 Challenging Problems in Data Mining research, *International Journal of Information Technology and Decision Making* 5(4), pp. 597–604.
13. Lavrac N., Motoda H., Fawcett T., Holte R., Langley P., Adriaans P., 2004. Lessons Learned from Data Mining Applications and Collaborative Problem Solving. *Machine Learning*. 57. 13–34.
14. Kdnuggets. 2014. What main methodology are you using for your analytics, data mining, or data science projects? Poll (Oct 2014). <http://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-methodology.html> (Último acceso 04/07/2016).
15. Martins, S. 2016. Artefactos para el Subproceso de Gestión del Proceso MoProPEI. Reporte de Tareas RT-UNLa-DDPyT-GISI-2016-01. <http://sistemas.unla.edu.ar/sistemas/gisi/papers/RT-UNLa-DDPyT-GISI-2016-01.pdf>
16. Pytel, P., Hossian, A., Britos, P., García-Martínez, R. 2015. Feasibility and Effort Estimation Models for Medium and Small Size Information Mining Projects. *Information Systems Journal*, 47: 01-14. Elsevier. ISSN 0306-4379.
17. Basso, B., Rodríguez, D., García-Martínez, R. 2015. Comportamiento de Métricas para Proyectos de Explotación de Información en PyMEs. XII Workshop de Ingeniería de Software. Libro de Actas del XXI Congreso Argentino de Ciencias de la Computación. Pág. 485-494. ISBN 978-987-3724-37-4. Universidad Nacional del Noroeste de Buenos Aires.

Proposal for Optimizing the Design of the National University of La Plata Service Cloud

José Nahuel Cuesta Luengo¹, Miguel Carbone¹, Claudia Banchoff Tzancoff²,
Claudia Queiruga², Christian Rodriguez¹

¹High Center for Information Processing (CeSPI), UNLP, 50 and 115, La Plata (1900), Argentina
²Laboratory of Investigation in New Information Technologies (LINTI), Computer Science School, UNLP, 50 and 120, La Plata (1900), Argentina
{ncuesta, mcarbone, car}@cespi.unlp.edu.ar
{cbanchoff, claudiaq}@info.unlp.edu.ar

Abstract. One of the great difficulties facing the informatic systems development team of the Department of Development of the CeSPI of the National University of La Plata (UNLP) is the use and maintenance of the service cloud. This issue triggered a theoretical-practical analysis of the state of the art around service oriented architectures and a proposal for a new design for the service cloud.

A real use case was implemented from this design proposal, as well as a reduced cloud architecture, which included everything from a Gateway API routing the incoming requests to services organized according to the guidelines of the microservices pattern. This approach encourages the decoupling for the architecture components, simplifying its development, maintenance, deployment and scalability.

This paper presents the advances achieved in the redesign of the UNLP service cloud and a case study implemented on it.

Keywords: Cloud, Web Service, API, REST, Microservices, Software Architecture Design Patterns, ESB, API Gateway, SOA, Loose Coupling.

1 Introduction

CeSPI¹ is the computing center of the National University of La Plata (UNLP) and its mission is to encourage, implement and manage ICT². The participants in the work here presented include developers and systems analysts of the Direction of Development of the CeSPI and teachers of the Computer Science School, who have been involved in the survey, implementation, maintenance and production of multiple daily use Web applications in the different academic units (high schools and university schools belonging to the UNLP) and other administrative dependencies of the University. In the course of over 7 years of work, multiple architectural paradigms have been used for the development and integration of these applications.

The implementation of the current UNLP architecture is an application based on a REST architecture, which we call “Integrador”, and whose main function is to concentrate the information, making it easier for different applications to unify and correlate data. Although its implementation was adequate and functional for the needs of the moment in which it was developed, the change in requirements and technological advancement have posed the need for a new analysis that led to the solution proposed in this paper.

¹ High Center for Information Processing of the UNLP – <http://www.cespi.unlp.edu.ar/>

² Information and Communications Technology

Section 2 describes the current UNLP service cloud, the “Integrador”, as well as the difficulties of its use and maintenance that triggered the proposal offered in this paper. Section 3 enumerates the technologies evaluated to achieve the final redesign proposal described in section 4. Section 5 describes the case study and finally conclusions are exposed in section 6.

2 The current UNLP cloud

“Integrador” is the implementation of the UNLP cloud service currently in use. It is a Web-based system that concentrates information, allowing multiple Web applications to unify data, combine and correlate the information each of them has. It contains information such as unique identifiers for different types of documents (e.g., 1 equals National Identity Number, 2 equals Personal Identity Booklet Number, 3 equals Passport Number), values identifying the academic units or dependencies of the University (e.g., 33 is the Computer Science School, 26 is the CeSPI, etc), and data related to persons linked to the UNLP, among others.

The services provided by “Integrador” can only be consulted, i.e., it is not possible to modify or destroy them. The applications that consume this information, cloud clients, access it by means of different Web services.

When analyzing the current state of the cloud service, we find a great amount of monolithic applications containing strongly coupled components, which makes changing, testing and deploying them difficult [1, p-29].

The design of “Integrador” does not allow efficient escalation. In order to achieve that, it is necessary to replicate the virtual instance of “Integrador” and implement a load balancer that allows redirection of requests to their replicated instances. The same happens when escalating the APIs that compose the cloud – they escalate poorly. The strong coupling of the applications and their respective APIs makes it necessary to replicate each application and its API, as they have an API implemented in the same virtual instance. The generation of so many applications with their respective APIs results in a proliferation of point-to-point connections, generating dependencies between them.

As regards caching, each client implements their cache without server directives, i.e., there is no centralized caching policy or shared cache, which generates unnecessary processing in the APIs.

Given the amount of applications that use the services of this cloud for its basic functioning, it is highly relevant to optimize its performance, minimize maintenance and achieve a scalable and fault tolerant architecture.

3 Analyzed technologies

The aforementioned issues triggered the research and generation of a proposal for their solution. The following section describes the evaluated technologies that were used to base decisions made regarding the optimization of the UNLP service cloud architecture.

3.1 Service-Oriented Architecture

A Service-Oriented Architecture (SOA) establishes a design framework for integrating distributed and independent applications, allowing network access to its features, offered as services. SOA is generally implemented by means of Web services, a standard based technology and independent from the platform that

provides the data. Thus, SOA can decompose monolithic applications into a set of services [2, p. 2].

This architectural model includes practices and processes based on the fact that distributed systems are not controlled by the owners themselves. Different equipments, dependencies or even different organizations may manage these distributed systems. In the past, a great amount of methods has been proposed to solve the problem of the integration of distributed systems by means of the elimination of heterogeneity, but experience has shown that these approaches do not work. [3, p. 14].

Distributed systems from medium to large scale usually have different owners, which are usually heterogeneous. The approach followed by SOA accepts this heterogeneity based on the principle that it is necessary to deal with the fact that most legacy systems that are already in production will remain in production [3, p. 15].

3.2 Microservices

One of the ways to implement SOA is by means of the microservices architecture pattern. Monolithic applications generally consist of strongly coupled components, which are part of a single deployable unit, which is unpractical and generates difficulties as regards changing, testing and deploying the application. For this reason, big IT organizations with applications with these features tend to use monthly cycles for the deployment of their products.

The microservices pattern deals with these matters, separating the application into multiple deployable units called service-components³, which may be developed, tested and deployed independently from another service-components [4, p. 27].

As shown in Fig. 1, all client requests are performed through a layer called *user interface layer*, which is in charge of accessing service-components.

Taking into account that the main components of an application are divided into smaller deployable parts individually, applications built using the microservices architecture pattern are generally more robust, provide better scalability and can provide support for continuous delivery, enabling production environment updates by means of real time deployments, eliminating the need for monthly or weekly updates [4, p. 33]. All this is possible given that the change is generally reduced to a single service-component and only the units that change must be updated. As previously mentioned, this is a noticeable improvement in the face of monolithic application development, where the strong coupling of its components leads to fragile applications that tend to fail with each new deployment.

³ A service-component is a service logic grouping basic unit, its granularity may vary from a simple module to a great part of an application.

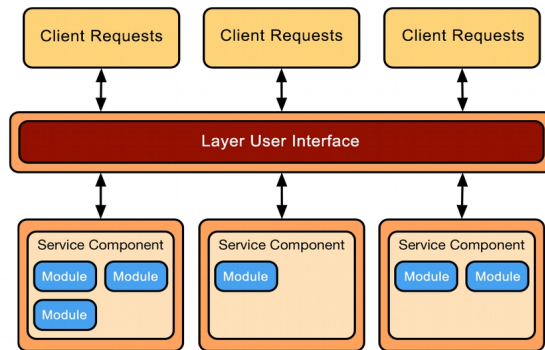


Fig. 1. Basic microservices architecture.

Although there are multiple ways to implement the microservices architecture pattern, the three main topologies are: REST-based API, REST-based application and centralized messaging [5, p. 29]. According to the analysis performed, the centralized messaging topology presented in Fig. 2 is the one that will be implemented, due to its adequacy to the needs of the project. This topology is similar to the REST-based application topology, only that instead of using REST to access a service component remotely, it uses a centralized and light message broker.

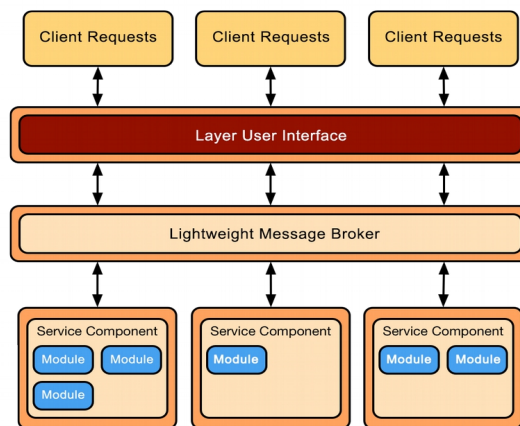


Fig. 2. Centralized messaging.

3.3 Enterprise Service Bus

Although UNLP service cloud applications are integrated, this integration generates a great amount of point-to-point connections, resulting in a high degree of coupling. One of the main SOA design principles is service loose coupling and a usual way to implement it is through an Enterprise Service Bus (ESB): a means of communication that connects and abstracts service providers and consumers.

ESBs themselves do not implement service oriented architectures, but provide the features by means of which it is possible to implement one, i.e., they provide an

abstraction layer for the endpoints, thus achieving flexibility and easier service connection.

There are multiple different opinions regarding the exact role and responsibilities of an ESB, mainly due to the existence of multiple technical approximations to implementing an ESB [3, p. 47]. According to the technical and organizational approaches adopted for the application of the ESB, it may imply one or more of the following tasks:

- Providing connectivity.
- Information transformation.
- Intelligent routing.
- Management of security aspects
- Dealing with service reliability.
- Service management, activity monitoring and logging.

4. UNLP services design optimization proposal

The previously exposed analysis resulted in the proposal of a service-based architecture that is more decoupled than the current service cloud, thus minimizing maintenance and development costs and simplifying its deployment in production environments. The proposed solution is based on standards that allow the integration of heterogeneous systems, accepting the fact that most legacy systems that are currently in production will remain in production, thus making the underlying infrastructure facilitate the incorporation of changes that may be needed in the CeSPI and UNLP.

The service model facilitates information access and consumption through the network. Given that services are independent and autonomous, they can be combined as many times as needed in a simple manner, generating new applications that respond to the constantly evolving needs of the UNLP [2, p. 3]. The possibility of adding and combining services makes this strategy a highly beneficial option, with the goal of creating complex services and applications that are independent from underlying technologies. Therefore, the service cloud design optimization has taken into account features such as redundancy, scalability, decoupling, simplicity and standard use. The new name for the proposed service cloud is “Cloud”.

4.1 Redundancy and Scalability

The application scalability model called scale cube [6] classifies the different ways of scaling applications in 3 axes: X-axis scaling, Y-axis scaling y Z-axis scaling. The “Integrador” and the APIs integrating the cloud do not scale efficiently (X-axis scaling), which is the reason for the proposal of a new UNLP service cloud architecture that must be replicable and scalable.

In order to achieve this, each API must be executed in an independent virtual instance, so it can be replicated as many times as necessary (X-axis scaling). The abstraction layer of these replicated instances will be a load balancer, implemented with NGINX, that will serve both for load balancing and for failover, giving continuity to services in case that one of the replicated instances fails. The load balancer will be able to distribute incoming requests to a group of servers (backends) according to a decision and weighing algorithm called scheduler. This structure that is transparent to the client avoids direct accesses between service providers and consumers.

A shared cache was implemented in front of the load balancer using Varnish⁴, which avoids unnecessary access by keeping an in-memory copy of the replies generated by one of the replicated instances, achieving better response times.

4.2 Decoupling

As previously mentioned, one of the principles of SOA design is loose coupling, frequently implemented by means of an ESB, in charge of providing interoperability among different platforms.

The integration achieved in the current service cloud results in the proliferation of point-to-point connections between systems, which is often known as a “spaghetti architecture” (as exemplified in Fig. 3), generating dependencies between the applications. Although the problem of interoperability between applications is solved, it is difficult to maintain [7, p. 4].

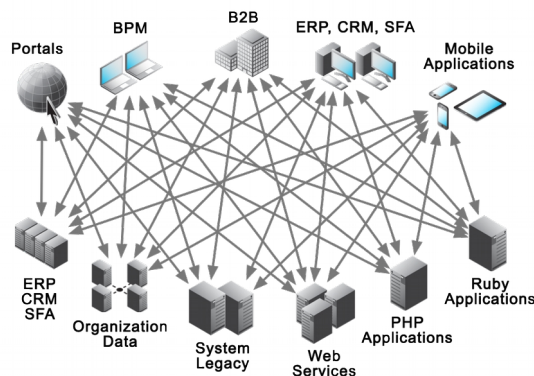


Fig. 3. Scheme of point-to-point connections.

In an architecture implementing ESB, applications communicate through this central bus acting as a message broker among them. This architectural system is called “mediation”. The number of point-to-point connections necessary to allow communications between applications is minimized, simplifying the maintenance and deployment of a system while decreasing the degree of direct dependency that may exist between the instances. As shown in Fig. 4, consumers still use the same endpoint where the task is delegated, when messages arrive, the mediator distributes them to the instances that provide this service [3, p. 52].

The new architecture will use Tyk⁵ as an ESB in front of Varnish, adding mechanisms for authentication, rate limiting, monitoring and a single access point to all the cloud. Moreover, including the ESB avoids the need to develop extra functionalities in the APIs.

In the interest of working on the decoupling of the platforms, the proposal is to develop an API dedicated to serving reference data in which to implement the necessary services that will allow access to this information from different applications. This API will comprise the endpoints of the Integrador that will be in use.

⁴ Varnish is an HTTP reverse proxy, sometimes referred as an HTTP accelerator or Web accelerator, that stores files and file fragments in memory which reduces response times and bandwidth for the same requests [4, p. 20].

⁵ API Gateway Open Source - <https://tyk.io/about-tyk>

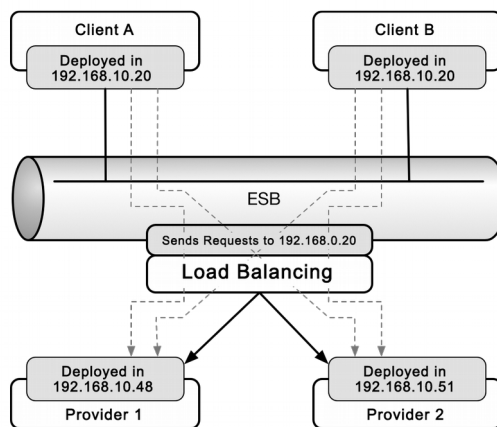


Fig. 4. Scheme of connections with mediator ESB.

Additionally, it is necessary to decouple the APIs from their respective applications and rewrite those that are currently in production using Ruby on Rails as a development framework. Version 5.0 of Ruby on Rails incorporates a new Web application creation mode: api mode. This mode launches applications configured with a reduced subset of components obtained from the elimination of those unnecessary for Web API development, and prepared to serve content in JSON format by default, seizing the main caching techniques for this type of applications.

Applying the microservices pattern will result in different service-components that used to be implemented in each application and coupled to them, and will now be implemented in a new independent and dedicated application. Thus, the logic of the application is decoupled from the data generated by it, allowing it to scale horizontally in a simple manner. This solution will allow independence from the language used for the application development: application can be developed in Ruby, PHP, JavaScript, Java or any other language different from that used to develop the APIs, generating an independent service layer where applications will delegate to the APIs the access, management and persistence of data that they themselves generate.

4.3 Simplicity

Monolithic applications consist of strongly coupled components that are part of one single deployable unit, which makes it difficult to apply changes, test and deploy without service failures. The microservices pattern deals with these matters, separating the application into multiple deployable units, which can be developed, tested and deployed independently from one another [1].

Dividing the application into smaller, deployable components that are independent facilitates development, testing and production due to the change being isolated from a service-component, enabling better control.

4.4 Standard use

Standardization is the process by means of which commonly accepted norms are established that allow communication between different applications⁶. Therefore, for each application that requires so, an API must be implemented to give access to the

⁶ Dictionary of the Royal Spanish Academy

data it generates, based on the JSON API⁷ specification. This standardization facilitates the development of clients that consume information from different services of the APIs, since they define rules that specify how the data can be accessed, what will be their reply structure and even assist in achieving independence from the language in which these clients are written, which must only respect the specifications to implement access to services.

5. Case Study

In order to test the proposed architecture, the process of registering a new Single Sign-on (SSO) user for the UNLP service cloud was used as a case study. All the agents that are part of the UNLP staff have a single user to access the SSO integrated applications, among them, to consult their pay stub, use an application as part of their daily tasks, present extension projects or access any application that may be developed as a UNLP service in the future.

In the process of self-management of a new user, the agent must complete the requested data in a series of previously-established steps leading to access to the applications using the SSO scheme. We can summarize the registration process in the following steps:

1. The agent enters the self-managed application and indicates they wish to register a new user. For this, they must enter their institutional email account in order to receive an access link to effectively begin the process of registration of a new user.

2. Upon accessing the link received by email, the application will ask the agent to identify with the type and number of Identification Document, asking to confirm that they are still among the University agents without a single access user.

3. Once the agent is identified, they are suggested a username following the username policy, which can be modified as part of the process of entering information. This username is confirmed as vacant and as abiding by the policy.

4. Once the username is selected, the agent is asked to enter an alternative email account for a second means of contact.

5. To finish entering personal data, the agent is asked to indicate the dependency of the UNLP where they are employed. This is asked with a captcha to avoid bots attempting mass registration of new users and to keep ill-intentioned users from registering users in the name of real agents.

Decomposing the services of the self-managed process, we identified the following dependencies as services of “Cloud”:

- **Reference services:** to visualize and mark dependencies (or academic units) and lists of identification document types.
- **Staff information services:** to identify the person and later consult the academic units in which they are employed.
- **User services:** to consult the existence of a username for the selected person, suggest usernames that abide by the defined name policy and the creation of the new name.
- **Notification service:** to send emails.

Developing the functional prototype implied:

- Implementing the aforementioned services (reference, staff information, user and notification), in which we only included the endpoints necessary for the logic of the case study.
- Implementing a new version of the registration client application, respecting the aforementioned steps, that will consume all the information of the services of “Cloud”.

⁷ Specifications to build APIs in JSON - <http://jsonapi.org>

- Developing a gem⁸ that encapsulates the service access logic (client) from authentication to query and abstraction of objects from the response of their APIs, based on the JSON API standard.

For a more detailed explanation, it is advisable to consult the thesis “Proposal for the Redesign of the UNLP Service Cloud (Propuesta de rediseño de la nube de servicios de la UNLP)” [8].

6. Conclusions

The proposed architecture is scalable and more robust, with high availability, organized in layers (Gateway, Caching L1, Balancing, Services, Caching L0 and Persistency), all this independent from the technologies used in each layer. The technology of any of the layers can be changed transparently without it affecting service providers or consumers.

With the concept test we obtained a notion of the cost entailed in implementing services following the microservices pattern and taking these changes to our applications, which allowed for an estimation of the time and resources necessary for the implementation of this change in existing and future developments.

The key point and benefit of the proposal is the decomposition of application into microservices that simplifies development, testing and deployment, similarly achieving looser coupling and easier extension of the service cloud.

Using an ESB allows for the replacement of a point-to-point connection topology by a star topology, simplifying maintenance and error detection. At the same time, the functionality is deleted from the APIs (authentication) and new functionalities are achieved (rate-limit, monitoring, and logging, among others), centralized and delegated to another layer.

7. References

1. Richards Mark. Software Architecture Patterns - Understanding Common Architecture Patterns and When to Use Them. O'Reilly, Febrero 2015.
2. Microsoft Corporation. La arquitectura orientada a servicios (soa) de microsoft aplicada al mundo real, Diciembre 2006.
3. Josuttis Nicolai M. SOA in Practice: The Art of Distributing System Design. O'Reilly, paperback edition, 2007.
4. Francisco Velázquez; Kristian Lyngstøl; Tollef Fog Heen; Jérôme Renard. The Varnish Book. Manning, 2016.
5. Richards Mark. Software Architecture Patterns - Understanding Common Architecture Patterns and When to Use Them. O'Reilly, Febrero 2015.
6. akfpartners.com. Splitting applications or services for scale, Mayo 2008.
7. D'Emic David, Dossot; John. Mule in Action. Manning, Febrero 2010.
8. Cuesta Luengo, José Nahuel; Carbone, Miguel. Propuesta de rediseño de la nube de servicios de la UNLP. Junio 2016.

⁸ Name given to reusable libraries in Ruby

Flip-Flop. Aplicación de Buenas Practicas a partir de la Gamificación

Facundo Rodriguez², Franco Vannasaeng³
Silvia Esponda¹, Ariel Pasini¹, Patricia Pesado¹

⁽¹⁾Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – Universidad Nacional de La Plata
50 y 120 La Plata Buenos Aires
{sesponda, apasini, ppesado}@lidi.info.unlp.edu.ar

⁽²⁾iseka.87@gmail.com

⁽³⁾francodv@gmail.com

Abstract. En la actualidad la gestión de la calidad es una parte fundamental dentro de cualquier organización al momento de prestar un mejor servicio o tener un producto que cumpla las exigencias del cliente, aumentando la competitividad y optimización a la hora de desarrollar un proyecto de software. La Norma ISO/IEC 29110 ha sido creada para que las PyMEs alcancen la mejora buscada, proponiendo mejores prácticas.

Es un hecho habitual que, por cuestiones de tiempo, fallas en la comunicación, etc., el hecho de seguir las mejores prácticas, no siempre es posible o, en otros casos, simplemente se carece del conocimiento de las mismas y de sus beneficios.

La realización de una herramienta que incentive e inculque a sus usuarios sobre el uso de las mejores prácticas, haciendo uso de la Gamificación resulta ideal.

Este artículo presenta Flip-Flop, una componente de accesible inserción en herramientas de Gestión de Proyectos, donde se propone un juego con la finalidad de incentivar a los integrantes del grupo desarrollador de software a seguir las buenas prácticas descritas en la norma ISO/IEC 29110, dando recompensas para prosperar en el juego.

Keywords: ISO/IEC 29110- Gestión de procesos – Gamificación.

1 Introducción

ISO/IEC/IEC 29110 "Perfiles de ciclo de vida para pequeñas organizaciones" [1] es una norma creada para PyMEs dedicadas al desarrollo de software, que colabora con la realización de proyectos utilizando guías de gestión de proyectos y desarrollo de software, incrementando así las probabilidades de éxito en función de costo, tiempo y alcance, ayudando a las empresas a percibir y adecuarse a los cambios tecnológicos.

La implementación de esta norma aumenta la competitividad y optimización para desarrollar un proyecto de software, con mayor control en cada etapa. Impacta directamente en la calidad de software mejorando la imagen de la empresa y reduciendo riesgos en el desarrollo.

Es un hecho habitual que, por cuestiones de tiempo, fallas en la comunicación, pocas reuniones con clientes, y otras tantas situaciones comunes en la mayoría de equipos de desarrollo de software, seguir las mejores prácticas no siempre es posible o, en otros casos, simplemente se carece del conocimiento de las mismas y de sus beneficios. Por esta razón se ve la necesidad de lograr una herramienta que incentive e inculque a sus usuarios sobre el uso de las mejores prácticas. Para lograr esto la idea de incorporar un juego con bases en la Gamificación resulta ideal.

El objetivo general de la Gamificación es influir y motivar a los usuarios a que logren la adquisición de hábitos y alcanzar objetivos. Para ello, se incentiva al jugador a participar, compartir e interactuar en alguna actividad de forma individual o con la comunidad. Una Gamificación eficaz, dinámica y con una rica experiencia puede ser utilizada para llevar a cabo una variedad de objetivos en todo tipo de ámbitos y sectores.

El trabajo plantea la creación de una componente de accesible inserción en herramientas de Gestión de Proyectos, donde se propone un juego con la finalidad de incentivar a los integrantes del grupo desarrollador de software a seguir las buenas prácticas descritas en la norma ISO/IEC 29110, dando recompensas de algún tipo en el desarrollo del juego.

Para el desarrollo del componente propuesto se emplearon mecánicas y dinámicas de juego para adquirir hábitos y alcanzar objetivos, lo que se conoce como Gamificación [2]. Dicho concepto hace referencia a la aplicación de mecánicas de juego [3] en entornos no lúdicos, es decir, distintas acciones, comportamientos, técnicas y mecanismos de control que se utilizan para convertir en juego una actividad. Se trata de los aspectos que, en conjunto, crean una experiencia atractiva y fácil de adhesión para el jugador.

El juego, como el que se propone en este proyecto, es una manera de acercar el conocimiento de una forma diferente, en donde el aspecto lúdico resulta esencial en situaciones de aprendizaje.

En la siguiente sección se describe brevemente la Norma ISO/IEC 29110. En la tercera sección se presenta el termino Gamificación en el cual se basa el juego propuesto. A continuación se describe la herramienta propuesta para lograr una inserción mayor en las buenas prácticas. Por último se presentan las conclusiones y trabajos futuros.

2 Buenas Practicas en PyMEs – ISO/IEC 29110

La mejora de procesos en una organización se define como un conjunto de tareas llevadas adelante para obtener productos o servicios de mejor calidad a partir de la revisión y adaptación de sus procesos. Si se ve a la organización como una empresa prestadora de servicios, el concepto de calidad está directamente relacionado a la obtención de la satisfacción de las necesidades del cliente. En consecuencia, el nivel de calidad de la organización estará ligado al nivel de calidad de sus procesos.

Estos procesos deben ser lo suficientemente ordenados para obtener la confiabilidad en las estimaciones y además ser flexibles para adaptarse rápidamente a los cambios que,

particularmente, la industria del software requiere. Este es el gran desafío que hoy enfrenta este tipo de organizaciones.

La norma ISO/IEC 29110 "Perfiles de ciclo de vida para pequeñas organizaciones" aborda las necesidades específicas de las pequeñas empresas para hacer frente a los problemas de pequeñas empresas. Los requisitos básicos de un proceso de desarrollo de software deben adaptarse a las necesidades del proyecto y contribuir al éxito del mismo. La característica central de las entidades a las que apunta la norma ISO/IEC 29110 [4] es el tamaño, sin embargo, hay otros aspectos y características de las pequeñas empresas que pueden afectar el proceso de desarrollo. Dicha norma tiene por objeto resolver los problemas señalados anteriormente y aborda las necesidades específicas de las pequeñas empresas para hacer frente a los problemas de pobre adopción de estándares.

3 Gamificación

El término Gamificación posee múltiples interpretaciones. Sin embargo, la definición que mejor se ajusta al término es la siguiente: *El uso de técnicas propias de los juegos y las mecánicas de juego para atraer a los usuarios y resolver problemas [5].*

El jugador es la base de la Gamificación. En cualquier sistema de juego, la motivación del jugador en última instancia conduce el resultado. Por lo tanto, la comprensión de la motivación del jugador es fundamental para la construcción de un sistema de Gamificación exitoso.

Una teoría de por qué la gente está motivada a jugar juegos sostiene que hay cuatro razones subyacentes, que pueden ser vistas juntas o por separado como motivadores individuales: 1) el dominio/maestría/superioridad; 2) liberarse del estrés; 3) divertirse; 4) socializar. Cuanto más se sabe acerca de los jugadores, más fácil es diseñar una experiencia para impulsar el comportamiento de los mismos en la forma deseada

4 Flip – Flop

Flip-Flop, es una aplicación lúdica desarrollada para el sistema operativo Android [6], pensada para ser aplicada en herramientas de gestión, orientada a grupos de desarrollos de software de PyMEs, con el objetivo de incentivar la aplicación de buenas prácticas, propuestas en la norma ISO/IEC 29110 [7].

Flip-Flop entrega premios a los jugadores en base al comportamiento de los mismos, las tareas de los jugadores son gestionadas mediante Trello [8] (herramienta de gestión seleccionada). Para lograr esto, FLIP-FLOP tiene la capacidad de: obtener desde Trello, toda la información de jugadores, tareas y actualizaciones de las mismas; realizar el cálculo y entrega de puntos a los jugadores; permitir a los jugadores visualizar sus logros e interactuar con los otros jugadores; registrar todas las tareas realizadas por los jugadores y datos propios del juego con fines estadísticos que sirvan para crear un perfil de jugador [9].

Trello es una herramienta web de gestión de proyectos, basada en el método Kanban [10] de gestión de proyectos, con tarjetas (representación de una actividad descrita en una frase) que viajan por diferentes listas en función de sus estados (pendientes, en proceso, terminados). Así, se puede tener una lista de tareas por hacer, que se están haciendo y hechas.

Las listas se encuentran dentro de tableros, dentro de estas se encuentran las tarjetas. Las mismas van cambiando de lista conforme se van realizando, o pasan por diferentes fases. Las tarjetas soportan cualquier tipo de documentación, imágenes, vídeos, listas, comentarios, etc.; por lo que sirven también como instrumento de comunicación en grupos de trabajo, acerca de un determinado tema.

La elección de Trello como herramienta de gestión reside principalmente su API web RESTful [11], donde cada tipo de recurso (por ejemplo, una tarjeta, un tablero o un miembro del equipo de desarrollo) tiene un URI [12] que se puede interactuar. No solo esto presenta una gran ventaja, sino que nos permite trabajar con paradigmas y arquitecturas de programación modernas. Adicionalmente la presencia de una API de este estilo trae asociada una documentación completa, fundamental para el desarrollo de una componente como FLIP-FLOP.

4.1 Prácticas seleccionadas

De las prácticas propuestas por ISO/IEC 29110, FLIP-FLOP trabaja con un subconjunto de las actividades de Implementación de Software:

- SI.4 Construcción del software
- SI.5 Integración del Software y Pruebas
- SI.6 Entrega del producto

En base a la selección de actividades, se obtuvo un subconjunto de roles que son los involucrados en las actividades anteriormente listadas:

- TL: Líder Técnico
- PR: Programador
- AN: Analista
- DES: Diseñador

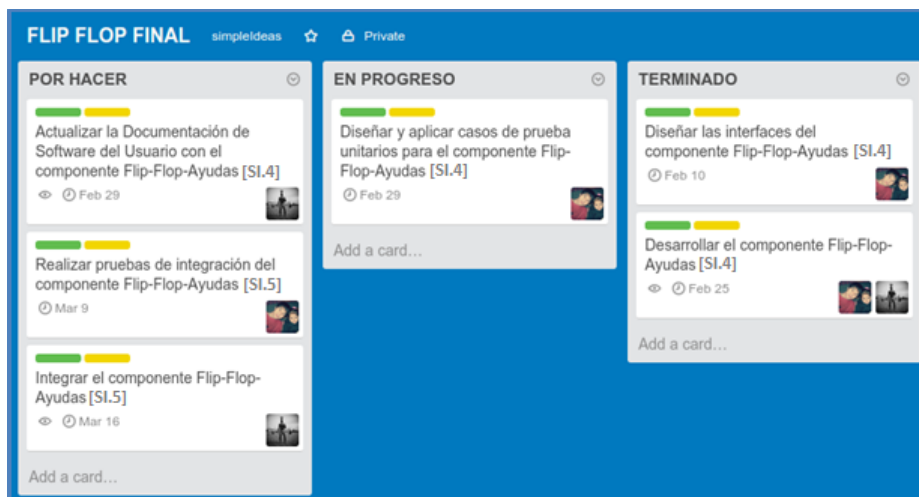


Fig 1. Tablero de Trello Flip-Flop 1

Tablero de Trello-Flip-Flop 1

4.2 ¿Cómo jugar?

El Líder Técnico es el encargado de configurar inicialmente Trello para trabajar con FLIP-FLOP. En primer lugar debe crear un tablero con las listas mencionadas en la sección anterior y las etiquetas correspondientes a los roles y las dificultades de las tareas. Una vez que el tablero ha sido creado se deben crear las tareas correspondientes a las mejores prácticas e ir ubicándolas en la lista “POR HACER”. Estas son las tareas que los jugadores podrán asignarse más adelante. Luego, agregar a los miembros de su equipo al tablero recientemente creado en Trello.

Una vez finalizada esta configuración, el Líder Técnico y los integrantes del equipo pueden jugar con FLIP-FLOP en Trello (Fig 1).

4.3 Desarrollo del juego

El Líder Técnico del equipo de desarrollo es el encargado de incorporar tarjetas a Trello. Crea tarjetas y las coloca en la lista “POR HACER”, dejándolas visibles a los jugadores. Cada una de estas tarjetas está relacionada a una buena práctica de la ISO/IEC 29110 y debe ser válida para el juego.

Por otro lado, los jugadores se asignarán las tareas que deseen realizar. No hay restricciones en la cantidad de tareas ni en el tipo de las mismas que puede elegir un jugador. De esta forma cada jugador tiene la posibilidad de trabajar en n tareas y debe actualizar el estado de las mismas a medida que las vaya resolviendo. Es un requisito que cuando se finalice una tarjeta la misma quede situada en la lista “TERMINADO”.

Todos los viernes a última hora se realiza la entrega de puntos. Se analiza el trabajo de cada jugador y se entregan los puntos, medallas y/o premios correspondientes. Para esto se tienen en cuenta las tarjetas que se encuentren en la lista “TERMINADO” al momento del cálculo; por esta razón es fundamental mantener Trello actualizado. De esta forma queda finalizada la competencia semanal correspondiente, de donde se obtendrán los respectivos ganadores y perdedores. Se denomina a cada competencia semanal como una partida de FLIP-FLOP.

4.3.1 Solicitudes de ayuda

Durante el desarrollo de una tarea, muchas veces se necesita ayuda de otro integrante del equipo de desarrollo para resolver alguna situación particular. Para poder representar esta situación en FLIP-FLOP, y también para tener un seguimiento de las acciones de los jugadores, se brinda la funcionalidad de “Solicitar ayuda”. Cuando un jugador necesita ayuda para la resolución de una tarea puede mandar una solicitud al equipo.

En la sección “MIS TAREAS” se listan las tareas que el jugador posee en la lista “EN PROGRESO” de Trello. Desde cada una de ellas se puede crear una solicitud desde la opción “Solicitar ayuda”. Al resto de los jugadores del equipo se les enviará una notificación informando acerca de esta solicitud.

Por otro lado, en la sección “SOLICITUDES DE AYUDA” se listan todas las solicitudes de ayuda actuales del equipo de desarrollo. Cada solicitud contiene la tarea relacionada, el jugador que creó la solicitud y el tiempo transcurrido desde la creación de la misma. Cuando un jugador responde a una solicitud se le envía una notificación al emisor de la solicitud informando al jugador que ha decidido ayudarlo (ayudante) para que puedan avanzar juntos en la resolución de la tarea lo antes posible, y en la tarjeta de Trello correspondiente se agrega al ayudante como un miembro más. En consecuencia el ayudante tendrá una tarjeta más en la sección “MIS TAREAS” pero con la etiqueta “AYUDANDO”. De esta forma cada jugador puede observar fácilmente cuáles son sus tareas y en cuales está como ayudante.

Para incentivar el uso de la funcionalidad se tendrán en cuenta estas ayudas en el cálculo de puntos. El ayudante ganará un porcentaje de los puntos obtenidos por la realización de dicha tarea.

Los jugadores no pueden hacer uso de esta función ilimitadamente. Los mismos comienzan el juego con 3 pedidos de ayuda a su disposición, pero una vez usados se debe cumplir una condición que está directamente asociada a la actividad del jugador para poder seguir usando la funcionalidad. La condición hace referencia a la relación entre la cantidad de veces que en su equipo se enviaron solicitudes de ayuda y la cantidad de veces que el jugador ha sido ayudante en dichas solicitudes.

Se define de la siguiente manera;

Siendo:

n el número de solicitudes de ayuda que se enviaron en un equipo;

j el número de jugadores del equipo; y

a el número de veces que el jugador brindó ayuda;

el jugador puede solicitar ayuda si y sólo si $a \geq n / (j * 2)$

A modo de ejemplo: En un equipo de 4 jugadores, para que un jugador pueda pedir ayuda (exceptuando las 3 veces que se dan al inicio del juego) debe haber sido ayudante en 1 de cada 8 pedidos de ayuda. Es decir, si hubo 17 pedidos de ayuda un jugador puede usar la funcionalidad si brindó ayuda en al menos 2 oportunidades. De lo contrario no puede hacer uso de la misma.

4.3.2 Documentación de la tarea

Una cuestión no menos importante en los equipos de desarrollo es la falta de documentación. Por diversas cuestiones relacionadas con tiempo, falta de costumbre o simplemente no saber los beneficios de tener una documentación actualizada y completa; la misma queda, muchas veces, relegada.

Por este motivo en FLIP-FLOP se implementó un sistema de recompensas basado en la documentación de las tareas. Un jugador puede documentar la tarea que desee a través de Trello. Una tarea puede tener múltiples archivos adjuntos. Para poder distinguir estos archivos del archivo de documentación que un jugador puede cargar es necesario respetar un prefijo en el nombre del mismo. De esta forma, cuando un jugador desee adjuntar documentación el nombre del archivo debe contener el prefijo “DOC:”. FLIP-FLOP se encargará de identificarlos y premiar a los jugadores.

Esta funcionalidad puede verse como un sistema de recompensas. Específicamente, se diseñó como un sistema de puntos de habilidad en donde el jugador consigue recompensas por realizar actividades específicas dentro del juego:

- En primer lugar; el jugador ganará un puntaje extra.
- En segundo lugar; se establece una relación con la funcionalidad de “SOLICITUDES DE AYUDA”. Cada 5 (cinco) tareas documentadas el jugador ganará un pedido de ayuda.

4.3.3 Sistema de puntos

Los puntos son un requisito de los juego y existen diferentes sistemas de puntuación. En FLIP-FLOP los puntos son un elemento fundamental. El objetivo de los jugadores es realizar tareas y estas tareas les entregarán puntos. A esto se agregan todas las funciones diseñadas que pueden utilizarse para obtener aún más puntos.

Al finalizar cada partida se suman los puntos de todas las tarjetas finalizadas de cada jugador y se incrementa el puntaje total del mismo con el resultado de esta sumatoria. Se define para cada tarea un puntaje básico que hace referencia a los puntos que esa tarea tiene asociados sin tener en cuenta otras variables.

Luego de obtener el puntaje básico, se aplican las siguientes reglas que pueden modificar dicho puntaje:

- Roles involucrados Si la tarea es de un rol diferente al rol del jugador se adiciona un 25% del puntaje básico.
- Fecha de finalización: Si la tarea se finaliza antes de la fecha estimada se adiciona un 20% del puntaje básico. En caso contrario, en el que la tarea se finaliza vencida, se penaliza al jugador restando un 40% del puntaje básico.
- Documentación de la tarea: Si la tarea presenta documentación se adiciona un 20% del puntaje básico.
- Solicitudes de ayuda: Los jugadores que figuran como ayudantes recibirán un 30% del puntaje básico de la tarea en la que ayudaron.

Esta modalidad de cálculo de puntaje es conocida como Puntos de experiencia. Todas las acciones que los jugadores hacen en el juego tienden a incrementar sus puntos de experiencia.

4.3.4 Insignias/Logros

Los Logros pueden ser una gran manera de aumentar la participación de usuarios dentro del juego. Luego de realizar el cálculo de puntos semanalmente se procede a la habilitación (desbloqueo) de Logros. Para el diseño e implementación de esta funcionalidad, se utilizaron los servicios de Google Play Game [9].

4.3.5 Tabla de clasificación

Al igual que para la implementación de Logros, se utilizan los servicios de Google Play Game para incluir una tabla de clasificación en el juego. Se indica de una forma clara la

posición del jugador, e incluso se presentan opciones para filtrar los resultados históricos o de la última semana si se quisiera ver el desempeño de la última partida, por ejemplo.

4.3.6 Métricas y Estadísticas

El juego registra todas las acciones de los jugadores, con esa información, se definieron métricas para poder analizar el comportamiento de los jugadores y luego presentarlas como estadísticas. Esto influye en la competitividad del juego y, por ende, es una fuente de motivación para algunos jugadores. También brinda una herramienta para que ellos mismos puedan saber qué características deben mejorar y en cuales se desempeñan satisfactoriamente.

Por otro lado, y con el objetivo de valorar el desempeño de los jugadores, estas métricas son de gran utilidad para el Líder Técnico. A través del análisis de las mismas podría tomar acciones para corregir y/o mejorar el desempeño de su equipo de trabajo. Incluso se podrían tener en cuenta las mismas a la hora de realizar cambios de categorías, cambios de roles o revisiones salariales. Las métricas definidas son:

- **Documentador:** Se relaciona directamente con la funcionalidad de documentación de una tarea. Se define la misma como el porcentaje de tareas documentadas de un jugador.
- **Versatilidad:** Calcula el porcentaje de tareas de otros roles (no el que se ha asignado) que un jugador ha realizado. Esta métrica puede ser de gran utilidad para que el Líder Técnico detecte a los jugadores que realizan tareas de diferentes roles y pueda así proponer, por ejemplo, un cambio de rol para el jugador.
- **Puntualidad:** Otro de los datos requerido en las tareas de Trello es la fecha de vencimiento de la tarea. Esta fecha, refleja una estimación que previamente el Líder Técnico debe haber realizado como parte de la actividad de Planificación del Proyecto. Esta métrica indica el porcentaje de tareas que un jugador finalizó antes de la fecha de vencimiento de la misma. Con el mismo objetivo se penalizará a aquellos jugadores que finalicen la tarea fuera de fecha.
- **Independencia:** Se define como el porcentaje de tareas que ha realizado sin ayuda.
- **Cooperación:** Se calcula a partir del uso de “SOLICITUDES DE AYUDA”. Indica en qué medida un jugador ayuda a sus compañeros. Esto se calcula como el porcentaje de tareas en las que un jugador figura como ayudante, del total de solicitudes de ayuda que hubo en el equipo al que pertenece. Esta métrica pretende impulsar el compañerismo dentro del equipo.
- **Superación:** Se define como el promedio de dificultad de las tareas que un jugador resuelve. Este simple cálculo dice mucho acerca de un jugador. Aquel jugador que tenga un nivel de superación alto es debido a que realiza tareas dificultosas. Esta métrica utilizada en conjunto con la puntualidad podría servir para detectar futuros cambios de seniority de un desarrollador.

5 Caso de estudio

Se seleccionó un equipo de desarrollo específico compuesto por: un Líder Técnico, un Analista, un Diseñador y dos Programadores. Se creó un tablero de Trello especialmente para la prueba y se les explicó a los integrantes del equipo el objetivo de FLIP-FLOP (Fig 2).

Todos los jugadores del equipo de desarrollo se mostraron entusiasmados por instalar la aplicación y asombrados por el concepto de FLIP-FLOP: una componente lúdica que puede emplearse en el contexto laboral. Se realizó una demostración de las principales funcionalidades del juego y luego comenzaron a explorar por sí mismo las diferentes opciones para lograr habilitar logros. Después de una semana de competencia, se pudo observar cómo los jugadores estaban atentos al tablero de Trello. En particular, a las fechas de vencimiento de sus tareas para poder realizarlas a tiempo. Por otro lado, los jugadores valoraron que FLIP-FLOP actualice las tarjetas de Trello cuando un integrante brindaba ayuda a otro. Otro punto de gran interés fueron los logros definidos en el juego.

Los jugadores pudieron jugar a lo largo de una semana FLIP-FLOP sin ningún tipo de problemas, se sintieron atraídos por el juego y se interesaron por cumplir las tareas que se asignaban.

Se evaluaron las métricas logradas luego del juego y se evidenció que:

- Se logró una mayor generación de la documentación del proyecto, ya que dicha métrica fue mayor al 70% en el 100% de los jugadores.
- La métrica Versatilidad mostró que el 50% de los jugadores afrontó actividades que no eran las correspondientes a su rol.
- Se comprobó que el grupo logró una mayor cooperación interna.

En consecuencia, se deduce que los objetivos planteados para esta prueba fueron cumplidos y se considera una experiencia exitosa.

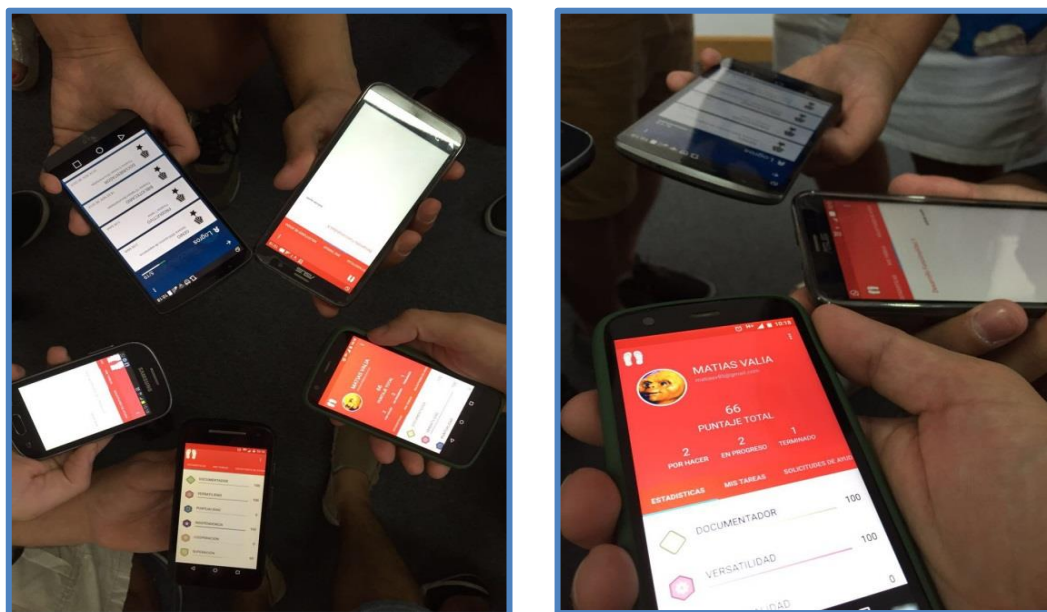


Fig 2. Caso de Estudio

6 Conclusiones

Se logró desarrollar una componente lúdica utilizando las mecánicas y dinámicas de juego para incentivar la utilización de buenas prácticas, en el desarrollo de software. Dicha componente es una aplicación nativa desarrollada para el sistema operativo Android. Adicionalmente, y con el fin de enriquecer la aplicación, se investigaron los servicios que proporciona Google para sistemas de juegos y para notificaciones. Ambos servicios se incluyeron satisfactoriamente. Por otro lado, el desarrollo de una aplicación backend permitió la integración con los servicios de Trello y la persistencia de datos.

El resultado final es una aplicación que utiliza los conceptos de la gamificación para incorporar calidad en las actividades, motivándolo a través de una experiencia de juego dinámica para los usuarios.

La utilización de Flip-Flop permitió mejorar la documentación y el trabajo en equipo (mediante las solicitudes de ayudas) del proyecto donde fue aplicado.

7 Referencias.

1. ISO 29110-2011 Software engineering -- Lifecycle profiles for Very Small Entities (VSEs)
2. Gabe Zichermann, Christopher Cunningham. Gamification by Design. Implementing Game Mechanics in Web and Mobile Apps., ISBN 978-1449397678, Año 2013.
3. Daniel J. Dubois - Giordano Tamburrelli. Understanding gamification mechanisms for software development (<http://dl.acm.org/citation.cfm?id=2494589>)
4. ISO TR 29110-5- 1-1:2012, “Management and engineering guide” 2012
5. Karen Robson, Leyland Pitt , Kirk Plangger, Jan H. Kietzmann, Ian McCarthy. Is it all a game? Understanding the principles of gamification.
6. Android - History, <https://www.android.com/history/#/marshmallow>
7. Oscar Pedreira, Félix García, Nieves Brisaboa, Mario Piattini. Gamification in software engineering – A systematic mapping.
8. Trello, <https://trello.com/>
9. Google, <https://myaccount.google.com/intro>
10. Método Kanban, <http://www.javiargarzas.com/2011/11/kanban.html>
11. Richardson, Amundsen, “RESTful Web APIs”, Año 2013
12. URI, https://en.wikipedia.org/wiki/Uniform_Resource_Identifier

Búsqueda de Funcionalidades basada en Expansión de Consultas para SPLs

Maximiliano Arias^{*‡}, Agustina Buccella^{*‡}, and Alejandra Cechich^{*}

^{*}GIISCO Research Group, Universidad del Comahue, Neuquén, Argentina

[‡]Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

{maximiliano.arias, agustina.buccella, alejandra.cechich}@fi.uncoma.edu.ar

Abstract. El desarrollo de Líneas de Productos de Software (LPS) es un paradigma basado en reuso que se basa en la instanciación rápida de productos similares de un dominio particular. Este paradigma hace uso intensivo del reuso de funcionalidades con un grado de personalización y configuración muy alto. Esto hace evidente la necesidad de un adecuado conocimiento del conjunto de funcionalidades actualmente existentes en la línea, debiendo sostenerse en una completa documentación. De esta manera, un correcto procesamiento de los requerimientos de software favorece al reuso efectivo de las funcionalidades de la línea. En este trabajo se propone un proceso de búsqueda de funcionalidades basado en métodos de expansión de consultas. Estos métodos demostraron mejorar la efectividad de la búsquedas y la precisión en las primeras posiciones del ranking de sugerencias.

Keywords: Líneas de productos de software, expansión de consultas, requerimientos de software

1 Introducción

El desarrollo de Líneas de Productos de Software [5, 9] (LPS) es un paradigma centrado en reuso que busca identificar servicios comunes y variables que definen las funcionalidades de un dominio particular. Encontrar servicios comunes y variantes deriva en la generación de una plataforma de la cual se pueden instanciar productos de software, estableciendo configuraciones sobre los servicios variantes o bien agregando funcionalidades particulares de ese producto. De esta manera, una LPS conlleva un elevado tiempo de desarrollo inicial que puede verse compensado a medida que se instancian más productos.

Generar un mecanismo para el reuso efectivo de las funcionalidades implementadas no es trivial [10] y requiere un esfuerzo que debe acompañar todo el proceso de desarrollo de la LPS misma. Siempre resulta de importancia ser capaz de interpretar de alguna manera los requerimientos que se toman del usuario en lenguaje natural, para poder extraer el conocimiento de los mismos. Este conocimiento es importante porque si se cuenta con el mecanismo indicado puede favorecer a la búsqueda de requerimientos ya implementados, evitando

que se generen funcionalidades duplicadas que puedan perjudicar la evolución de la línea.

En trabajos previos [6, 7] hemos desarrollado una metodología completa para la generación de Líneas de Productos de Software, enfocándonos en particular dentro del dominio geográfico. Dentro de esta metodología existe una serie de artefactos de software que se apoyan en la correcta búsqueda de requerimientos para favorecer el reuso efectivo. Para lograr esto se lleva a cabo una serie de transformaciones que refinan los requerimientos y los transforman en sub-productos que se utilizan en subsiguientes búsquedas. En este trabajo presentamos un *proceso de búsqueda de funcionalidades* en donde nos centramos en los métodos de expansión de consultas para lograr resultados más efectivos. Como contribución principal de este trabajo, se presentan alternativas en el mecanismo de búsqueda de funcionalidades implementadas dentro de la LPS utilizando nuevas relaciones semánticas, reconocimiento de entidades e identificación de categorías sintácticas.

El trabajo se encuentra organizado de la siguiente manera. En la Sección 2 se describen los trabajos previos que motivaron al desarrollo de los métodos de preparación y expansión de consultas presentados. En la Sección 3 se presenta un análisis de trabajos relacionados con nuestra propuesta. Luego se presenta el *proceso de búsqueda de funcionalidades* dentro de LPS a partir de requerimientos del usuario, haciendo hincapié en los métodos de preparación, limpieza y expansión de consultas. La Sección 5 muestra una evaluación experimental, con sus respectivos resultados, realizada para medir la eficacia de los métodos presentados. Finalmente, la Sección 6 exhibe las conclusiones y trabajos futuros que se derivan del presente.

2 Trabajos previos

En trabajos previos [6, 7] se propuso una metodología para el desarrollo de LPS la cual abarca 4 pasos. Como podemos observar en la Figura 1, estos pasos hacen uso de *artefactos de software* numerados de 1 al 6. En primer lugar, durante el paso 1 se deben analizar las fuentes de información del dominio que permiten generar y/o modificar una *taxonomía de servicios del dominio* (artefacto 1). Tras analizar las fuentes de información y generar la taxonomía, se debe conceptualizar el dominio (paso 2) generando un nuevo artefacto de software, denominado *hojas de datos funcionales* (artefacto 3), que contienen los modelos de variabilidad de cada funcionalidad. Estas hojas se crean siguiendo lo establecido en el artefacto 2, la *arquitectura de referencia* y buscando servicios en la taxonomía antes creada. Tras generar las hojas de datos funcionales se debe realizar un estudio de reusabilidad de los componentes existentes. Los diseños son refinados en el paso 3 obteniendo como resultado un esquema de componentes abstracto (artefacto 4). Luego, en el paso 4 se debe diseñar la plataforma definiendo una arquitectura concreta para la misma (artefacto 5). La arquitectura concreta será la que se seguirá estrictamente para la etapa de implementación, que generará los *componentes concretos* (artefacto 6) implementados.

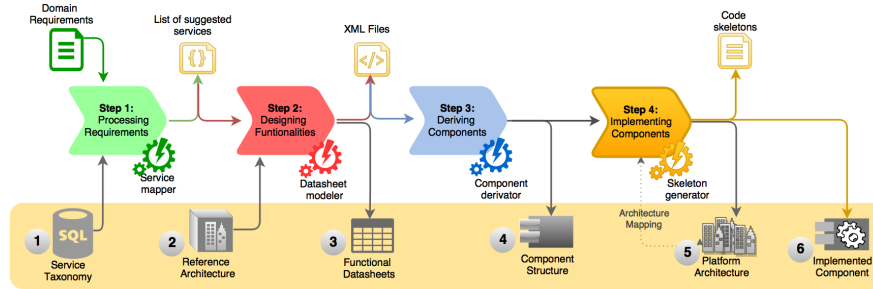


Fig. 1. Proceso de desarrollo de una LPS.

Como podemos observar, dentro de este proceso los artefactos de software cumplen un papel fundamental como soporte para las etapas del proceso de desarrollo. Este trabajo se enfoca en los pasos 1 y 2 del mismo, más particularmente en el proceso de correspondencia de requerimientos del dominio con servicios de la taxonomía. Este trabajo fue también implementado en un herramienta de soporte, llamada *Service Mapper*, la cual genera una lista de servicios candidatos a partir de un conjunto de requerimientos del dominio.

3 Trabajos relacionados

Las técnicas de procesamiento del lenguaje natural y recuperación de información se han utilizado ampliamente para el procesamiento de requerimientos de las LPSs [4]. Podemos distinguir un gran número de trabajos relacionados con la extracción de conocimiento de requerimientos de software y centrados en el procesamiento del lenguaje natural. Por ejemplo, en [2] se presentan técnicas de recuperación de información para el procesamiento de documentación del dominio. En este trabajo los autores procesan grandes documentos de distintas organizaciones y buscan encontrar similitudes que puedan ser plasmadas en un modelo de variabilidad. En [8] los autores proponen el uso de técnicas de clustering para la identificación de relaciones de necesidad entre requerimientos. Estas relaciones son luego utilizadas para asistir a los ingenieros en la construcción de un modelo de funcionalidades (feature model). Finalmente, en [12] se presenta un framework para la generación de un documento de requerimientos anotado a partir de texto plano. Este framework analiza el texto que describe un dominio en busca de agentes, acciones, restricciones, y otras categorías definidas por los autores. Finalmente, la elicitación etiquetada es utilizada para asistir en la construcción de un modelo de variabilidad.

Dentro de las técnicas para el procesamiento natural podemos nombrar el análisis semántico mediante el uso de WordNet [11], la búsqueda de palabras similares mediante el uso DISCO¹ y el proceso reconocimiento de entidades con nombre (Name Entity Recognition - NER) [1]. La primera de estas técnicas

¹ http://www.linguatools.de/disco/disco_en.html

sugiere el uso de un diccionario léxico que organiza las palabras de manera jerárquica, para establecer relaciones de sinonimia, hiponimia e hiperonimia. La segunda hace uso de una base de datos de co-ocurrencias de palabras que establece similitud entre conceptos de acuerdo a la cantidad de veces que éstas se utilizan en conjunto, analizando un gran corpus de datos. Por último, NER sirve para el reconocimiento y clasificación de entidades dentro de un conjunto de categorías predefinidas.

En nuestro trabajo, proponemos la aplicación de técnicas de procesamiento del lenguaje natural y recuperación de información para llevar a cabo la búsqueda de funcionalidades implementadas dentro de una LPS. De esta manera, buscamos generar un mecanismo de reuso efectivo que permita reducir el riesgo de duplicar funcionalidades, eliminando tiempos de desarrollo innecesarios. Así, buscamos comparar la eficiencia de la aplicación de distintas técnicas de expansión de consultas entre sí.

4 Proceso de Búsqueda de Funcionalidades

En nuestro proceso de desarrollo, mostrado en la Figura 1, una funcionalidad es introducida como un conjunto de requerimientos del dominio o de la organización y se transforma para generar los subsiguientes artefactos de software. Como una funcionalidad puede ya existir total o parcialmente implementada en la plataforma, resulta necesario verificar esta existencia para realizar un reuso efectivo. De esta manera, cada artefacto de software se relaciona con otros artefactos de etapas posteriores, generando una secuencia de artefactos que relacionan los requerimientos con el diseño y la implementación.

Para poder garantizar esto es necesario que la búsqueda de funcionalidades dentro de la plataforma sea eficiente. Las funcionalidades se representan mediante requerimientos escritos en lenguaje natural, que debe ser procesado. De esta manera en nuestra propuesta, buscar funcionalidades completas a partir de un conjunto de requerimientos conforma una actividad compleja que se apoya inicialmente en la búsqueda de servicios en la taxonomía del dominio [3]. A medida que una descripción en lenguaje natural crece, también crece la complejidad para procesarla. Así, resulta diferente tratar requerimientos atómicos [3] a tratar funcionalidades completas, haciendo visible la necesidad de algún nivel de control sobre la estructura con la que se escriben los requerimientos. Por esto, este trabajo también propone el uso reconocimiento de entidades [1] y desambiguación de palabras según su clasificación sintáctica.

Como muestra la Figura 2 los requerimientos son ingresados al buscador de funcionalidades (*Functional mapper*) en forma de consultas. Estas consultas son preprocesadas por el *preprocesador de consultas* y utilizadas por el motor de búsqueda para encontrar servicios dentro de la taxonomía de servicios. De esta manera se obtiene un conjunto de servicios candidatos para cada requerimiento, ordenados según el criterio de similitud del buscador. Los n candidatos que obtuvieron la mejor clasificación se convierten en nuevas consultas que se utilizan para buscar dentro del repositorio de hojas de datos funcionales.

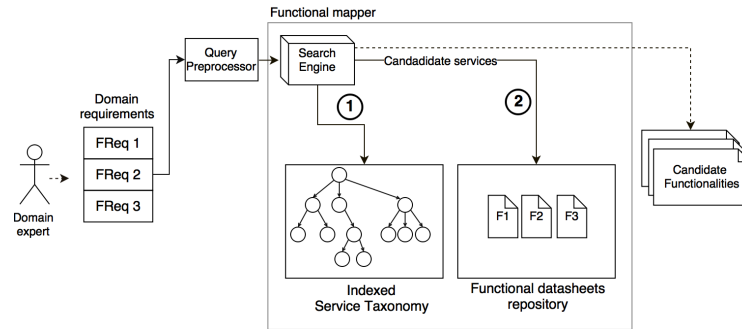


Fig. 2. Búsqueda asistida de una funcionalidad en el repositorio de funcionalidades.

Así, como las hojas de datos se construyeron a partir de la taxonomía, el buscador la utiliza como intermediaria del proceso de búsqueda de funcionalidades. Los requerimientos escritos en lenguaje natural son mapeados a servicios de la taxonomía normalizando la subsiguiente búsqueda. Entonces, si por ejemplo se partiera de una necesidad expresada por un usuario en donde se solicita la posibilidad de que el sistema exporte sus mapas en formato un TIFF esta necesidad se podría expresar mediante un conjunto de requerimientos como:

- *FREQ1: Permitir exportar mapas del usuario*
- *FREQ2: Permitir exportar un mapa en formato TIFF*

Estos dos requerimientos se utilizarían para buscar dentro de la taxonomía de servicios, sugiriendo un conjunto de servicios candidatos que incluiría por ejemplo el servicio *PS-T3 Export visible map* (donde *PS-T3* representa el código que identifica a dicho servicio y *Export visible map* su descripción). La tarea siguiente del buscador es recorrer la base de datos de funcionalidades en busca de hojas de datos funcionales que incluyan a éste y a los demás servicios candidatos (del conjunto de n servicios mencionado anteriormente) ordenándolos según el número de servicios incluidos. En nuestro ejemplo, el servicio encontrado se utilizó durante la creación de la línea de productos para implementar la funcionalidad *Export maps*, que tiene su correspondiente hoja de datos funcional. En este contexto la hoja de datos *Export maps* formaría parte del conjunto de funcionalidades candidatas para la solicitud del usuario, junto con otras funcionalidades similares.

Al mismo tiempo, podemos notar que resulta de gran interés que los servicios de la taxonomía recuperados sean correctos pues estos resultados se utilizarán luego para continuar la búsqueda. Identificamos dos actividades importantes que debe realizar el preprocesador de consultas la *limpieza (query cleaner)* y la *expansión (query expander)*[3]. Dichas actividades se muestran en la Figura 3.

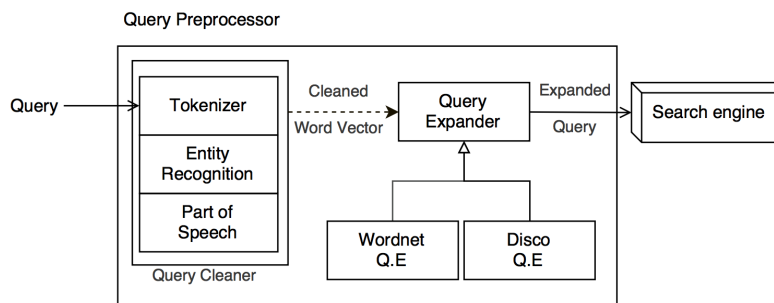


Fig. 3. Esquema del módulo de preprocesamiento de consultas.

4.1 Limpieza

En primera instancia, es necesario preparar la consulta (requerimiento funcional escrito en lenguaje natural) de modo que esta pueda ser expandida correctamente y luego utilizada para la búsqueda. De esta manera, se deben realizar una serie de transformaciones que convertirán cada una de las consultas en un vector de palabras a expandir. Por un lado, se debe comenzar con la separación de las consultas en sus palabras constituyentes (tokenización) de modo que cada una de ellas pueda ser tratada por separado. Luego, el listado de tokens debe pasar por un proceso de eliminación de símbolos, letras, palabras u otros elementos no deseados o que no aportan valor semántico a la consulta, denominados *stopwords*.

Además de filtrar las palabras que no aportan valor semántico, aplicamos la técnica de reconocimiento de entidades con nombre (NER²) para mejorar la tokenización. Una entidad con nombre es una palabra o conjunto de palabras que forma una entidad que puede ser clasificada dentro de una serie de categorías predefinidas. Estas categorías incluyen: *Lugar*, *Persona*, *Organización*, *Dinero*, *Porcentaje*, *Fecha* y *Tiempo*. Así aplicamos el NER presentado en [1], refinando la lista de palabras inicial. Aquellas entidades que estaban conformadas por más de una palabra se agrupan nuevamente y se etiquetan con la clase de entidad que les corresponde.

A su vez, hay que considerar que una misma palabra puede tener distintos significados según el contexto o forma en la cual sea utilizada [13]. Así, aquellas palabras que no fueron reconocidas como entidades se deben etiquetar con su categoría sintáctica para luego facilitar la tarea de expansión. Para esta tarea se utilizó el etiquetador presentado en [13] (part-of-speech) y se etiquetaron según las 36 categorías definidas por el *Penn Treebank Project*³ que incluyen distintos tipos de sustantivos, verbos en distintas formas, conjunciones, adverbios etc.

² <http://nlp.stanford.edu/software/CRF-NER.html>

³ <https://www.cis.upenn.edu/~treebank/>

4.2 Expansión

La expansión de una consulta refiere al agregado de valor semántico mediante algún proceso de expansión, puede llevarse a cabo siguiendo distintos métodos propios del campo de la recuperación de información. Estos buscan explorar relaciones semánticas entre términos para expandir el conjunto de palabras que conforman la consulta. Como dichas relaciones semánticas pueden variar de acuerdo a cada método, utilizamos dos métodos de expansión, explicados a continuación, y comparamos los resultados obtenidos de la búsqueda de funcionalidades con cada método entre sí y con respecto a la búsqueda sin ningún tipo de expansión de consultas (realizada en trabajos previos). La comparación se realizará no solo considerando los métodos en sí mismos, sino dependiendo de las entradas ingresadas a los mismos.

Expansión de consultas utilizando Wordnet Uno de los métodos de expansión presentado en este trabajo hace uso de la base de datos léxica WordNet. Inicialmente se deben buscar las raíces de los términos (*stemming*) utilizando el algoritmo provisto por WordNet. Así, comenzamos a construir el vector de términos expandido que utilizamos para consultar al motor de búsqueda. Como cada palabra está relacionada con un gran número de definiciones, las etiquetas que determinan las categorías sintácticas de cada palabra son usadas para buscar más precisamente dentro del conjunto. Luego se buscan sinónimos (palabras con exactamente el mismo significado), hipónimos (palabra con un significado más específico) e hiperónimos (palabras con un significado más general) y se agregan todos ellos al vector.

Expansión de consultas utilizando DISCO El segundo de los métodos de expansión presentados hace uso de la herramienta DISCO. Esta herramienta establece relaciones de similitud entre palabras mediante un análisis de co-ocurrencias. Es decir, las palabras similares a otra palabra son aquellas que aparecen junto a la palabra inicial dentro de un corpus de texto extraído de Wikipedia. De esta manera, para cada palabra del vector de términos que ya fue limpiado se obtiene una lista de co-ocurrencias ordenada de manera descendente, se toman las k primeras palabras y se agregan al vector. Así, la nueva consulta contiene las palabras iniciales más las k co-ocurrencias más frecuentes de cada una de éstas.

5 Evaluación experimental

El experimento presentado tiene como objetivo evaluar el desempeño de los dos métodos de expansión de consultas, en el contexto del proceso búsqueda de funcionalidades entre sí, y con respecto a la búsqueda de funcionalidades por parte de ingenieros de software. De esta manera, se busca definir cuál de los métodos

descriptos genera los mejores resultados. Para este experimento se contó con la colaboración de un grupo de ingenieros de software que buscaron manualmente funcionalidades dentro de un repositorio a partir de un conjunto de requerimientos de software. El objetivo de los ingenieros era el de encontrar una funcionalidad que se asemeje lo más posible a los requerimientos de software solicitados. Debido a que las funcionalidades guardadas en el repositorio se crearon a partir de los servicios de la taxonomía, los ingenieros utilizaron una aplicación que permitía la visualización de la misma. Así, los requerimientos de software fueron mapeados manualmente a servicios de la taxonomía y luego a funcionalidades.

5.1 Configuración del experimento

Se definió un escenario experimental en el que fueron considerados los 120 servicios de la taxonomía previamente preprocesados e indexados. Con el objetivo de poder medir los resultados de la recuperación de servicios, se consideró un conjunto de 58 requerimientos de usuario definidos en conjunto con un grupo de biólogos marinos del Instituto de Biología Marina y Pesquera Almirante Storni (IBMPAS)⁴ y el Centro Nacional Patagónico (CENPAT)⁵ que se encuentran expresadas en lenguaje natural. Los requerimientos fueron inicialmente correspondidos con servicios de la taxonomía por un experto del dominio para poder luego analizar los resultados.

En trabajos previos [3] se obtuvieron los resultados de corresponder estos requerimientos con servicios de la taxonomía manualmente (llamado *método manual*) y mediante el uso de un buscador que realizaba una expansión básica utilizando WordNet (llamado *WordNet básico*). Estos resultados fueron contrastados contra las dos nuevas variantes que hemos definido: utilizando el método de expansión de WordNet con reconocimiento de entidades (llamado *WordNet mejorado*) y categorías sintácticas, y utilizando el método de expansión de palabras similares con DISCO.

5.2 Resultados

Para analizar el desempeño se consideraron dos métricas pertenecientes al campo de la recuperación de información: *Precision-at-n (acumulada)* y *Recall*. La primera calcula la precisión en diferentes puntos de la lista de candidatos retornados durante la selección. Formalmente, *Precision-at-n* para una consulta individual se define como $Precision - at - n = \frac{RetRel_n}{n}$, donde *RetRel_n* es el número de servicios de la taxonomía relevantes recuperados en las primeras *n* posiciones.

La métrica *Recall* mide el desempeño del proceso de selección de servicios de la taxonomía a partir de los servicios relevantes que recupera. La medida *Recall* es 100% cuando cada servicio relevante es devuelto en la lista de candidatos.

⁴ <http://www.ibmpas.org/>

⁵ <http://www.cenpat-conicet.gob.ar/>

Métrica	Manual	WordNet_Básico	WordNet_Mejorado	DISCO
Precision-at-1	-	0.65	0.63	0.59
Precision-at-2	-	0.75	0.73	0.75
Precision-at-3	-	0.82	0.76	0.81
Precision-at-4	-	0.86	0.76	0.81
Precision-at-5	-	0.96	0.78	0.84
Precision-at-6	-	1	0.96	1
Precision-at-7	-	1	1	1
Recall	0.56	0.74	0.76	0.82

Tabla 1. Resultados obtenidos del experimento para cada uno de los métodos de expansión.

La Tabla 1 muestra los resultados de precisión acumulada y recall en los experimentos realizados. En ella se puede ver la primera columna que muestra los resultados de la búsqueda manual y las siguientes 4 columnas que representan los resultados de los distintos métodos de preprocesamiento de consultas descriptos previamente.

Como se puede observar en la tabla, la medida de recall para el método de expansión de disco presenta una 8% de mejora con respecto al método básico de WordNet y un 6% de mejora con respecto al mejorado. Con respecto a las medidas de precisión acumulada, ambos algoritmos de WordNet se encuentran por arriba de disco en la primera posición de la lista. El algoritmo mejorado de WordNet presenta una desmejoría en su precisión para las posiciones intermedias de la lista, alcanzando el 100% de precisión en la posición 7. Todos los mecanismos de expansión han demostrado ser más de un 18% mejores en encontrar los servicios candidatos con respecto a la búsqueda manual, con su pico máximo en el algoritmo que utiliza DISCO (26% mejor).

6 Conclusiones

El reuso efectivo de las funcionalidades que conforman una LPS se basa en utilizar una funcionalidad existente cuando se obtiene un requerimiento que ya ha sido implementado. De esta manera, es necesario llevar un estricto control sobre cuáles funcionalidades ya se encuentran implementadas, de modo que un requerimiento no conlleve al desarrollo de funcionalidades que ya existen generando duplicados.

A lo largo de este trabajo hemos presentado distintas técnicas que permiten estructurar y buscar funcionalidades dentro de una LPS. Utilizando herramientas construidas a partir de estas técnicas un ingeniero de software podría verificar si un requerimiento entrante necesita o no ser implementado. De esta manera, el reuso efectivo permitirá la instanciación de productos en menor tiempo y favorecerá la evolución reduciendo la redundancia funcional. Hemos demostrado además que el uso de distintas técnicas de expansión de consultas mejoró la efectividad y la precisión de las búsquedas en ciertas posiciones de la lista de candidatos.

References

1. *Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling*, ACL '05, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.
2. V. Alves, C. Schwanninger, L. Barbosa, A. Rashid, P. Sawyer, P. Rayson, C. Pohl, and A. Rummler. An exploratory study of information retrieval techniques in domain analysis. In *Software Product Line Conference, 2008. SPLC '08. 12th International*, pages 67–76, Sept 2008.
3. Maximiliano Arias, Alan De Renzis, Agustina Buccella, , Alejandra Cechich, and Andres Flores. Búsqueda de servicios para asistir en el desarrollo de una línea de productos de software. In *Proceedings of ASSE 2015*, Rosario, Argentina, 2015.
4. Noor Hasrina Bakar, Zarinah M. Kasirun, and Norsaremah Salleh. Feature extraction approaches from natural language requirements for reuse in software product lines: A systematic literature review. *Journal of Systems and Software*, 106:132 – 149, 2015.
5. Günter Böckle, Frank J van der Linden, and Klaus Pohl. *Software product line engineering: foundations, principles and techniques*. Springer Science & Business Media, 2005.
6. Agustina Buccella, Alejandra Cechich, Maximiliano Arias, Matias Pol'La, Maria del Socorro Doldan, and Enrique Morsan. Towards systematic software reuse of gis: Insights from a case study. *Computers & Geosciences*, 54:9–20, 2013.
7. Agustina Buccella, Alejandra Cechich, Matias Polla, Maximiliano Arias, Maria del Socorro Doldan, and Enrique Morsan. Marine ecology service reuse through taxonomy-oriented {SPL} development. *Computers & Geosciences*, 73(0):108 – 121, 2014.
8. Kun Chen, Wei Zhang, Haiyan Zhao, and Hong Mei. An approach to constructing feature models based on requirements clustering. In *13th IEEE International Conference on Requirements Engineering (RE'05)*, pages 31–40, Aug 2005.
9. P. Clements and L. Northrop. *Software Product Lines: Practices and Patterns*. Addison-Wesley, 2002.
10. Orlena CZ Gotel and Anthony CW Finkelstein. An analysis of the requirements traceability problem. In *Requirements Engineering, 1994., Proceedings of the First International Conference on*, pages 94–101. IEEE, 1994.
11. George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, 1995.
12. Y. Mu, Y. Wang, and J. Guo. Extracting software functional requirements from free text documents. In *Information and Multimedia Technology, 2009. ICIMT '09. International Conference on*, pages 194–198, Dec 2009.
13. Kristina Toutanova and Christopher D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, EMNLP '00*, pages 63–70, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

Definición de una Herramienta de Calificación y Priorización de Nuevos Requerimientos de Software por medio de la Valoración de los Clientes

Sara De Federico¹, Javier Gago¹, Mariela Avogradini¹,
Lucía Melfi¹, Silvio Gonnet²

¹ Departamento Ingeniería en Sistemas de Información, Universidad Tecnológica Nacional Facultad Regional Rosario, Zeballos 1341. 2000 Rosario, Santa Fe, Argentina
{sdefederico, jgago, mavogradini, lmelfi}@frro.utn.edu.ar
<http://www.frro.utn.edu.ar/contenido.php?cont=604&subc=26>

² INGAR Instituto de Desarrollo y Diseño, Conicet, UTN, Avellaneda 3657
S3002GJC Santa Fe, Argentina
sgonnet@santafe-conicet.gov.ar,
<http://www.ingar.santafe-conicet.gov.ar/institucional/personal/investigadores/silvio-gonnet/>

Resumen. Un producto software en etapa de mantenimiento debe contemplar la incorporación de nuevos requerimientos, permitiendo su actualización e incorporación de características que surgen de las nuevas exigencias del mercado. Pero cuando distintos clientes solicitan cambios en forma simultánea, es primordial el orden en el que se seleccionarán para ser desarrollados e implementados. De un estudio realizado sobre información de empresas de la ciudad de Rosario y alrededores, se elaboró una metodología para la clasificación y priorización de nuevos requerimientos. Ésta complementa el uso de métricas de priorización con un análisis del impacto de la implementación de nuevos requerimientos en el sistema, en los clientes y en la empresa desarrolladora. Es necesario considerar minuciosamente el perfil del usuario solicitante, la viabilidad y conveniencia presupuestaria y de negocio. Este trabajo muestra la estructura de una herramienta de priorización y selección de nuevos requerimientos en software, detallando el proceso de calificación de clientes.

Palabras clave: Herramienta, valoración clientes, selección, priorización, nuevos requerimientos

1 Introducción

Un producto de software en etapa de versionado, utilizado por gran cantidad de usuarios, supone un trabajo importante de mantenimiento. Es fundamental que la incorporación de nuevos requerimientos sea estudiada cuidadosamente. Esto se debe a que no solo se deben contemplar los beneficios de una buena negociación [1], para una correcta valorización [2], sino que en esta etapa del ciclo de vida las variables y aspectos de la incorporación de un nuevo requerimiento se multiplican y se dispersan hacia otros sec-

tores de la empresa desarrolladora [3]. Estas consideraciones son esenciales para garantizar la continuidad de los estándares de calidad y performance [4] [5] [6]. La investigación realizada comprendió la recopilación y el análisis de la información sobre las prácticas que se realizan en empresas desarrolladoras de software de la ciudad de Rosario y alrededores. Se obtuvo una metodología para la calificación y priorización de nuevos requerimientos a implementar en un producto de software [7]. En dicha metodología, se hace especial énfasis en la preservación de los datos concernientes a los clientes del producto de software y sus peticiones de nuevos requerimientos. Actualmente en un nuevo proyecto se está comenzando la construcción en un entorno ágil, de una herramienta informática basada en dicha metodología, Esta herramienta proveerá a las empresas desarrolladoras una gestión de apoyo gerencial para la calificación y priorización de nuevos requerimientos. En este trabajo se presenta en la sección 2 la metodología de calificación y priorización de nuevos requerimientos obtenida como resultado de la investigación sobre las empresas de Rosario y alrededores. En la sección 3 se muestran los procesos de la metodología, en la sección 4 se muestra detalles del proceso de calificación de los clientes de una empresa desarrolladora y algunos prototipos obtenidos para las interfaces de dicho proceso. Finalmente, en la sección 5 y 6 se presentan las líneas de trabajo futuras, las conclusiones y recomendaciones sobre los temas expuestos.

2 Definición de la metodología de calificación y priorización de nuevos requerimientos

El estudio inicial fue realizado a las empresas desarrolladoras de software de la ciudad de Rosario y alrededores, y entre las cuales se encuentran las empresas más influyentes y de peso del mercado. Se recolectaron datos sobre los métodos y prácticas para la calificación y priorización de nuevos requerimientos, y los posibles problemas que puedan existir en la realización de este proceso. Luego se realizó un análisis de la información obtenida. Los resultados muestran inconvenientes en la calificación y priorización de nuevos requerimientos, como la ubicación de datos dispersa por toda la empresa y la intervención de diferentes departamentos y secciones en el proceso de incorporación de un nuevo requerimiento. Además se detecta un uso esporádico y discontinuo de métricas. No existe un seguimiento consistente del desarrollo y la implementación del nuevo requerimiento, y no se hacen mediciones posteriores del impacto de sobre la empresa, el sistema y la comunidad de usuarios. Generalmente se prioriza en función de la *criticidad* del requerimiento, que es definida por los Team Leaders o Project Managers. Los procedimientos para la priorización son personalizados y no estandarizados. Las técnicas de priorización son poderosas, pero todas requieren una calificación previa de cada requerimiento. Muchas de ellas obvian datos importantes como el estudio preliminar de los clientes solicitantes. Algunas empresas usan algún tipo de mecanismo para obtener información sobre la implementación del nuevo requerimiento, y suelen producirse problemas relacionados con la implementación, por ejemplo el *no uso de la nueva implementación*, la *no aceptación del presupuesto*, y el *desarrollo interrumpido del requerimiento*. La gran diversificación de métodos de clasificación indica la posible inexistencia de un nexo unificador de procedimientos

2.1 Metodología obtenida

Con toda la información recolectada se procedió a la generación de una nueva metodología para la calificación y priorización de nuevos requerimientos a incorporar en un software en etapa de mantenimiento [8]. La inclusión de estos pasos garantiza el cubrimiento de todos los aspectos que involucran el proceso completo de calificación y priorización. Los pasos que la componen se describen someramente a continuación:

1. Creación de un registro histórico de los clientes de un producto: recolección de toda la información relacionada con la interacción de los clientes con las distintas áreas de la empresa desarrolladora.
2. Calificación de los clientes solicitantes a partir del historial reunido: este registro de todas las actividades de los clientes con la empresa permitirá hacer una calificación de los mismos.
3. Precalificación de un nuevo requerimiento ingresado en función del cliente que lo solicitó: la calificación del cliente es considerada para calificar los requerimientos que éste solicita.
4. Calificación final del mismo mediante un análisis previo de su costo, impacto en el sistema y posibles consecuencias de su implementación: el mismo registro permite hacer un análisis más profundo de un requerimiento por comparación de presupuestos y desarrollos de otros similares.
5. Priorización de los requerimientos calificados previamente: se pueden utilizar diversas métricas escogidas en función de distintos intereses gerenciales o reglas de negocio.

Todos los pasos descriptos se resumen en diagramas de procesos con tareas secuenciales. Los procesos definidos en la metodología son: *Sección Clientes*, *Sección Requerimiento* y *Sección Priorización* [8].

3 Construcción de una herramienta de calificación y priorización de nuevos requerimientos

A partir de la metodología antes descrita, y siendo uno de los objetivos de un nuevo proyecto de desarrollo, este año se comenzó con la construcción de una herramienta de apoyo a las decisiones para la calificación y priorización de nuevos requerimientos de software. Esta herramienta podrá servir de apoyo a la toma de decisiones en las reuniones entre Team Leaders o Project Managers, para la determinación de las próximas tareas a realizar. En estas reuniones, la importancia que se les da a los requerimientos está ligada a la experiencia profesional de los mismos. Es así como el Team Leader trata de priorizar los requerimientos que encajen mejor en los desarrollos actuales, o que requieran menos recursos de personal o programación, el Project Manager prefiere los requerimientos que cree agregarán valor al producto, o preferirá trabajar con el cliente al que tiene mejor calificado por diversas razones.

Mientras tanto las otras áreas de la empresa enfocan su calificación de los clientes según su comportamiento financiero, comercial o personal. Muchas veces se debe elegir un nuevo requerimiento sin tener más ayuda que las opiniones que les van exponiendo los referentes de las diferentes secciones en forma informal e indocumentada. La metodología descrita en la sección 2, está basada en información obtenida en parte de esas reuniones, se plasmó en un conjunto de diagramas de procesos que engloban las tareas para la calificación de requerimientos teniendo múltiples aspectos y variables, y luego priorizar con las técnicas de priorización más conocidas y probadas [8]. Esto garantiza que la herramienta proveerá medios para la inclusión de los criterios de calificación y teniendo en cuenta los lineamientos de uso probados globalmente [9] [10].

La construcción de la herramienta se está realizando en el marco de un convenio de trabajo conjunto entre la Universidad Tecnológica Nacional Facultad Regional Rosario y la empresa TECSO Cooperativa de Trabajo Ltda. [11], empresa desarrolladora de software oriunda de la ciudad de Rosario, cuyo valor más importante reside en la práctica del cooperativismo como recurso de trabajo. Tecso brindará sus instalaciones para uso y capacitación de los integrantes y alumnos que participen del proyecto. Se requiere una pronta elaboración de prototipos funcionales para obtener un rápido feedback. El contexto de desarrollo será un entorno ágil, utilizando metodología Scrum, por lo que de cada uno de los procesos se extraerán las especificaciones de tareas para la organización de los sprints (organización iterativa del desarrollo), sin mucha documentación estructurada y con negociación permanente de los cambios. El equipo, con la colaboración de las empresas colaboradoras, realizará el análisis, definición y especificación de requerimientos con el objetivo de alcanzar una versión operativa del producto de valor agregado para el negocio [12]. Los procesos definidos se detallan, con una descripción de las tareas que realizará la herramienta a un nivel más extenso que el necesario para la realización de los sprints. Se pondrá especial énfasis en la sección de calificación previa de los requerimientos.

3.1 Proceso Sección Clientes

Cuando un cliente de la empresa solicita una nueva funcionalidad para el producto de software del cual es usuario, su pedido no es un hecho aislado. Esto se debe a que existe una amplia cantidad de información sobre la figura del cliente y su interacción y conexión con el producto o con la empresa. El proceso Sección Clientes muestra las tareas que se deben realizar para la recolección de la información del cliente, que se encuentra dispersa a lo largo de toda la organización. Se puede obtener información del cliente en su interacción con las áreas de Soporte a través de consultas y pedidos, Diseño y Desarrollo por su comportamiento ante requerimientos anteriores, Contabilidad por su actuar financiero, Gerencia por su relación de negocio, etc.

La Sección Clientes comprende la unificación y volcado de estos datos, implicando la transformación y transcripción de textos, importación desde gestores y bases de datos, y otros mecanismos de manipulación de datos. Se construyen *Repositorios Provisorios* para trabajar con los datos, y se rastrea todo tipo de información que pueda haber lo largo de todas las áreas de la empresa. Las áreas que fueron identificadas en la in-

investigación son: *Contabilidad, Desarrollo, Soporte, Gerencia*, pero a la hora de la implementación cada empresa tendrá una distribución particular, por lo tanto este proceso debe ser personalizado y ejecutado en función de las ubicaciones reales. La recopilación de la información se vuelca en un Repositorio principal llamado *Historial de Clientes*. En la Figura 1 se puede observar su estructura.

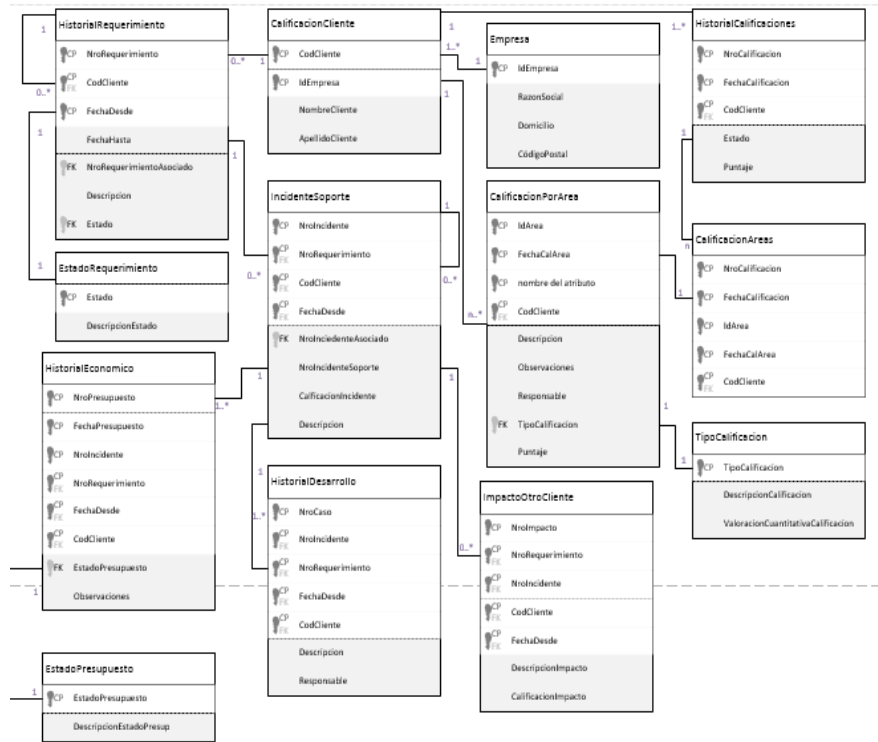


Fig. 1. Estructura del Historial de Clientes

En el *Historial de Clientes* se almacena cada uno de los incidentes de Soporte y la historia del desarrollo de un requerimiento desde su solicitud inicial, unido al cliente que lo solicitó y la historia contable que tuvo esa solicitud, desde el primer presupuesto hasta el pago final.

A la información recopilada se añaden un conjunto de valoraciones del cliente obtenidas de cuestionarios y tablas de valoración que se enviarán a los referentes de cada área. Las áreas pueden observar la información de los clientes para contestar los cuestionarios. Estas valoraciones son temporales y clasificadas por área, constituyen un conjunto ordenado. Además, se obtiene una calificación para un rango de fechas solicitado a través de un método, como por ejemplo el promedio móvil ponderado, y se almacena. El tratamiento de las calificaciones de los clientes permitirá a futuro un análisis de comportamiento de los clientes. En la sección 4 se mostrarán detalles de las valoraciones y calificaciones de los clientes.

3.2 Proceso Sección Requerimiento

La Sección Requerimiento es el proceso de calificación del nuevo requerimiento solicitado. La primera calificación, que puede considerarse *a priori*, se obtiene a partir de la calificación del cliente que lo solicitó. Además se deben considerar todas las variables que serían afectadas si se eligiera implementar la nueva solicitud, y analizar las posibles consecuencias e impacto de esta acción. Primero se precalifica el requerimiento y se hace el pedido a las áreas de una opinión sobre el nuevo requerimiento. La información de *Historial Clientes* muestra los datos de desarrollos anteriores solicitados, los incidentes de soporte, todos relacionados con el cliente. También las áreas pueden usar sus propios repositorios para hacer otras comparaciones, como por ejemplo buscar similitudes con otros casos. Finalmente, con el conjunto de calificaciones parciales del requerimiento, se obtiene una *Calificación Final del Requerimiento*.

Entre las situaciones que se deben contemplar las diferentes áreas se encuentran:

- *Análisis de Factibilidad Financiera*: La Sección Contabilidad crea una estimación del presupuesto, junto con el perfil financiero del cliente (calificaciones bancarias, Veraz, etc.), análisis de las posibilidades financieras que posee actualmente el cliente para afrontar el costo del requerimiento.
- *Análisis del Impacto sobre los demás clientes*: Desde Gerencia se analiza el impacto que generaría la implementación del requerimiento sobre los otros clientes, basándose en el Perfil personal del cliente.
- *Estimación de cambios en el código*: La Sección Desarrollo hace una estimación previa de los cambios que se deberán efectuar en el código, teniendo en cuenta los objetos disponibles actualmente, y la información de requerimientos anteriores solicitados por el cliente. Esta última documentación muestra el comportamiento con respecto a los objetos e interfaces de solicitudes anteriores.
- *Análisis del Impacto sobre el sistema*: En función de la estimación de los cambios en el código, la Sección Desarrollo realiza un Análisis del Impacto sobre el código en caso de desarrollar el requerimiento.

Teniendo las calificaciones parciales y los análisis realizados en las áreas y junto a la calificación del cliente, según los criterios de la empresa, la herramienta puede proveer informes para que Project Managers y Team Leaders utilicen la información para apoyo a la decisión de calificar el requerimiento. También se podrá hacer una calificación directa los nuevos requerimientos a partir de la calificación de los clientes. Por último, la calificación obtenida se guarda en el repositorio Pool de Requerimientos Calificados para su uso en la sección siguiente.

3.3 Proceso Sección Priorización

La última sección de la herramienta es la priorización de los nuevos requerimientos. El proceso consiste en la evaluación y priorización de los nuevos requerimientos pendientes de desarrollo. El sistema se encargará de crear un orden de prioridades utilizando la calificación obtenida en el proceso anterior. Previamente, para el correcto funcionamiento de la priorización, se deben haber establecido un conjunto de reglas que determinarán los criterios de priorización, es decir, dimensiones o características (forma de

pago, cumplimiento, tamaño de requerimiento, etc.) y la importancia que tienen cada una de éstas. La aplicación de éstas en la calificación quedará reflejada en el ordenamiento de los nuevos requerimientos. Las reglas y parámetros descritos previamente quedarán combinadas con el conjunto de requerimientos pendientes de desarrollo. Éstos estarán ingresados previamente al sistema con la calificación de las distintas dimensiones que le correspondan según la naturaleza de los mismos. Periódicamente, se deberá ejecutar el proceso con el que, junto a la definición de las metas estratégicas de la empresa, se podrá determinar una lista priorizada de requerimientos a desarrollar. Esta lista brindará un marco de soporte en la toma de decisiones al momento de definir si el equipo de desarrollo toma un requerimiento u otro. Dentro del proceso se encontrará un *Pool de Técnicas de Priorización*, en donde estarán disponibles los métodos de priorización más probados y considerados importantes por la comunidad de desarrollo de software [16]. Se podrá hacer diferentes priorizaciones según distintas técnicas. Además, la lista de requerimientos calificados podrá estar sujeta al análisis que realizaría Gerencia para un control de cumplimiento de las reglas de negocio deseadas.

4 Detalles del proceso de calificación del Cliente

En esta sección se muestran algunos detalles del complejo proceso de calificación de un cliente de la empresa. Es importante la valoración de cada área involucrada de alguna forma con los procesos de implementación de un nuevo requerimiento. Además, es fundamental la *temporalidad* de las valoraciones de cada una de las áreas. Cada cierto período de tiempo, un representante de un área abre una interface de valoración de un cliente específico, y completa una valoración que será guardada junto con la fecha de realización. En la Figuras 2 a 4 se observan prototipos de interfaces de valoración para las áreas Soporte, Contabilidad y Gerencia.

SECCIÓN CLIENTES

CLASIFICACIÓN CLIENTE A NIVEL SOPORTE

Nombre:

Número:

Ver Datos Completos

VALORACIÓN SOPORTE

	Pésimo	Malo	Regular	Bueno	Excelente
Trato al Personal	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Comunicación	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Nivel de Ansiedad	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Accesibilidad	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mediación	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Apreciación Personal	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fig. 2. Prototipo Valoración del Cliente según Área Soporte

SECCIÓN CLIENTES

CLASIFICACIÓN CLIENTE A NIVEL GERENCIAL

Nombre

Número

Ver Datos Completos

Ver Evauación Financiera

VALORACIÓN PERSONAL

	Muy Baja	Baja	Media	Alta	Muy Alta
Relación Personal	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Nuestra Recomendación	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Antgüedad	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Importancia	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Nivel de Retención	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Jerarquía en el Mercado	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fig. 3. Prototipo Valoración del Cliente según Gerencia

SECCIÓN CLIENTES

CLASIFICACIÓN CLIENTE A NIVEL CONTABLE

Nombre

Número

Ver Datos Completos

Crear Gráficos Estadísticos

Scoring Financiero

DESEMPEÑO FINANCIERO

	Pésimo	Malo	Regular	Bueno	Excelente
Productividad	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Rentabilidad	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Frecuencia de Participación	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Liquidez	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Morosidad	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Nivel de Presupuestos	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ranking en el Mercado	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ranking Interno	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fig. 4. Prototipo Valoración del Cliente según Área Contabilidad

4.1 Calificación Final de un cliente

Del conjunto de calificaciones establecidas por las áreas haciendo un promedio ponderado se genera una calificación final del cliente. La ponderación de cada calificación parcial puede determinarse según distintos criterios la gerencia. La variación de los índices de ponderación la recopilación de los datos brindará una forma de adecuar las calificaciones a las reglas de negocio de la empresa, y su supervisión evitará los sesgos por subjetividades o situaciones personales. La ecuación de la calificación por área se obtiene tomando i registros de calificación pertenecientes al rango de fechas solicitado ($fecha_in, fecha_fin$) siendo $fecha_fin$ la fecha actual. La *Calificación Final del Cliente* es un promedio ponderado móvil [13] de las k áreas, con ponderaciones p_i definidas por la gerencia:

$$C_{(área, fecha)} = \sum_{i=1}^n c_i \quad \cdot \quad C_{final} = \sum_{k=1}^n C_{(k, fecha)} \times p_k \quad (1)$$

La calificación final del cliente será primordial para la calificación de los requerimientos que él solicite, tal como se observará en la siguiente sección.

1. Definición de criterios para calificación de los clientes: Visión de las características del cliente desde el punto de vista de las áreas Finanzas, Gerencia, Soporte, Desarrollo. Apreciaciones basadas en la historia de las solicitudes anteriores.
2. Historial del cliente (para los requerimientos anteriores): Situaciones en el desarrollo. Conclusión exitosa o no del desarrollo. Problemas y cambios presupuestarios. Resultados del impacto sobre el sistema (código). Resultados del uso de los recursos de la empresa. Resultados del impacto sobre otros clientes. Mediciones de la satisfacción sobre la implementación. Otros datos relevantes
3. Calificación de los clientes en función de los puntos 1 y 2
4. Calificación de requerimientos en función de la historia del cliente
Priorización de los requerimientos: Uso de diferentes técnicas. Adecuación a las reglas de negocio (matrices de selección).

5 Líneas de trabajo futuras

El trabajo conjunto entre la universidad y las empresas que colaboran en el proyecto, será administrado por los Project Managers de la empresa anfitriona en coordinación con la dirección del proyecto. La tarea inicial será la recopilación de datos para la construcción del Historial de Clientes. Ya se han creado prototipos de varias interfaces, sobre todo en las tareas que pueden ser más complejas de comprensión. Se tiene programado comenzar con el inicio de los sprints desde agosto. Se han solicitado 3 alumnos becarios de investigación y 3 alumnos adscriptos para trabajar, que obtendrán formación profesional en un entorno real de desarrollo ágil. En todo momento se harán pruebas de los entregables para analizar posibles cambios que deban realizarse. La dirección del proyecto será el representante del cliente en los sprints, velando siempre por la concreción de una herramienta que cumpla con los objetivos del proyecto, y sea de utilidad real para las empresas. El desarrollo posee una base que al momento de la implementación será de libre uso para el público.

6 Conclusiones

La participación de las empresas desarrolladoras de software de la ciudad de Rosario y alrededores trabajando en conjunto con la Universidad Tecnológica Nacional Facultad Regional Rosario, y especialmente la colaboración y compromiso de la empresa Tecso Cooperativa de Trabajo Limitada, han posibilitado los medios para la construcción de una herramienta informática de calificación y priorización de nuevos requerimientos. Esta herramienta proveerá los medios para que en la calificación y priorización se consideren todos los aspectos que afectan a la empresa y permita que se respeten las reglas de negocio y los criterios y metas de la misma.

Además, el estudio de la priorización de nuevos requerimientos, provee una base para análisis posteriores y posibles investigaciones, que surgen gracias al conocimiento del tema adquirido durante la construcción de la herramienta. Estas nuevas tendencias de investigación aportan un valor agregado a los productos del proyecto actual.

Referencias

1. Sommerville I.: Software engineering 9th ed. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 201.
2. Berander, P., Andrews A.: Requirements Prioritization. In: Engineering and Managing Software Requirements. Aybüke Aurum · Claes Wohlin (Eds.). 1sted. Springer Verlag. 2005.
3. Bourque P., Farley R.: SWebok v3.0 Gude to the Software Engineering Book of Knowledge IEEE Computer Society Project. 2014.
4. Canfora G., Cimitile A., “Software Maintenance”. *University of Sannio, Faculty of Engineering at Benevento Palazzo Bosco Lucarelli, Piazza Roma 82100, Benevento Italy*, 29 November, 2000.
5. IEEE 12207-2008, “Systems and software engineering -- Software life cycle processes”, <http://standards.ieee.org/findstds/standard/12207-2008.html>
6. Erdil, K. et al: Software Maintenance As Part of the Software Life Cycle, Department of Computer Science Tufts University (2003)
7. De Federico S., Avogradini M., Sincosky N., Gago J., Moschetti D., Melfi L., Gonnet S. “Estrategia para la Calificación y Priorización de Nuevos Requerimientos de Software Propietario Utilizando un Registro de Historial de Clientes” CACIC 2015, Congreso Argentino de Ciencias de Computación, Buenos Aires, Argentina. ISBN: 978-987-3724-37-4 (2015) pps7624-7633.
8. De Federico S., Sincosky N., A., Avogradini M., Moschetti D., Melfi L., Gonnet S. “Definición de Procesos de una Herramienta para la Calificación y Priorización de Nuevos Requerimientos en Software Propietario”, Actas del 3º Congreso Nacional de Ingeniería Informática/Sistemas de Información 3º CoNaIISI Vol 1.. ISBN: 978-987-1896-47-9 (2015)
9. Aksyonov K., Spitsina I., Aksyonova O., Schaible P.: The state analysis of intelligent tools of Computer Aided Software Engineering, 24th International Crimean Conference of Microwave & Telecommunication Technology CriMiCo (2014)
10. Pohl K., “*Requirements Engineering. Fundamentals, Principles, and Techniques*” Springer (2010).
11. Tecso Cooperativa de Trabajo Limitada <http://www.tecso.coop/es>
12. Scrum, Improving the Profession of Software Development. <https://www.scrum.org/>
13. Triola M. Estadística. Pearson. 10th edn. (2012)

Metodologías para el desarrollo de software en PYMES

Marcelo Alberto Colombani¹, Martín Mauricio Pérez¹, Marcelo Alejandro Falappa²

¹ Facultad de Ciencias de la Administración
Universidad Nacional de Entre Ríos
marcelocolombani@hotmail.com, martinmperez@gmail.com

² Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur
mfalappa@gmail.com

Abstract. En este trabajo se presenta un estudio de los diferentes modelos de proceso de software, desarrollando sus características, bondades y deficiencias, con el fin de detectar qué metodología se encuadra o adapta mejor al escenario de las Pequeñas y Medianas Empresas productoras de software o empresas que, en su estructura, albergan un equipo de desarrollo no demasiado numeroso. Este trabajo pretende establecer guías para que dichas empresas puedan llevar adelante un proceso de desarrollo organizado, confiable y de calidad, en ambientes de requerimientos confusos, incompletos y cambiantes. Como parte del trabajo se realizó una experiencia práctica proponiendo la utilización de la Metodología XP (*eXtreme Programming*) con ciertas adaptaciones.

Keywords: PYMES, requerimientos, proceso de desarrollo de software, modelos de proceso de software, modelos y guías de certificación.

1 Introducción

El objetivo de este trabajo es detectar qué metodología se encuadra o adapta mejor al escenario de las Pequeñas y Medianas Empresas (PYMES) productoras de software o empresas que, en su estructura, albergan un equipo de desarrollo no demasiado numeroso y puedan llevar adelante un proceso de desarrollo organizado, confiable y de calidad, en ambientes de requerimientos confusos, incompletos y cambiantes.

Si bien existen trabajos relacionados a la utilización de metodologías en PYMES, el presente artículo se enfoca en la necesidad de contemplar diferentes aspectos relacionados a los requerimientos, y la posibilidad de que éstos reflejen las necesidades reales de los usuarios para lograr el producto correcto.

El artículo está organizado en varias secciones. La Sección 2 pone en contexto la problemática, la Sección 3 detalla las diferentes metodologías analizadas, desarrollando un ejemplo de la metodología XP y puntualizando cuestiones que se deben tener en cuenta al aplicar dicha metodología. La Sección 4 presenta los resultados obtenidos y la Sección 5 presenta las conclusiones y el trabajo futuro.

2 Situación Problemática

La búsqueda de técnicas que mejoren la calidad y permitan reducir los costos de las soluciones basadas en computadoras ha sido uno de los objetivos más perseguidos desde el inicio de la industria del software. La utilización masiva de las computadoras y, por ende, de los sistemas informáticos en los negocios ha generado una cultura de “sistematización de los procesos”. Sumado a esto, las herramientas de desarrollo han permitido la generación de software de una manera mucho más sencilla, rápida y menos costosa, lo que ha sustentado la cultura de la sistematización. Así se logró insertar sistemas software en empresas o negocios pequeños, que de otra manera no hubiesen podido encarar un proyecto de este tipo. A raíz de esto cada vez se requieren más y mejores sistemas y en tiempos considerablemente menores, aunque la construcción de software no es tarea sencilla por muy variados motivos. Normalmente se requiere que el software cubra todas las necesidades de la empresa, y que se desarrolle en un tiempo limitado. Además, existen innumerables metodologías de desarrollo de software que sustentan este proceso de construcción, lo cual complica la decisión de qué metodología utilizar en cada caso. Existen diferentes propuestas ortodoxas [1, 2] que se dirigen especialmente en el control del proceso, definiendo rigurosamente las actividades a desarrollar, los resultados a producir, las notaciones, y las herramientas a utilizar. Como alternativa para mejorar el desarrollo se pueden incluir más actividades, más diagramas [2], pero esto sólo puede llevar a un desarrollo más complejo y a sobrecargar al equipo de trabajo.

Una alternativa a este proceso de desarrollo se basa en centrarse en otros factores, como son el humano o el producto software a construir. Esta idea se conoce como Metodologías Ágiles [3], que dan mayor valor al individuo, mayor participación al cliente, y tienen como objetivo realizar entregas incrementales del software. Estos métodos son utilizados en proyectos en los cuales los requerimientos o expectativas del cliente/usuario son cambiantes o difusos. También suele exigirse un tiempo de entrega reducido con un producto que respete estándares de calidad [2].

En experiencias en desarrollo de software surge una problemática recurrente: la falta de adecuación de los requerimientos establecidos de un software a desarrollar con las expectativas que el cliente/usuario tiene del mismo [4, 2]. Esto tiene su origen en que los métodos tradicionales exigen que los requerimientos sean establecidos rigurosamente, en contraposición con las expectativas del cliente/usuario, que son dinámicas a lo largo del proyecto. De hecho, tales expectativas se modifican a medida que se visualizan las potencialidades del software desarrollado en las entregas parciales del mismo; entregas que, por otro lado, son necesarias para verificar tempranamente el producto de software y renovar el compromiso hacia el proyecto [1, 3, 5]. A medida que el proyecto se desarrolla, es posible que otros interesados (*stakeholders*) intervengan en el mismo, y vean potencialidades en el dominio de aplicación no contempladas en los requerimientos. En este escenario, es habitual que la empresa cliente vea que algunas expectativas no se cumplen con la versión final del producto, por más que éste cumpla plenamente con los requerimientos. Por ende, resulta necesario para la aceptación del producto (y la impresión positiva del equipo de desarrollo) reflejar estas expectativas no cubiertas en los requerimientos de alguna manera, para que se materialicen en la versión final del producto.

Esta problemática ha llevado a la necesidad de utilizar diferentes modelos de proceso de software adaptados y combinados de diferente manera para lograr un proceso óptimo sin perder de vista uno de los objetivos que es lograr un software de calidad en todos los aspectos, y confiable en todas sus dimensiones tanto en disponibilidad, fiabilidad, seguridad y protección. Aún se está en un proceso de maduración, y cada método se ajusta mejor a algún tipo de sistema. Por otro lado, no existen proyectos iguales, con lo cual un proceso que funcionó para desarrollar un sistema puede llegar a encontrar dificultades en otro. El desafío para los próximos años será definir y *consensuar* un modelo estándar que permita llevar adelante el proceso de desarrollo de una manera eficiente y ordenada.

Los modelos son simplificaciones de los procesos de software y, por lo tanto, un modelo de procesos de software es una simplificación o abstracción de un proceso real. Se puede definir a un modelo de procesos del software como una representación abstracta de alto nivel de un proceso de software [2]. Revisando los enfoques propuestos [1, 2, 5, 6] es posible identificar una variedad de modelos de desarrollo de software, como así también modelos similares mencionados de diferente manera, pero realizando un análisis minucioso de su conformación y manera de organizar el proceso, éstos realmente reflejan el mismo funcionamiento.

Cada modelo es una descripción de un proceso de software que se presenta desde una perspectiva particular [2], describiendo una sucesión de fases y una relación entre ellas. Según las fases y el modo en que se produzca esa relación, tenemos diferentes modelos de proceso. Es por ello que un modelo es más adecuado que otro para desarrollar un proyecto dependiendo de un conjunto de características de éste [1, 2]. Análogamente, las versiones entregables de cada etapa varían conforme su tasa de cambio y validación; en el caso que se acepte la dinámica de requerimientos como una suposición presente en el modelo, estos cambios deben documentarse y trazarse adecuadamente, para justificar posibles desviaciones no contempladas en el proyecto.

Los procesos del software que se basan en el enfoque en cascada se siguen utilizando para el desarrollo de software, particularmente cuando éste es parte de grandes proyectos de ingeniería de sistemas. El modelo permite que se produzca la documentación necesaria en cada etapa, haciendo posible un seguimiento del proyecto. Esto provoca una inflexibilidad al dividir el proyecto en etapas, ya que se hacen compromisos en las etapas iniciales, impidiendo los cambios en los requerimientos del cliente una vez que se comienza el desarrollo. El modelo en cascada se debe utilizar cuando los requerimientos se comprendan bien y sea improbable que cambien radicalmente durante el desarrollo del sistema. Es difícil utilizarlo en empresas pequeñas que requieren rápidas respuestas con costos reducidos.

Por otro lado, la esencia de los procesos iterativos es que la especificación se desarrolla junto con el software. Sin embargo, esto crea conflictos con el modelo de obtención de muchas organizaciones, donde la especificación completa del sistema es parte del contrato de desarrollo del mismo. En el enfoque incremental, no existe una especificación completa del sistema hasta que el incremento final se especifica. Esto requiere un nuevo tipo de contrato, que a los clientes grandes como las dependencias del gobierno les puede resultar difícil de incorporar [2].

Se ha desarrollado una variante de este enfoque incremental denominada Metodologías Ágiles. Éstas se basan en el desarrollo y la entrega de incrementos de

funcionalidad muy pequeños, en la participación del cliente en el proceso, en la mejora constante del código y en la programación por parejas. Pero en esta metodología de trabajo también es difícil estimar el costo de un proyecto, dado que el alcance del mismo no está completamente definido al comienzo, y los cambios durante todo el proceso son una característica del modelo.

Todos los métodos tienen limitaciones, y los métodos ágiles son apropiados para algunos tipos de desarrollo de sistemas. Son los más idóneos para el desarrollo de sistemas de negocio pequeños y de tamaño medio, y para el desarrollo de productos para computadoras personales. No parecen ser adecuados para el desarrollo de sistemas de gran escala con equipos de desarrollo ubicados en diferentes lugares y donde pueda haber complejas interacciones con otros sistemas hardware o software. Existe la posibilidad de que, si un proyecto es grande, puede ser dividido en módulos más pequeños, lo suficientemente independientes y con una orientación a servicios se pueden crear equipos ágiles que atiendan el desarrollo de los módulos. No se deben utilizar métodos ágiles para el desarrollo de sistemas críticos en los que es necesario un análisis detallado de todos los requerimientos del sistema para comprender sus implicaciones de seguridad o protección.

En las metodologías ágiles se ve reflejada la manera de llevar adelante los proyectos de software para PYMES. Se aprecia una alta similitud en cuanto a los principios del manifiesto ágil [7], donde se plantea que la mayor prioridad es satisfacer al cliente a través de entregables tempranos y continuos, presentaciones frecuentes de software, requisitos y diseños emergen de equipos que lo organizan por sí mismos, participación activa de los involucrados, entre otros. El inconveniente de las metodologías ágiles puede pasar por la falta de documentación, lo cual podría llegar a afectar el mantenimiento del sistema a futuro, si éste debe ser realizado por una persona o grupo de personas que no estuvieron involucrados en el desarrollo del mismo. Otro problema está relacionado con la estructura general del sistema. La falta de un diseño global del sistema podría llegar a entorpecer las estructuras, generar estructuras de datos no normalizadas y código fuertemente acoplado, si no se logra tener en las primeras etapas una concepción global de lo que se va a realizar.

3 Aplicación de una Metodología en PYMES

A continuación, se presenta una metodología de desarrollo de software y un marco de trabajo pensado para las PYMES. Este tipo de empresas son especialmente sensibles al mal uso de los recursos, por lo que es importante que puedan contar con herramientas que les permitan administrar el proceso productivo y les ayuden a controlar el proceso de desarrollo y en consecuencia mejorar la calidad del software que desarrollan [8]. En particular, este trabajo se enfocó en empresas ubicadas en el área de influencia de la Facultad de Ciencias de la Administración de la Universidad Nacional de Entre Ríos con sede en Concordia en el marco del Proyecto de Investigación y Desarrollo PID-UNER 7049: “Modelos de Certificación para los procesos de Ingeniería de Software en productos desarrollados con Lenguajes de Programación *Open Source*: relevamiento y aplicación en PYMES de la zona de influencia de la UNER Concordia” [9].

La metodología que se expone en el trabajo se basa en el supuesto de que la mayoría de los departamentos o PYMES de la zona dedicados al desarrollo de software, se encuentran formados por un número pequeño de entre uno a quince integrantes, por lo que es muy probable que el proceso de desarrollo de software, desde el análisis de requerimientos hasta la implementación, sea realizado por las mismas personas que tomarán múltiples roles.

El ejemplo se desarrolló utilizando la metodología ágil XP, que fue puesta en práctica en una entidad pública. A través del ejemplo práctico se buscó proveer de una estructura capaz de orientar la aplicación de un proceso de software conocido, ya que la existencia de un modelo es el comienzo en los primeros pasos en la dirección al gerenciamiento y a la mejora del proceso de software.

La metodología XP consiste en una serie de prácticas y reglas que permiten a los programadores desarrollar software de una manera dinámica y ágil, con mínima documentación. En caso de que el cliente necesite documentación formal, se deberá adecuar la metodología para ir documentando a medida que se avanza en la construcción del software. Además, es recomendable para proyectos pequeños donde se involucre un número reducido de personas. Por último, es importante recordar que XP es adecuado en ambientes donde se producen rápidas y frecuentes modificaciones de los requerimientos [10].

El ejemplo se desarrolló describiendo las diferentes fases de XP, ejemplificando las mismas, y mostrando las tareas básicas a realizar en cada una. Como lo propone XP, se utilizaron algunas de las herramientas de software para apoyar el desarrollo del software, como SVN server para el versionado del código y la documentación, Junit para realizar las pruebas unitarias del código Java, y Visual Paradigm para documentar los diagramas UML.

3.1 Fase de Exploración

El objetivo de esta fase es entender lo que el sistema debe hacer, para poder estimar los tiempos del proyecto. Esta fase comienza describiendo las reglas de negocio, donde el usuario escribe las historias de usuario y el programador estima los tiempos de las historias [10].

Se realizó una primera reunión con el jefe del área de cada empresa y las personas responsables, lo cual permitió definir el proyecto, ver la dimensión del mismo, las relaciones existentes con otros sistemas, la cantidad de procesos de usuarios involucrados, los usuarios que intervienen en las operatorias, los controles generales que el sistema debe tener, y el tipo de interfaz que el usuario desea, entre otros puntos. En estas reuniones no es necesario que estén presentes todos los integrantes del equipo de desarrollo. Posteriormente se realizaron varias reuniones más a fin de recabar mayor información y centrarse en algunos de los puntos anteriores. En estas reuniones se utilizaron algunas de las técnicas de descubrimiento de requerimientos, como, por ejemplo, entrevistas, escenarios, casos de uso, historias de usuarios, etc.

Como parte de las tareas de los programadores, se redactaron las diferentes historias de usuarios, ya que fue difícil lograr que las redacten los mismos. Por tal motivo, se optó por dialogar con ellos e ir recabando la información y redactar así las diferentes historias de usuarios.

Como resultado de esta etapa se desarrolló un documento que contó con los objetivos del proyecto, las áreas involucradas, un resumen del circuito de trabajo general, sin entrar en detalles de cada una de las tareas, y las historias de usuarios. En proyectos de mayor complejidad se determinó que será necesario agregar un diagrama de casos de uso de los casos identificados, y un diagrama de actividad de los casos de uso más significativos. Otra de las actividades realizadas en las primeras reuniones de los programadores fue la definición de cuestiones técnicas del proyecto.

3.2 Fase de Planificación de Entregas

La siguiente fase fue la de planificación, cuyo objetivo fue definir las historias de usuario o funcionalidades a implementar en la primera iteración y el tiempo de cada una de ellas. Este cálculo se efectuó en base a las estimaciones que realizaran el cliente y los programadores, dándole puntos a cada historia en base a la experiencia de desarrollo en problemas similares. Los usuarios del sistema (clientes) decidieron cuáles historias de usuarios eran vitales para una primera versión, estableciendo una lista de prioridades de las historias.

Esta etapa se realizó en dos días. En general, si la etapa de exploración se realizó correctamente, unos pocos días deben ser suficientes. Como resultado de esta etapa se estableció que la primera iteración se debía realizar en un mes. Se sugiere agregar al documento de proyecto la información que va surgiendo en cada fase con el título “Número de planificación de entrega”.

3.3 Iteraciones

A partir de la definición realizada en la fase anterior, se comenzaron a producir las funcionalidades asignadas a la primera iteración. La recomendación que surge de la experiencia es la construcción de casos de uso para cada historia de usuario. Cuando el caso de uso poseía cierta complejidad, también se utilizaron diagramas de actividad. Si bien una de las fundamentaciones de XP es la de tener una documentación mínima, la que se utilizó se considera indispensable para una mejor comprensión de cada historia de usuario y para su posterior seguimiento.

Se definieron los modelos de datos necesarios para sustentar los programas realizados para cada una de las historias de usuario asignadas a la iteración. Se diseñaron y programaron las pruebas unitarias de los diferentes módulos. Además, se diseñaron pruebas manuales o test de aceptación a aplicar a las interfaces de usuarios, describiendo los pasos necesarios para completar las acciones y los resultados finales a obtener. Como tarea extra se podrían realizar pruebas de integración con herramientas de automatización.

Por cada historia de usuario se llevó a cabo la programación. Tomando en cuenta la recomendación de XP sobre la programación en parejas, el comienzo de cada historia de usuario se programó en grupos de dos programadores. Una vez que el trabajo estaba avanzado, se continuó programando individualmente. En esta etapa también surgieron casos de usos y problemas propios del proceso de desarrollo, como requerimientos no funcionales de seguridad, tiempo de respuesta, interfaces de

usuarios, etc. que no fueron especificados en la fase de exploración. Estas tareas, que son propias de los programadores, se fueron organizando en las reuniones denominadas “reuniones de parado” (*stand-up meeting*).

Otra recomendación realizada fue la de publicar lo antes posible las nuevas versiones de código, aunque no sean las últimas, siempre que estén libres de errores. Idealmente, todos los días deben existir nuevas versiones publicadas. Todos los desarrolladores necesitan trabajar siempre con la “última versión”. Si bien la metodología XP, no plantea la utilización de un servidor de control de versiones, es aconsejable su utilización.

Además, a medida que se avanzó en el desarrollo del sistema, se fueron detectando requerimientos del sistema que no son parte del dominio de la aplicación específica, necesitando desarrollar funcionalidad de creación de usuarios, menú dinámico del sistema, perfiles de usuarios y auditorías. Es importante subrayar que no siempre los clientes o usuarios expresarán la necesidad de contar con este tipo de funcionalidad, pero es probable que una vez que se pruebe o se intente poner en producción las historias de usuarios programadas en la iteración surjan estas inquietudes. Por este motivo todos estos casos fueron registrados y puestos a consideración para la próxima planificación de iteración.

Otra situación identificada es la adecuación constante de las historias de usuario a medida que se avanza en el desarrollo y se recaba mayor información respecto al funcionamiento, los datos que se procesan y los controles que se realizan sobre los datos. Es frecuente encontrar que ciertos requerimientos o historias de usuario hayan sido planteados de una manera confusa e incompleta, y a medida que se trabaja en el código surgen inconsistencias entre lo programado y lo que el usuario realmente intentó plantear. Una premisa importante de la metodología XP es la idea de mantener una retroalimentación constante con el usuario. Se utilizó la primera iteración para desarrollar un prototipo del sistema, el entorno de trabajo y establecer los formatos de las interfaces.

3.4 Producción

La fase de producción implica trasladar el software generado al cliente. La actividad de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Si el sistema es muy grande se podrá comenzar a implementar por módulos de funcionamiento, pero si el problema se acota a un módulo de funcionamiento se deberá esperar a tener gran parte del sistema desarrollado. En el trabajo se optó por la alternativa de continuar el desarrollo y ejecución de nuevas iteraciones sin la implementación definitiva. Sólo se fueron realizando pruebas con los usuarios a fin de verificar la correctitud del producto desarrollado.

3.5 Mantenimiento

La fase de mantenimiento recién se pone en práctica una vez completada una fase de producción, teniendo una versión operativa que el cliente esté utilizando, ya que un

proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Es importante recalcar que una vez que se pone en marcha el proyecto se comienza a dar soporte a los usuarios del sistema, y la velocidad de desarrollo decae ya que parte del esfuerzo se pone en la solución de los inconvenientes.

4. Resultados Obtenidos

Como se mencionó en la Sección 3, la metodología propuesta se aplicó en una entidad pública de la ciudad de Concordia, que cuenta con un equipo de desarrollo integrado por 4 personas. El tiempo insumido por el proyecto fue de 5 meses.

Además del proyecto puntual tomado como ejemplo, la metodología XP se puso en práctica en otros 4 proyectos dentro de la misma institución. En proyectos anteriores se habían utilizado metodologías tradicionales, intentado abarcar la mayor cantidad de situaciones. El inconveniente detectado es el esfuerzo considerable que exigía la confección de la documentación y la gestión del proyecto, lo que generaba una sobrecarga innecesaria.

Teniendo en cuenta la experiencia realizada con la utilización de la metodología XP, se pudo apreciar que es una metodología centrada principalmente en la relación de las personas. A lo largo del trabajo se promovió el trabajo en equipo, el aprendizaje de cada uno de los desarrolladores, y siempre se generó un buen clima de trabajo. Siguiendo los pasos propuestos, se hizo hincapié en la satisfacción del usuario/cliente, y esto permitió a los desarrolladores responder a los cambios incluso en las etapas finales del proyecto. Esto permitió dar respuesta a problemas de cambios de requerimientos, ya que los usuarios no tenían una idea firme de lo que el sistema debía hacer, pero sí de su trabajo.

Los programadores no necesariamente tenían una experiencia considerable en la metodología XP, y se podría recalcar que para proyectos con requisitos dinámicos o de alto riesgo, es probable que un equipo pequeño de programadores sea más eficaz que un equipo grande. Resumiendo, XP propone una metodología ágil y concreta, aunque requiere de una nueva manera de pensar, ver y hacer las cosas, tanto por parte de los gerentes, como de los desarrolladores y también del cliente. La aplicabilidad de esta metodología, y de las adaptaciones propuestas, a cada proyecto en particular, debería ser analizada en cada caso, teniendo en cuenta el tamaño y tipo de proyecto, sus ventajas y desventajas.

A partir de la experiencia se están elaborando guías de diseño con la intervención de todos los programadores desarrolladores involucrados en los diferentes proyectos a fin de identificar fortalezas y debilidades de la metodología utilizada (XP), así como aquellas actividades no propias de la metodología pero que contribuyen fuertemente a mejorar el trabajo en equipo.

5. Conclusiones y Trabajos Futuros

En base a la investigación realizada se puede asegurar que los cambios son inevitables en todos los proyectos de software, sean éstos pequeños, medianos o grandes. Los requerimientos del sistema cambian cuando el negocio donde está inserto el sistema responde a las presiones externas.

Las prioridades de gestión cambian y, cuando se dispone de nuevas tecnologías, cambian los diseños y la implementación. Esto significa que el proceso del software no es un proceso único; más bien, las actividades del proceso se repiten regularmente conforme el sistema se rehace en respuesta a peticiones de cambios. Como surge del estudio de las diferentes metodologías, el problema de cada una de ellas siempre gira alrededor de la falta de una correcta definición de los requerimientos, de la gestión de la documentación y de la falta de comprensión del cliente con respecto a los requerimientos. Todos estos factores llevan a requerimientos confusos, incompletos y cambiantes.

Además de la metodología que se utilice, es importante definir una buena estructura de documentación, a fin de poder reflejar los requerimientos que dieron origen al sistema, los diferentes componentes del sistema software, sus relaciones y sus efectos en las modificaciones del mismo.

Otro aspecto que no debe dejarse de lado, es el aseguramiento de la calidad en el proceso y producto de software; en particular si es un requisito para certificar o validar conforme a estándares internacionales de calidad [11]. Como equipo (empresa) de desarrollo, resulta adecuado instaurar buenas prácticas en el proceso de desarrollo conforme a guías de certificación aceptadas: ISO/IEC 9001- 90003 [12, 13], ISO 15504 SPICE [14], ISO/IEC 15288 [4], ISO/IEC/IEEE 12207 y complementarias [15, 16]. Como empresa usuaria o productora de software sería importante verificar la calidad del producto conforme a esquemas de certificaciones ISO/IEC 25000 [17, 18, 19], permitiendo como efecto deseado tener ventajas competitivas y comerciales.

Cómo parte del trabajo se planteó el uso de la metodología XP para el desarrollo de software en las PYMES, dando como resultado un proceso de desarrollo de software confiable, que permite involucrar a todos los participantes, y que logra mantener los requerimientos con cambios previsibles y sin un impacto negativo en el resto del sistema ya desarrollado. Esto evita que los mismos se tornen confusos, y permite que se vayan alcanzando a medida que se avanza en el proyecto. En el caso de que algún requerimiento cambie, la metodología propuesta evita que este cambio impacte negativamente en todo el proyecto.

Se realizaron algunas adaptaciones a la metodología, a fin de incorporar documentación necesaria. Por ejemplo, se creó un documento de proyecto y se propuso la utilización de diagramas de casos de uso y actividad. Por otro lado, el abordaje del sistema completo al inicio permitió tener una noción general del sistema y así evitar los problemas mencionados.

Como trabajo futuro será importante continuar el estudio de los Modelos de Proceso de Software y describir pasos metodológicos para llevar adelante un proyecto de software exitoso utilizando alguna combinación de diferentes metodologías tradicionales y ágiles, ya que cada una posee ventajas y desventajas. Así se espera lograr un enfoque híbrido, en el cual los métodos ágiles incorporen algunas técnicas

de planificación y documentación, pero sin perder de vista las iteraciones y los principios del modelo ágil. También, se buscará ampliar el espectro de pruebas a más PYMES de la zona de influencia de la Facultad de Ciencias de la Administración de la Universidad Nacional de Entre Ríos con sede en Concordia.

Agradecimientos

El presente trabajo fue parcialmente financiado por el proyecto de investigación PID UNER 7049 titulado *“Guías para aplicación de Normas de Calidad para los procesos de Ingeniería de Software en productos desarrollados con Lenguajes de Programación Open Source: relevamiento y aplicación en PYMES de la zona de influencia de la UNER Concordia”*.

Referencias

1. Pressman, R. S., Maxim, B.: “Software Engineering: A Practitioner's Approach”. McGraw-Hill, 7ma Ed, 2014.
2. Sommerville, I.: “Software Engineering” (10th Ed). Pearson Addison Wesley, 2015.
3. Beedle, M.: “Principles behind the Agile Manifesto”. <http://www.agilemanifesto.org/principles.html>
4. ISO: “ISO/IEC 15288:2008 - Systems and software engineering - System life cycle processes.” Mar. 2008.
5. Wells, D.: “Extreme programming: A gentle introduction”. Tech. Report, 2009. <http://www.extremeprogramming.org>
6. Jacobson, I., Booch, G., Rumbaugh, J.: “El proceso unificado de desarrollo de software”. Addison-Wesley, primera edición, 1999.
7. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D.: “Manifiesto for Agile Software Development” (2001).
8. Pérez, I., Castelló, R.: “Columna Vertebral del Software, Una Metodología de Desarrollo Para PyMES”. ITESM Campus Chihuahua, México.
9. Colombani, M., Pérez, M., Pacifico, C.: “Metodologías para el desarrollo de software en PYMES”. Workshop de Investigadores en Ciencias de la Computación WICC (2016).
10. Bona, C., Thiry, M.: “Processo de Software: um Estudo de Caso em XP”. CACIC - Congreso Argentino de Ciencias de la Computación. 2003.
11. Schneider, F., Berenbach, B.: “A Literature Survey on International Standards for Systems Requirements Engineering”, Procedia Computer Science, Volume 16, 2013, Pages 796-805.
12. ISO: “ISO/IEC 9001:2015 - Quality management systems - Requirements” Sep, 2015.
13. ISO: “ISO/IEC 90003:2014 - Software engineering - Guidelines for the application of ISO 9001:2008 to computer software” Dec, 2014.
14. ISO: “ISO/IEC 15504 Information technology - Process assessment” ISO/IEC, Nov, 2004.
15. ISO: “ISO/IEC/IEEE 12207:08 Systems and software engineering - Software life cycle processes,” Mar. 2008.
16. ISO: “ISO/IEC/IEEE 29148: 2011 - Systems and software engineering - Life cycle processes - Requirements engineering,” Nov. 2011.

17. ISO: "ISO/IEC 25030 - Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Quality requirements," Jun. 2007.
18. ISO: "ISO/IEC 25040 – Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Evaluation process." Feb. 2011.
19. ISO: "ISO/IEC 25010 –Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models," Mar. 2011.

Evolución de F/OSS: estudio de caso múltiple sobre el crecimiento

Jorge Ramirez¹, Carina Reyes², María Laura Massé³, Nilsa Sarmiento⁴

^{1,2,3}Facultad de Ciencias Exactas, Universidad Nacional de Salta, Centro de Investigación y Desarrollo en Informática Aplicada – Avda. Bolivia 5150 – Salta

⁴ Instituto De Investigaciones En Energía No Convencionales (INENCO), Av. Bolivia 5150, Salta, Argentina

^{1,2,3}{[ramirezj_reyes_mlmassep](mailto:ramirezj_reyes_mlmassep@cidia.unsa.edu.ar)}@cidia.unsa.edu.ar, ⁴nilsamsarmiento@gmail.com

Abstract: la disponibilidad de datos propia de los proyectos de Software Libre y de Código Abierto (F/OSS por sus siglas en inglés) posibilita realizar investigaciones reproducibles, al tiempo que habilitan un tratamiento estadístico poco frecuente en la investigación en ingeniería de Software. En este trabajo analizamos el ritmo de crecimiento del tamaño de un grupo de productos F/OSS a lo largo de sucesivas versiones, e indagamos sobre la posible relación entre ese crecimiento y un indicador de acoplamiento.

Palabras Clave: Software Libre y de Código Abierto, Evolución del Software, Métricas

1 Introducción

El estudio del Software Libre y de Código Abierto (F/OSS por sus siglas en inglés) se ha convertido en un campo de investigación de interés creciente para la ingeniería de Software. La disponibilidad de información públicamente accesible sobre un gran número de proyectos de software ofrece la oportunidad de abordar investigaciones [1] [2] en las que los estudios pueden ser repetidos y comparados por diferentes investigadores [3].

La investigación sobre la evolución de software es uno de los tópicos sobre los que se ha producido mayor cantidad de trabajos a partir del estudio de F/OSS [4] [5] [6].

El conocimiento sobre la forma en que el software evoluciona y los factores que intervienen en ese proceso son relevantes a para orientar distintos aspectos atinentes a la administración de proyectos de software [7]. Los sistemas de software sufren cambios para poder seguir cumpliendo con las funciones para las que fueron creados. El estudio de las distintas versiones de software llevó a esos autores a elaborar una serie de afirmaciones conocidas como Leyes de la Evolución del Software.

Las investigaciones de Lehman y otros[8] , así como diversos trabajos posteriores [5] [9] [10] [11], pusieron de relieve la importancia de conocer las características generales de la evolución del software, así como los factores que inciden en ella. Lehman observó que los sistemas que modelan operaciones humanas deben

modificarse para contemplar nuevos requerimientos, cambios organizacionales y tecnológicos, entre otros; esto implica que sus funciones aumentan y con ello la posible complejidad de los sistemas de software, lo que dificultaría nuevos cambios.

La formulación de las leyes de la evolución fueron modificándose con el tiempo [12]. A medida que los enunciados fueron más precisos, fue posible realizar más investigación empírica sobre estos temas; no obstante, las limitaciones en cuanto al acceso a los datos y a la divulgación de los mismos, como las normas de confidencialidad, dificultan un abordaje estadístico más profundo. [12] [13]

Las leyes que se relacionan con el presente trabajo son: la II, de complejidad creciente, que sostiene que los sistemas evolutivos aumentan progresivamente su complejidad si no se adoptan medidas para mantenerla o reducirla; la VI, que afirma que el software debe aumentar sus funciones a lo largo del tiempo para seguir siendo útil; y las VII, que plantea que la calidad tiende a descender, a menos que se adapte y mantenga de acuerdo con el contexto en el que opera.

Teniendo en cuenta las observaciones de los párrafos anteriores, nos planteamos aquí explorar la evolución de un conjunto de proyectos de F/OSS en base a la información públicamente disponible en cualquier emprendimiento de ese tipo, es decir, el código fuente y la fecha de publicación de cada versión; esos datos posibilitan el análisis del código fuente mediante métricas de producto y la visualización de la evolución cotejando cambios entre una versión y otra.

Específicamente, analizamos en este trabajo la evolución de un conjunto de proyectos F/OSS a fin de comprobar:

- Si se observa que el aumento del tamaño del software tiende a ser menor en las sucesivas versiones, lo que podría relacionarse con el aumento de la complejidad a lo largo del proceso evolutivo (en consonancia con la ley II de la evolución)
- Si existen indicios de que el aumento del tamaño del software de una versión a la siguiente está relacionada con el acoplamiento entre clases.

El presente trabajo se organiza de la siguiente forma: en la sección siguiente se resumen trabajos relacionados con la temática abordada; a continuación, se describe la metodología utilizada, detallando las herramientas utilizadas para la extracción de métricas y el procesamiento de los datos; posteriormente, se presentan resumidamente los datos obtenidos; en la sección siguiente se discuten los resultados; finalmente, se exponen las conclusiones y se proponen trabajos a futuro.

2 Trabajos Relacionados

Como señalamos en la introducción, existe un cuerpo creciente de investigaciones sobre el F/OSS en los que se aborda la evolución del software y en particular las características del aumento del tamaño a lo largo de las diferentes versiones.

Godfrey y Tu [14] publicaron en 2000 una investigación sobre la evolución del núcleo de Linux, cuyos resultados contrastaban con las investigaciones realizadas por Lehman y otros en el ámbito del software privativo o tradicional. Estos autores comprobaron que el tamaño del núcleo del sistema operativo libre crece a tasas supralineales, lo que contradice el enunciado de la ley de Lehman y en particular los modelos propuestos por Turski [15] [16].

El trabajo de Godfrey y Tu fue repetido, con algunas variantes, por Robles y otros [17], extendiéndose la investigación a un conjunto amplio de proyectos de F/OSS [18] que posibilitan una visión más amplia.

Como señalamos en la introducción, las características del Software Libre han posibilitado un número creciente de investigaciones, muchas de ellas vinculadas con la evolución. Se han realizado numerosos estudios de caso [19] [20] [21], análisis comparativos de diferentes proyectos [13] [22], análisis de muestras amplias con procesamiento estadístico [23] [13], etc.

El trabajo actual retoma investigaciones anteriores, en particular el estudio de caso de la evolución del proyecto Sweet Home 3D [24] y el análisis de la evolución de un conjunto de indicadores de diseño en un grupo de proyectos de F/OSS [25]. Aunque la temática es distinta, el análisis estadístico que nos llevó a proponer umbrales para promedios de métricas en proyectos F/OSS [26] comparte con este trabajo un marco conceptual y algunas de las herramientas utilizadas.

3 Metodología

Nuestro objetivo en este trabajo apunta a conocer con mayor profundidad algunos aspectos de la evolución de un conjunto de productos de F/OSS. En ese sentido, el enfoque que adoptamos es el de realizar una serie de estudios de caso entendidos como un método de investigación de *“una instancia (o un pequeño número de instancias) de un fenómeno contemporáneo de ingeniería de software dentro de su contexto en la vida real”* [27] [28].

Se trata de un estudio de carácter exploratorio, donde buscamos obtener conclusiones preliminares susceptibles de profundizar y contrastar en estudios posteriores.

Para este trabajo analizamos todas las versiones publicadas y disponibles para un conjunto de 13 proyectos de software. La selección se limitó a desarrollos realizados en Java, a fin de descartar que el lenguaje de desarrollo pudiera incidir en los resultados.

Para cada versión publicada de cada proyecto se obtuvieron a partir del código fuente las siguientes métricas: el tamaño medido en líneas de código (LOC) y la dispersión de acoplamiento (DA), definida como la proporción de las llamadas a métodos diferentes (es decir, las llamadas a cada método se cuentan una sola vez)

respecto del total de métodos definidos en toda el código de la versión. Ambas métricas se extrajeron mediante la herramienta iPlasma [29], que puede descargarse de <http://loose.upt.ro/reengineering/research/iplasma>. La selección de la métrica de acoplamiento tuvo en cuenta su carácter global como atributo de diseño [30] , no dependiente del tamaño, siendo además una de las medidas que analizamos en trabajos anteriores [26].

No existe acuerdo en cuanto a las métricas que conviene utilizar en este tipo de investigaciones; no obstante cabe destacar que Lehman sugiere la utilización del número de secuencia de versión para analizar la evolución y que las métricas de tamaño SLOC (líneas de código fuente), LOC y cantidad de módulos parece exhibir una importante correlación entre ellas[31] [32] .

La información obtenida se organizó en planillas de cálculo mediante LibreOffice Calc, agregando a cada versión la fecha de publicación para poder considerar el intervalo entre cada versión y la siguiente.

La lista de proyectos analizados se detalla en la tabla 1, señalándose además el período de tiempo en que fueron publicadas las versiones estudiadas y la cantidad de versiones.

Tabla 1: Lista de proyectos analizados

Proyecto	Intervalo	Versiones
PDFBox	2010-2014	18
JFreeChart	2004-2009	20
Art Of illusions	2002-2010	28
FreeMind	2000-2014	41
ICEPDF	2009-2014	25
Lwjgl	2002-2013	49
Batik	2001-2008	19
Chemistry Development Kit	2007-2014	36
Vuze	2010-2015	28
Sweet Home 3D	2007-2016	36
HSQLDB	2001-2014	21
Dynamic Reports	2010-2015	48
Jxplorer	2002-2014	42

En la planilla de cálculo se agregó una columna con el número de días entre una versión y la anterior, y el aumento de tamaño en LOC por día; para cada proyecto se

calculó el coeficiente de Pearson a fin de observar la posible existencia de correlación entre el intervalo de tiempo entre la publicación entre versiones y la dispersión de acoplamiento.

A continuación, se realizaron gráficas para visualizar el incremento de LOC por día entre versiones; para explorar la posible existencia de tendencias en cuanto al ritmo en de crecimiento a lo largo de las sucesivas versiones, se agregó la línea de tendencia calculada por LibreOffice Calc[33].

El empleo de herramientas de libre disponibilidad, como iPlasma y LibreOffice, favorece la reproducibilidad y la extensibilidad de la experiencia.

4 Resumen de los Datos Obtenidos

Los datos obtenidos de acuerdo a la metodología expuesta en el apartado anterior se organizaron en gráficas y tablas.

En la Tabla 2 resumimos la información observada en cada proyecto. La clasificación de la tendencia se basa en la observación visual

Tabla 2: Coeficiente de Pearson y caracterización de la tendencia en la variación del incremento del tamaño por día entre versiones, para cada proyecto analizado.

Proyecto	Coeficiente de Pearson entre DA y +LOC/día	Tendencia Observada
PDFBox	0,171	Levemente Ascendente
JFreeChart	-0,143	Descendente
Art Of illusions	-0,015	Descendente
FreeMind	-0,063	Constante
ICEPDF	0,094	Descendente
Lwjgl	-0,05	Constante
Batik	0,274	Descendente
Chemistry Development Kit	0,5	Ascendente
Vuze	0,021	Descendente
Sweet Home 3D	-0,335	Descendente
HSQLDB	0,53	Levemente Ascendente
Dynamic Reports	-0,05	Descendente
Jexplorer	0,11	Constante

En este trabajo se contemplaron 411 versiones de los 13 proyectos analizados

5 Análisis de los datos

Los gráficos elaborados de acuerdo con la metodología descrita en la sección 3 permiten observar que la mayoría de los proyectos analizados (7 sobre 13) muestran una tendencia descendente en cuanto a la incremento de las líneas de código por día entre versiones, según se resume en la tabla 3.

En 3 de los proyectos no se visualizan variaciones que puedan constituir tendencias en cuanto al crecimiento del tamaño del software.

Si bien en otros 3 parece verificarse una tendencia creciente, es decir, de mayor incremento de tamaño por día de una versión a la siguiente, sólo en un proyecto se visualiza esta tendencia con claridad: Chemistry Development Kit (ilustración 1); esta aplicación lleva ya 9 años de desarrollo.

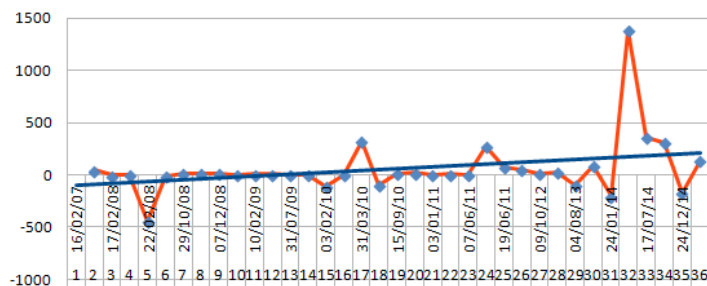


Ilustración 1: Chemistry Development Kit exhibe una tendencia a aumentar el crecimiento en LOC por día

Por razones de brevedad, incluimos sólo un gráfico que muestra claramente una tendencia decreciente (ilustración 2) y el que parece aumentar su crecimiento con el tiempo.

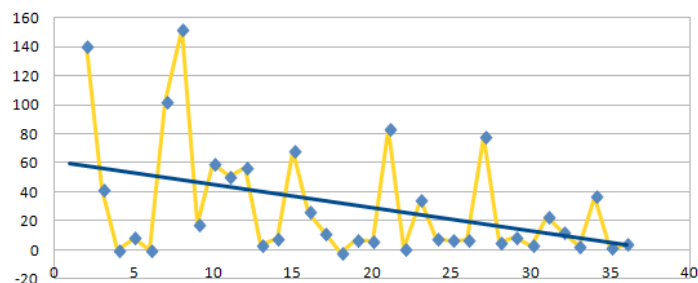


Ilustración 2: Sweet Home 3D muestra una tendencia a disminuir el ritmo de crecimiento en LOC por día

En cuanto a la relación entre el acoplamiento y el ritmo de crecimiento del tamaño, sólo en dos de los proyectos aparece una correlación que podría indicar una vinculación relevante, al menos en término de relaciones lineales: HSQLDB y Chemistry Development Kit. No obstante, cabe destacar que el último ha publicado una cantidad de versiones importante (36), en tanto que en los demás proyectos que superan las 30 versiones los valores del coeficiente son muy cercanos a cero.

6 Conclusiones y trabajos futuros

El trabajo realizado permite observar que, al menos en el conjunto de aplicaciones analizado, se verifica un paulatino descenso en el ritmo de crecimiento de productos F/OSS a lo largo de su evolución. Este resultado es compatible con las afirmaciones condensadas en las llamadas Leyes de la Evolución del Software.

Sin embargo, se observa también que en algunos proyectos no se verifica un descenso en ese ritmo; en un caso, inclusive, se observa un aumento en esa tasa. Al tratarse de una aplicación que lleva muchos años de desarrollo y que ha producido más de una treintena de versiones, ese comportamiento es llamativo; sería de esperar que un programa alcance cierta estabilidad en cuanto a los requerimientos y demás factores que afectan al proceso evolutivo, por lo que el aumento del ritmo del crecimiento amerita indagar sobre los factores que están incidiendo en ese comportamiento.

Los proyectos que exhiben una tendencia constante pueden relacionarse con la realización de trabajos por parte del equipo de desarrollo orientados a reducir las dificultades de modificación del software para futuras versiones; esto sería compatible con la formulación de las leyes II y VII de la Evolución del Software. De todos modos, requeriría una investigación más profunda para verificar esa hipótesis, o para detectar otros factores que podrían estar en juego.

Respecto de la relación entre la medida de acoplamiento “dispersión de

acoplamiento” y el ritmo de crecimiento del software entre versiones, los datos parecen descartar que exista una correlación -al menos de tipo lineal. Si en algunos casos esta vinculación aparece como significativa, podría deberse a características específicas del proyecto sobre las cuales hace falta mayor investigación.

En cuanto a la validez de estos resultados, cabe considerar que la métrica LOC como medida de tamaño es una de las más utilizadas y se ha verificado en otros estudios una fuerte correlación con otras métricas usadas en la investigación sobre la evolución del software[31]. En cambio, la métrica que elegimos para describir el acoplamiento podría no estar captando aspectos de la complejidad que tienen mayor incidencia en la modificabilidad o evolutividad del software (entendida como un atributo que evalúa la facilidad o dificultad de que un producto evolucione).

En futuros trabajos sería pertinente indagar en las relaciones con otras métricas, y adoptar un enfoque multivariante que permita incorporar factores que tienen incidencia en el proceso evolutivo del F/OSS, como la cantidad de desarrolladores, la popularidad, características propias del dominio al que se aplican, etc.

Referencias

1. Kon, F., Meirelles, P., Lago, N., Terceiro, A.S., Chavez, C., Mendonça, M.G.: Free and Open Source Software Development and Research: Opportunities for Software Engineering. In: SBES. pp. 82–91 (2011).
2. von Krogh, G., von Hippel, E.: The Promise of Research on Open Source Software. *Manage Sci.* 52, 975–983 (2006).
3. González-Barahona, M., Robles, G.: On the reproducibility of empirical software engineering studies based on data retrieved from development repositories. *Empir. Softw. Eng.* 17, 75–89 (2012).
4. Bauer, V., Eckhardt, J., Hauptmann, B., Klimek, M.: An Exploratory Study on Reuse at Google. In: Proceedings of the 1st International Workshop on Software Engineering Research and Industrial Practices. pp. 14–23. ACM, New York, NY, USA (2014).
5. Kemerer, C.F., Slaughter, S.: An Empirical Approach to Studying Software Evolution. *IEEE Trans. Softw. Eng.* 25, 493–509 (1999).
6. Pei-Breivold, H., Chauhan, M.A., Babar, M.A.: A Systematic Review of Studies of Open Source Software Evolution. In: 17th Asia Pacific Software Engineering Conference (APSEC), IEEE (2010).
7. Moazeni, R., Link, D., Boehm, B.: Lehman’s Laws and the Productivity of Increments: Implications for Productivity. In: 2013 20th Asia-Pacific Software Engineering Conference (APSEC). pp. 577–582. IEEE (2013).
8. Lehman, M.M., Ramil, J.F.: An approach to a theory of software evolution. In: IWPSE ’01: Proceedings of the 4th International Workshop on Principles of Software Evolution. pp. 70–74. ACM, New York, NY, USA (2001).
9. Ciraci, S., Broek, P.V.D.: Evolvability as a Quality Attribute of Software Architectures. In: International ERCIM Workshop on Software Evolution 2006 (2006).
10. Crnkovic, I.: Predictability and evolution in resilient systems. In: *Software Engineering for Resilient Systems*. pp. 113–114. Springer (2011).
11. Barais, O., Lawall, J., Meur, A.-F.L., Duchien, L.: Software Architecture Evolution. In: Mens, T. and eds, S.D. (eds.) *Software Evolution*. pp. 233–262. Springer Verlag (2008).
12. Herraiz, I., Rodriguez, D., Robles, G., Gonzalez-Barahona, J.M.: The Evolution of the

- Laws of Software Evolution: A Discussion Based on a Systematic Literature Review. *ACM Comput Surv.* 46, 28:1–28:28 (2013).
13. Herraiz, I.: A statistical examination of the evolution and properties of libre software. In: *ICSM*. pp. 439–442 (2009).
 14. Godfrey, M.W., Tu, Q.: Evolution in Open Source Software: A Case Study. In: *Proceedings of the International Conference on Software Maintenance*. pp. 131–142 (2000).
 15. Turski, W.M.: Reference Model for Smooth Growth of Software Systems. *IEEE Trans Softw Eng.* 22, 599–600 (1996).
 16. Turski, W.M.: The reference model for smooth growth of software systems revisited. *IEEE Trans. Softw. Eng.* 28, 814 (2002).
 17. Robles-Martínez, G., González-Barahona, J.M., Centeno-González, J., Matellán-Olivera, V., Rodero-Merino, L.: Studying the evolution of libre software projects using publicly available data. In: *3 rd Workshop on Open Source Software Engineering*. p. 111 (2003).
 18. Robles, G., Amor, J.J., Gonzalez-Barahona, J.M., Herraiz, I.: Evolution and Growth in Large Libre Software Projects. In: *IWPSE '05: Proceedings of the Eighth International Workshop on Principles of Software Evolution*. pp. 165–174. IEEE Computer Society, Washington, DC, USA (2005).
 19. González-Barahona, J., Robles, Gregorio, G., Herraiz, Israel, I., Ortega, Felipe, F.: Studying the laws of software evolution in a long-lived FLOSS project. *J. Softw. Evol. Process.* 26, 589–612 (2014).
 20. Capiluppi, A., Morisio, M., Ramil, J.F.: Structural Evolution of an Open Source System: A Case Study. In: *IWPC*. pp. 172–182 (2004).
 21. Terceiro, A., Chavez, C.: Structural Complexity Evolution in Free Software Projects: A Case Study. In: *Quality and Architectural Concerns in Open Source Software* (2009).
 22. Neamtiu, I., Xie, G., Chen, J.: Towards a better understanding of software evolution: an empirical study on open-source software. *J. Softw. Evol. Process.* 25, 193–218 (2013).
 23. Koch, S.: Software evolution in open source projects—a large-scale investigation. *J Softw Maint Evol.* 19, 361–382 (2007).
 24. Ramirez, J., Gimson, L., Gil, G.: Evaluación de la Evolución del Diseño en F/OSS: un Caso de Estudio. In: *VII Workshop Ingeniería de Software - XVI Congreso Argentino de Ciencias de la Computación* (2010).
 25. Ramirez, J., Reyes, C., Gil, G.: Métricas de Código fuente y Evolución de F/OSS: un estudio exploratorio. In: *42 Jornadas Argentinas de Informática*. pp. 198–209. , Córdoba (2013).
 26. Ramirez, J., Gil, G., Reyes, C.: Umbrales sugeridos para promedios de métricas de diseño de una aplicación en Java, (2015).
 27. Runeson, P., Höst, M.: Guidelines for conducting and reporting case study research in software engineering. *Empir. Softw Engg.* 14, 131–164 (2009).
 28. Ramirez, J.: F/OSS para el reuso: Métricas, Desarrollo de Herramientas y Marco para su Evaluación, <http://hdl.handle.net/10915/48176>, (2015).
 29. Marinescu, C., Marinescu, R., Mihancea, P.F., Ratiu, D., Wettel, R.: iPlasma: An Integrated Platform for Quality Assessment of Object-Oriented Design. In: *ICSM (Industrial and Tool Volume)*. pp. 77–80 (2005).
 30. Lanza, M., Marinescu, R., Ducasse, S.: *Object-Oriented Metrics in Practice*. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005).
 31. Herraiz, I., Robles, G., Gonzalez-Barahona, J.M.: Comparison between SLOCs and number of files as size metrics for software evolution analysis. *2011 15th Eur. Conf. Softw. Maint. Reengineering.* 0, 206–213 (2006).
 32. Thomas, L.G., Schach, S.R., Heller, G.Z., Offutt, J.: Impact of release intervals on empirical research into software evolution, with application to the maintainability of Linux. *Softw. IET.* 3, 58–66 (2009).

33. Document Foundation: Curvas de Regresión en LibreOffice, https://help.libreoffice.org/Chart/Trend_Lines/es#La_ecuaci.C3.B3n_de_regresi.C3.B3n_lineal.

“Taxonomía de Requisitos para Aplicaciones Web”

Silvia Sánchez-Zuaín¹, Elena Durán^{1,2}

¹Facultad de Matemática Aplicada
Universidad Católica de Santiago del Estero (UCSE)
szuain@ucse.edu.ar; elena.duran@ucse.edu.ar

²Facultad de Ciencias Exactas y Tecnologías
Universidad Nacional de Santiago del Estero (UNSE)
eduran@unse.edu.ar

Resumen: Las aplicaciones web ofrecen una variedad de funcionalidades con diferentes características y requisitos que dan lugar a diversas clasificaciones. Categorizarlas, es útil para la comprensión de sus necesidades y para desarrollar e implementar sistemas basadas en la Web. Definir los requisitos considerando la categoría de aplicación web puede resultar muy beneficioso. En este trabajo se propone una Taxonomía de requisitos para aplicaciones web, a partir de analizar las diferentes categorías de estas aplicaciones, e identificar los requisitos presentes en las mismas. Sobre esta base se elaboró una taxonomía compuesta de tres niveles: en el primer nivel se diferencian requisitos generales de los específicos; en el segundo nivel, para los requisitos específicos, se considera el tipo de aplicación web; y por último en el tercer nivel se clasifica a los requisitos como funcionales y no funcionales. La Taxonomía resultante ayudará a los desarrolladores Web a minimizar los errores durante la etapa de desarrollo, contribuirá a generar sistemas web, de excelente calidad, evitando retrasos que afectan tanto al equipo de desarrollo como al cliente.

Palabras Claves: Aplicaciones web, Ingeniería de Requisitos, Clasificación de requisitos de software

1 Introducción

Hoy las aplicaciones web ofrecen una variedad de funcionalidades con diferentes características y requisitos que dan lugar a diversas clasificaciones. Se han extendido en sus ámbitos y usos, generando fuertes dependencias al mundo cotidiano, además capturando al usuario con la incorporación de técnicas de personalización. Así se llega a la categorización de aplicaciones web y a categorías como la semántica o la ubicuidad en la Web, donde el software es capaz de procesar su contenido, razonar con este, combinarlo y realizar deducciones lógicas para resolver problemas cotidianos automáticamente desde cualquier lugar y dispositivo. Categorizar las aplicaciones web, es útil para la comprensión de sus necesidades y para desarrollar e implementar sistemas y aplicaciones basadas en la Web [1].

En el proceso de desarrollo de una aplicación, el equipo de trabajo se enfrenta al problema de la identificación de requisitos. El objetivo de esta etapa es analizar y documentar las necesidades funcionales que deberán ser soportadas por el sistema a desarrollar. El conjunto de requisitos ha de ser tan completo, consistente y correcto como sea posible. Las características especiales, de las aplicaciones web, hay que

tenerlas en cuenta en la fase de especificación de requisitos [2]. De esta manera se facilitará y simplificará el trabajo de los desarrolladores web y en consecuencia se dará una solución de excelente calidad y funcionalidad al cliente.

En este trabajo se presenta una taxonomía de requisitos elaborada a partir de diferentes categorías de aplicaciones web relevadas; ya que, identificando a qué categoría pertenece una aplicación web, se podrá establecer qué tipos de requisitos debe satisfacer, la misma, como mínimo.

El documento se organiza de la siguiente manera: en la sección dos se presenta una revisión de los antecedentes de categorizaciones de Aplicaciones Web y, en la sección tres, una revisión de clasificaciones de requisitos de software. Luego, en la sección cuatro, se explica el proceso llevado a cabo para la construcción de la taxonomía y se presenta la misma, y por último se explicitan las conclusiones del trabajo realizado.

2. Revisión de Antecedentes de categorías de aplicaciones web

Con el fin de seleccionar las categorías de aplicaciones web que se tomarían como base para definir la Taxonomía de Requisitos, se analizaron diferentes clasificaciones propuestas por los siguientes autores: Ginige y Murugesan [4] proponen una categoría, que ilustra la evolución de las Aplicaciones Web. Esto es útil en la comprensión de sus necesidades, para desarrollar e implementar Sistemas web. Kappel et al. [3] incluyen tipos nuevos a la anterior clasificación tales como aplicaciones web ubicuas y web semántica. Además, muestra la relación entre la cronología de desarrollo y la complejidad. Coelho et. al [5] categorizan basándose en el análisis del tráfico HTTP por medio del cual un cliente solicita una página Web al servidor, que busca y envía la página a través de la red hacia el cliente. Softaculous [6] ha confeccionado una lista teniendo en cuenta, que existen también en la nube, servicios de instalación de aplicaciones Web que disponen de una amplia variedad de las mismas. Pressman [7] presenta una categoría basada en las actividades del proceso. Rossi G. et al. [8] definen una categoría que atienden a características principales y la tecnología utilizada para su creación.

En base a estas clasificaciones se construyó una tabla con las categorías presentes en la mayoría de los trabajos analizados. Para ello, se unificaron aquellas categorías que tenían el mismo significado, pero que cada autor las denominaba con nombres diferentes. Además, algunos tipos de aplicaciones Web se agruparon en las categorías más generales, teniendo en cuenta el principal servicio ofrecido, uso o funcionalidad; reduciendo de esta forma el número de dominios lo máximo posible como para ser abordable en el momento de identificar los requisitos. La categorización de Aplicaciones Web resultante que se tuvo en cuenta para armar la taxonomía de requisitos es la que se muestra en la Tabla 1.

Tabla .1 Síntesis de las categorías resultantes del análisis

Categorías de Aplicación web	Autores					
	Kappel et al. [3]	Gini y Murugesan [4]	Coelho et al. [5]	Softaculous It [6]	Pressman [7]	Rossi et al. [8]
1. Interactivas	x	x		x	x	x
2. Transaccional	x	x		x		
3. Colaborativas	x	x		x		
4. Portales web	x	x		x	x	
5. web social		x		x		
6. Orientadas a Servicios		x	x	x	x	
7. Ubicua (móvil)	x					x
8. Semántica	x					x

A continuación, se definen, según los autores consultados, los tipos de aplicaciones web presentadas en la Tabla 1.

- 1. Interactivas:** interactúan con el usuario, su contenido está en constante cambio y se genera según las necesidades del usuario. El uso de formularios HTML y Common Gateway Interface (CGI) ofrece nuevas formas y aspectos que mejora la interfaz y menús de selección. Por ejemplo: aplicaciones de juegos en línea.
- 2. Transaccionales:** se crearon para brindar mayor interactividad al usuario, permitiéndole acciones no sólo de lectura sino también de actualización de los contenidos del sitio. Por ejemplo: aplicaciones para compras electrónicas.
- 3. Colaborativas:** se utilizan para fines colaborativos, tienen un sentido constructivo y activo; implican un proceso de desarrollo como una estrategia que invita al usuario a interactuar intercambiando conocimientos y capacidades. Por ejemplo: aplicaciones distribuidas de Auditoría.
- 4. Portales web:** ofrecen un único punto de acceso a fuentes independientes, potencialmente heterogéneas de información y servicios. Por ejemplo: www.educar.ar.
- 5. Web social:** permiten a las personas conectarse y trabajar a través de límites organizacionales y físicos, creando lo que se conoce como “comunidades”. Se basan en los vínculos que hay entre sus usuarios. Existen varios tipos de redes sociales: genéricas, profesionales y verticales o temáticas. Por ejemplo: www.facebook.com.
- 6. Orientadas a Servicios:** este tipo de aplicaciones intentan ofrecer un servicio especializado, se incluye un sin número de herramientas, softwares, plataformas en línea para ofrecer servicios de valor añadido al usuario final. Por ejemplo: aplicaciones para Seguimiento Postales.
- 7. Ubicuas:** proveen servicios personalizados. El usuario puede acceder a la información desde cualquier lugar, a toda hora, independientemente del tipo de dispositivo que utilice. Por ejemplo: aplicaciones de asistencia al turista.

8. **Semánticas:** brindar información no sólo para el entendimiento humano sino para los propios sistemas que la manipulan. Esto facilitaría la gestión del conocimiento en la Web. Por ejemplo: buscadores semánticos en la web.

3. Revisión de Antecedentes de clasificaciones de requisitos de software

Diferentes autores ofrecen diferentes clasificaciones y definiciones de diversos requisitos. En este trabajo se presentaron las clasificaciones más reconocidas. Estos autores usaron terminologías diferentes para nombrar a requisitos con el mismo significado, por lo tanto, se procedió a agruparlos para facilitar la identificación y comprensión de los tipos de requisitos planteados. A continuación, en la Tabla 2 se visualiza una síntesis de las categorías analizadas.

En el ámbito de la Ingeniería de Software, en general, los requisitos se clasifican en dos grandes grupos: funcionales, no funcionales. Sin embargo, cuando se trata de desarrollos de aplicaciones web, se deben considerar otros elementos de interés, por lo que se atienden requisitos de contenido, del ambiente del sistema, de interfaces con el usuario y de evolución de la aplicación, además necesitan cumplir con requisitos de calidad como son rendimiento, usabilidad, escalabilidad, mantenimiento, accesibilidad, entre otros.

Tabla 2. Síntesis de las categorías de requisitos analizadas

Requisitos			Autores				
<i>Tipos</i>	<i>Categorías</i>	<i>Subcategorías</i>	Sommerville [9]	Roman y Farley [10]	Griñbacher, P [11]	Escobedo, M. J., Koch, N [12]	Standard IEEE [13]
Funcionales			x	x	x	x	x
No Funcionales	Producto	Usabilidad	x		x		
		Eficiencia	x				
		Confiablez	x	x			
		Seguridad	x	x			x
		Mantenibilidad			x		x
		Portabilidad			x		x
	Organizacional	Entorno	x		x		
		Operacionales	x	x			
		Desarrollo	x	x			
	Externos	Regulatorios	x				
		Éticos	x				x
		Legales	x	x		x	
	Interfaz	Usuario		x	x	x	x
		Sistema		x	x		x
	Performance						x
	Económicos			x			
	Restricciones de Diseño			x	x		x
	De Contenidos				x	x	
De evolución					x		
Conceptuales o Almacenamiento						x	x
De Navegación						x	
Personalización						x	
De Adaptación							x

4. Construcción de la Taxonomía

Una taxonomía: es un proceso científico (o un sistema particular) de categorizar entidades, es decir, de organizarlas en grupos [14]. Un sistema taxonómico debe ser claro y consistente, flexible, exhaustivo y práctico. Por medio de la taxonomía se pueden establecer categorías dentro de la clasificación, según se deseen determinar relaciones de semejanza (principio de interacción) o de interdependencia (principio de dualidad). Se fija una unidad estructural que será a su vez la unidad de clasificación que se denomina Taxón [15].

4.1 Características de la Taxonomía de Requisitos para aplicaciones Web

Siguiendo lo establecido por Argudo y Centelles [16], se definieron las siguientes características básicas de la taxonomía:

- ✓ *Alcance:* Esta dentro de la Ingeniería de Requisitos y para Sistemas Web. Los usuarios principales serán los Ingenieros Web y Desarrolladores, quienes participan en el proceso de creación del Sistema Web (analistas, clientes, usuarios, diseñadores gráficos, expertos en multimedia y seguridad, etc.). La Taxonomía contendrá exclusivamente requisitos para aplicaciones web, estará escrita en el idioma castellano, salvo expresiones de habla inglesa que no tengan traducción al español.
- ✓ *Usos y funciones:* Se utilizará para apoyar el desarrollo de Aplicaciones web. La misma podría ser utilizada como: una herramienta para los ingenieros en Requisitos y para los desarrolladores web, una guía para la creación de una Aplicación web, en la etapa de especificación de requisitos, una lista de comprobación para determinar si se han considerado todos los tipos de requisitos según el tipo de aplicación web a desarrollar.
- ✓ *Tipo de taxonomía:* visible al usuario.
- ✓ *Grado de complejidad de categorización:* Podrá tener un crecimiento continuo en la medida que aparezcan otras categorías de aplicaciones web diferentes a las consideradas en este trabajo.
- ✓ *Objetivo de la Taxonomía:* “Ofrecer una estructura de clasificación de requisitos, relacionados con los diferentes tipos de Aplicaciones Web, que sea soporte para desarrollar sistemas web”.
- ✓ *Herramientas existentes que van a usarse para definir la Taxonomía:* Los recursos utilizados para definir la taxonomía de requisitos fueron: las categorías de aplicaciones web presentadas y sintetizadas en la Tabla 1, y las categorías de requisitos presentadas y sintetizadas en la Tabla 2.

4.2 Proceso llevado a cabo para la construcción de la Taxonomía

Para la construcción de una taxonomía es posible diferenciar cuatro etapas necesarias [16], las que se explican a continuación para la Taxonomía propuesta.

4.2.1 Identificación de criterios de facetación

Se entiende por facetas a los aspectos, propiedades o características de una realidad específica claramente definida, mutuamente exclusiva y, en conjunto, exhaustivos. Cada una de las facetas se descompone en categorías de diferentes niveles de

especificidad. Para identificarlas correctamente debe tenerse en cuenta el resultado del análisis del contexto, de la audiencia y de los contenidos.

Para la Taxonomía de Requisitos de aplicaciones Web la unidad de clasificación, es decir, el Taxón, son los Requisitos web. Si bien es cierto que los requisitos de software expresan las necesidades y restricciones en un producto software o de un servicio que contribuye a solucionar los problemas del mundo real, no tienen la simplicidad estructural ni la homogeneidad que proponen la mayoría de los autores en la bibliografía consultada. Por otra parte, los requisitos de software se definen en diferentes ámbitos y con diferentes niveles de detalle. Es por ello que, en este trabajo se ha considerado tres grandes criterios que a continuación se definen.

Criterio N° 1: según la granularidad de los requisitos, se consideran:

- **Requisitos Generales:** independientemente del tipo de aplicación web que se desarrollará. Identificar cuáles serán los requisitos fundamentales, básicos, que se deben tener en cuenta, para empezar a diseñar un prototipo del software y así reflejar los requisitos del Cliente. Además, esto servirá para que todo el equipo de desarrollo tenga en claro el objetivo que se quiere alcanzar al diseñar la aplicación web. Por lo tanto, es importante tener, en una primera etapa, los Requisitos Generales bien identificados en el momento de encarar un desarrollo de una Aplicación Web. Ellos son:
 - ✓ *Saber a qué usuario va dirigido:* definir para que tipo de público se va orientar la aplicación web, por ejemplo, si será para estudiantes, empleados de una empresa, o cualquier otro tipo de usuario.
 - ✓ *Definir el contexto de la Aplicación Web:* Todos los desarrolladores deben entender “por qué” y “para qué” se está creando esa Aplicación Web, para que no se pongan cosas fuera del contexto que se desea.
 - ✓ *Definir el tipo de Aplicación Web* que se creará en función del contexto y del tipo de Usuarios a la que va dirigido.
 - ✓ *Tomar en cuenta cada actor* (participante) que formarán parte del equipo que encarará el diseño y desarrollo de la Aplicación Web: contemplar con respecto al tipo de Aplicación y los requisitos básicos relacionados a la misma, que perfiles se requiere durante el desarrollo de la aplicación para dividir las responsabilidades. (ingenieros de requisitos, diseñadores, los arquitectos, etc.)
- **Requisitos específicos:** Se identifican y se definen los requisitos específicos teniendo en cuenta el tipo de Aplicación web, es decir, no sería aceptable un sistema web que no satisfaga alguno de los requisitos aquí presentados.

Criterio N° 2: Tener en cuenta la categoría de aplicación web establecida en este trabajo.

Criterio N° 3: para cada tipo de aplicación web definir requisitos:

- **Funcionales:** descripción de lo “que” hace el sistema. Estos requisitos responden a la pregunta *¿qué se puede hacer en la aplicación Web*, es decir que se debe definir qué se va a poder hacer con la información de la aplicación Web y las posibles funcionalidades [18].

- **No Funcionales:** descripción de “cómo” lo hace el sistema, esto implica, definir cómo el sistema tiene que comportarse o qué límites se imponen en la solución por el medio ambiente. En cualquier aplicación Web aparecen una serie de necesidades que se pueden catalogar dentro de los requisitos no funcionales para aplicaciones Web. Se incluyen las siguientes: Calidad: usabilidad, eficiencia, confiabilidad, seguridad, mantabilidad, portabilidad; Organizacional: entorno, operacionales, desarrollos; Externos: regulatorios, éticos, legales/políticos, económicos; Interfaz: usuario, sistema: incluye hardware, software y comunicación; Performance; Restricciones de Diseño; De Contenidos; De evolución; Conceptuales; De Personalización; De Adaptación.

4.2.2 La extracción del léxico

El objetivo de esta fase es identificar todos los términos o categorías que designan los conceptos propios del dominio. Cada categoría debe vincularse, como mínimo, a una de las facetas establecidas en la fase anterior.

En gran parte el proceso de extracción del léxico se realizó al construir las tablas de requisitos presentadas en sección 3. Los términos o categorías que se identificaron dentro del dominio son: Requisitos para Aplicaciones web, Requisitos Generales, Requisitos Específicos, Tipo de Aplicaciones Web, Requisitos Funcionales y Requisitos No Funcionales. El conjunto de designaciones se denomina, técnicamente, el léxico del dominio. Cada designación se denomina término o, categoría.

4.2.3 El control del léxico

Es posible que en la fase anterior se hayan identificado diferentes designaciones posibles para un mismo concepto (sinónimos o cuasisinónimos) y/o que una misma designación presente diferentes formas posibles (gramaticales, ortográficas, etcétera). Por Ejemplo: Requisitos de interfaz (al usuario), también llamados en algunas propuestas requisitos de interacción o de usuario. La eficacia de una taxonomía recomienda el establecimiento de un término preferente para representar cada uno de los conceptos que integran su dominio. Al hacer el control del léxico, proceso que se ejecutó al construir la Tabla 2, quedarían los Requisitos para Aplicaciones web listados y agrupados en esta tabla.

4.2.4 Estructura de la Taxonomía

El objetivo de esta fase es identificar y establecer dos tipos de relaciones entre las categorías de la taxonomía: la relación jerárquica y la relación asociativa. El resultado será un vocabulario controlado en forma de estructura conceptual. La relación jerárquica se basa en grados o niveles de orden superior y subordinación, en que un término general representa un todo o clase y los términos subordinados corresponden a sus miembros, partes o instancias. Todas las categorías de una faceta deben estar conectadas por relaciones jerárquicas. La relación asociativa conecta categorías (de diferentes facetas) a partir de asociaciones de ideas que se pueden dar entre: una acción y su resultado o producto; un concepto y una propiedad del mismo; un producto y el material con que está hecho; etcétera.

Tradicionalmente, se han distinguido dos técnicas para el desarrollo de la estructura de la taxonomía [17]:

- *De arriba a abajo* ("up to down") supone la identificación inicial de un número limitado de categorías superiores, y la agrupación del resto de categorías en niveles sucesivos de subordinación hasta alcanzar los niveles de categorías más específicas. Esta técnica puede orientarse tanto a la aplicación de un modelo estructural jerárquico (y/o arbóreo) como facetado.
- *De abajo a arriba* ("down to up") se basa en la identificación inicial de las categorías más específicas, que van agrupándose en niveles sucesivos de superordinación hasta alcanzar el nivel de categorías superiores. Generalmente, esta técnica se ha orientado, fundamentalmente, a la aplicación de un modelo estructural jerárquico (y/o arbóreo), aunque, como en el caso anterior, puede facilitar el análisis para la toma de una decisión sobre el modelo estructural que resulta idóneo aplicar.

La técnica elegida, en este trabajo, es la de "arriba a abajo". Con esta técnica se obtendrá una clasificación jerárquica, estableciendo un orden de mayor a menor, esto dará un sentido de colectividad y generalidad.

4.2.5 Determinación de los niveles de la Taxonomía

Teniendo en cuenta los criterios taxonómicos definidos en el punto 4.2 se determinaron tres niveles, como se muestra en la Figura 1.

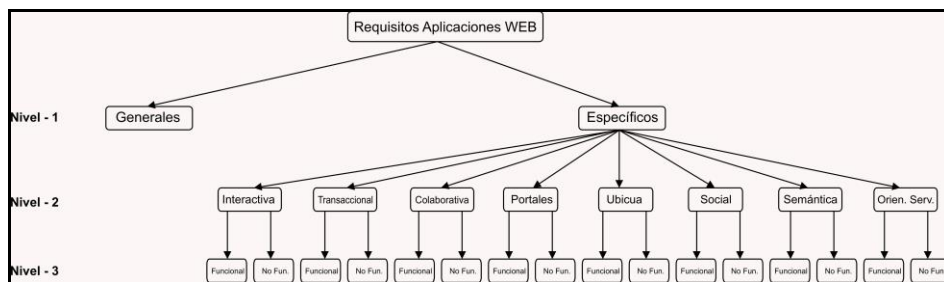


Figura 1: Estructura de la Taxonomía definida

Además, a fin de considerar en la estructura de la Taxonomía lo definido en el criterio 3 del apartado 4.2.1 se procedió a identificar qué requisitos presentes en la Tabla 2 debían definirse en cada tipo de aplicación web presentada en la Tabla 1. En base a ello, se construyó la Tabla 3, donde se marca el tipo de requisito que debe definirse para cada tipo de Aplicación Web.

Tabla 3 Requisitos relacionados al tipo de Aplicación web

Requisitos			Autores							
Tipos	Categoría	Subcategoría	Interactivas	Transaccional	Colaborativas	Portales	Orientadas a Servicios	web social	web Ubicua (móvil)	web Semántica
No Funcionales	Calidad	Usabilidad	x	x	x	x	x	x	x	x
		Eficiencia	x	x		x				
		Confiablez	x	x	x		x		x	x
		Seguridad	x	x	x		x	x	x	x
		Mantenibilidad	x	x	x	x	x	x		
		Portabilidad	x	x	x		x	x		
	Organizacional	Entorno	x	x	x		x			
		Operacionales	x	x	x		x	x	x	
		Desarrollo	x	x	x	x	x	x	x	x
	Externos	Regulatorios	x	x	x		x			
		Éticos	x	x	x	x	x	x	x	x
		Legales	x	x	x	x	x	x	x	x
		Económicos	x	x	x	x	x	x	x	x
	Interfaz	Usuario	x	x	x	x	x	x	x	x
		Sistema	x	x	x	x	x	x	x	x
		Performance	x	x	x		x			
		Restricciones de Diseño	x	x	x		x		x	x
		De Contenidos			x	x		x		x
		De evolución	x	x			x		x	
		Almacenamiento	x	x	x	x	x	x	x	x
		De Navegación		x					x	x
		De Personalización	x	x		x			x	x
		De Adaptación							x	x

5. Conclusiones y Trabajos futuros

La taxonomía propuesta fue diseñada para dar cabida a un propósito. El presentado en este trabajo consiste en definir una estructura de clasificación de requisitos, relacionados con los diferentes tipos de Aplicaciones Web, que será soporte para desarrollar sistemas web. Sobre la base del análisis de las diferentes categorías de las aplicaciones web más conocidas se planteó una taxonomía de requisitos para aplicaciones web. El presente trabajo sienta las bases conceptuales para la construcción de cualquier Taxonomía de requisitos, inclusive para un refinamiento de la aquí presentada, ya que se explican los cuatro procesos realizados: Identificación de criterios de facitación, extracción del léxico, control del léxico y por último el desarrollo de la estructura de la taxonomía

La propuesta planteada será útil durante la fase de Ingeniería Requisitos, fase primordial en el proceso de producción de software, ya que se enfoca en la definición de lo que se desea producir. Con la taxonomía será posible acelerar la identificación de requisitos previa identificación de los objetivos del cliente y del tipo de aplicación web que se pretende desarrollar. Contar con una Taxonomía de requisitos permite determinar las necesidades de un proyecto de software web cubriendo sus aspectos más relevantes. La taxonomía resultante en este trabajo se convierte en una herramienta de soporte para todos los actores Involucrados en la IR, permitirá en el proceso de la elicitación, obtener y especificar los requisitos teniendo en cuenta el tipo de Aplicación web a desarrollar.

Como trabajo futuro, y con el fin de obtener conclusiones consistentes a lo largo del tiempo, se aplicará la taxonomía a casos concretos, con el fin de comparar entre los requisitos identificados con un desarrollo tradicional y requisitos identificados usando

la taxonomía propuesta. Por otro lado, también se podría encarar una taxonomía de requisitos teniendo en cuenta las vistas de la arquitectura de un sistema. Para representar sistemas complejos lo mejor es dividir los puntos de vista, de diferentes interesados tales como usuarios finales, desarrolladores o directores de proyectos. Son puntos de vista distintos pero relacionados, cada uno de los cuales describe un aspecto independiente de la arquitectura. Colectivamente, las vistas describen todo el sistema, se pueden considerar cinco vistas estándares. Cada vista representa un comportamiento particular del sistema. Estas vistas tienen como propósito modelar los requerimientos de acuerdo a la función que desempeñen.

6. Referencias

1. Suh, Woojong, “*Web engineering: principles and techniques*” Idea Group Inc., 2005.
2. M. J. Escalona and N. Koch, “*Requirements Engineering for Web Applications A Comparative Study*,” J. Web Eng., vol. 2, N° 3, 2004.
3. Kappel, G. “*Web Engineering - Old Wine in New Bottles?*”, Invited Talk, International Conference on Web Engineering. Munich, July 28-30. 2004.
4. A. Ginige, S. Murugesan “*Web Engineering: An Introduction*”, IEEE Multimedia, Jan-Mar 2001.
5. Coelho, N.; Salvador, P.; Nogueira, A. “*Differentiation of HTTP Applications based on Multiscale Analysis*.” Bentham Science Publisher, vol. 2, N° 1, pp.12-25. 2013.
6. Softaculous Ltd. (s.f.). Obtenido de <https://www.softaculous.com/apps>.
7. Pressman, R. “*Ingeniería del Software. 6ª Ed.*” Mcgraw-Hill.n. 2005.
8. Rossi Oscar, Pastor Oscar, Schwabe Daniel and Olsina Luis: “*Web Engineering: Modelling and Implementing Web Applications*”. (Eds), Springer.2007.
9. Sommerville, Ian, “*Ingeniería del Software*”, 9ª ed. Addison Wesley, 2011.
10. Roman, Gruiá-Catalin, “*A Taxonomy of Current Issues in Requirements Engineering*,” IEEE Computer, Vol. 18, No.4, pp.14-22. April 1985
11. Fairley, Richard E., “*Software Engineering Concepts*”, McGraw-Hill, 1985.
12. Escalona, M. J., Koch, N.: “*Metamodeling the Requirements of Web Systems*”. 2nd International Conference on WebIST, Setubal, Portugal, 2006.
13. Abed Gregio AR, Barbato LGC, Duarte LO, Montes A, Hoepers C, Stedding-Jessen K. “*Taxonomias de Vulnerabilidades: Situação Atual*” Disponible en: <http://www.lbd.dcc.ufmg.br/colecoes/sbseg/2005/009.pdf> consultado el 8/04/2016.
14. Emilia Currás, “*Ontología, Taxonomía y Tesoros. Manual de Construcción y Uso*”. Ediciones TREA S.L. Tercera Edición 2005.
15. Silvia Argudo, Miquel Centelles “*Metodología para el diseño de taxonomías corporativas*”. Investigación Bibliotecológica, vol. 19, núm. 39, 2005.
16. Centelles, Miquel, “*Taxonomías para la categorización y la Organización de la información en sitios*”. 2005, disponible en <http://www.hipertext.net>, consulta realizada el 8/04/2016.
17. Escalona M.J, Mejías M, Torres J, Reina A.M. “*The NDT Development Process*.” Proceedings of IV ICWE 2003.

Una metodología para evaluar la madurez de servicios universitarios

Ariel Pasini¹, Elsa Estévez², Patricia Pesado¹, Marcos Boracchia¹

¹ Instituto de Investigación en Informática LIDI (III-LIDI) Facultad de Informática –UNLP,
50 y 120 La Plata, Buenos Aires, Argentina

² Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur
Av. Alem 1253, (8000) Bahía Blanca, Argentina
{apasini, ppesado, marcosb}@lidi.info.unlp.edu.ar,
eestevez@gmail.com

Abstract. Este trabajo introduce el concepto de Gobierno Electrónico Universitario y Servicio Universitario y propone una metodología para la evaluación de la madurez de dichos servicios aplicando conceptos de gobierno electrónico. En base a una identificación de 25 servicios básicos prestados por unidades académicas (UAs), que pueden ser parametrizados según el contexto, se define un nivel de madurez máximo que el servicio puede llegar a obtener en función a la automatización del mismo. La metodología y el modelo propuesto permiten hacer un diagnóstico y planificar posibles mejoras a la prestación de servicios prestados por las universidades a miembros de sus comunidades.

Keywords: Gobierno Electrónico Universitario, Servicios Públicos Universitarios, Modelo de Madurez, Evaluación de Calidad

1 Introducción

Los términos Gobierno Electrónico y Gobierno Digital se relacionan comúnmente con las estructuras relacionadas al Estado, ya sean, los gobiernos municipales, provinciales, o nacionales. Sin embargo, existen unidades del Estado que poseen su propio gobierno con amplia autonomía institucional. El Estado, a través de leyes, establece la conformación básica de gobierno de estas unidades y asigna un presupuesto para que las mismas puedan cumplir determinados objetivos, pero no interviene en la selección de autoridades ni en la ejecución del presupuesto. Dentro de estas unidades del Estado se encuentran las universidades nacionales y provinciales argentinas.

Si bien muchos trabajos de investigación y la práctica misma de los gobiernos han contribuido a avanzar el estado del arte en Gobierno Electrónico, poco se ha avanzado en la aplicación de los conceptos, métodos, herramientas y prácticas de Gobierno Electrónico al ámbito universitario, como se explica en la Sección 4.

Este trabajo introduce conceptos relacionados a gobierno electrónico universitario (EGOV-U) y propone una metodología para evaluar el nivel de madurez de servicios universitarios. Las principales contribuciones de este trabajo son introducir los conceptos de Gobierno Electrónico Universitario y Servicio Universitario y proponer

una metodología para definir el modelo de madurez para la prestación de dichos servicios.

El resto de este trabajo se organiza de la siguiente manera. La Sección 2 presenta antecedentes e introduce conceptos usados en este trabajo. La Sección 3 discute trabajos relacionados. La Sección 4 explica la metodología propuesta para evaluar los niveles de madurez de los servicios públicos universitarios y resume su aplicación en la Facultad de Informática de la Universidad Nacional de La Plata (UNLP). Finalmente, las conclusiones y trabajos futuros se discuten en la Sección 5.

2 Antecedentes y Conceptos

Las siguientes secciones presentan antecedentes del contexto universitario, introducen los conceptos de gobierno electrónico universitario y servicio universitario, y presentan un modelo de madurez para estos servicios.

2.1 Antecedentes

El sistema argentino de educación universitaria está conformado actualmente por 129 instituciones educativas, 53 universidades nacionales, 3 provinciales y 7 institutos estatales de educación superior de nivel universitario [1]. Además, cuenta con 50 universidades privadas, 14 institutos de educación superior universitaria de gestión privada, 1 universidad internacional y 1 universidad extranjera.

La ley de educación superior Argentina [2], en su artículo 29 establece que las instituciones universitarias tendrán autonomía académica e institucional. Entre estas atribuciones se encuentra la definición de su estatuto y su órgano de gobierno. En particular, el artículo 52 establece que los estatutos de las instituciones universitarias nacionales deben prever sus órganos de gobierno, tanto colegiados como unipersonales, así como su composición y atribuciones. Los órganos colegiados tendrán básicamente funciones normativas generales, de definición de políticas y de control en sus respectivos ámbitos, en tanto los unipersonales tendrán funciones ejecutivas.

El cuerpo colegiado se denomina “Consejo Superior”. En la mayoría de las universidades, la misma ley define su composición. Por ejemplo, en la Universidad Nacional de La Plata (UNLP) se requiere que el Consejo Superior esté constituido con un porcentaje mayor al 50 % por docentes y el resto por alumnos con un 30% de la carrera aprobada, representantes del personal administrativo y opcionalmente representación de graduados[3] . Los representantes del órgano de gobierno son elegidos por sus respectivos claustros. El representante unipersonal es denominado comúnmente Rector o Presidente. La forma de elección puede ser por voto directo o por elección del consejo superior según lo establezca el estatuto de cada Universidad.

2.2 Gobierno Electrónico Universitario y Servicios Universitarios

En base a lo explicado, el gobierno universitario está compuesto por docentes, no docentes, alumnos y graduados. Todos en su conjunto representan a la comunidad

universitaria que desarrollan sus actividades en el marco de las reglamentaciones que dispone dicho gobierno. Para llevar a cabo el cumplimiento de las reglamentaciones, la universidad pone a disposición de su comunidad un conjunto de servicios. Actualmente, varios de estos servicios son brindados a través del uso de las Tecnologías de la Información y las Comunicaciones (TICs).

Se define el concepto de Gobierno Electrónico Universitario (EGOV-U), cómo “el uso de las TICs como herramienta para mejorar los procesos y los servicios prestados por una universidad a los miembros de su comunidad”.

Se define Servicio Público como “el resultado de un proceso llevado a cabo por un organismo del sector público, o bajo un cierto control y regulación de este, por una organización especializada o no, destinado a satisfacer necesidades del público”. Adaptando esta definición al ámbito universitario, un Servicio Universitario se define como “el resultado de un proceso llevado a cabo por una universidad o, bajo un cierto control y regulación de esta, por una organización especializada o no, destinado a satisfacer necesidades de los miembros de la comunidad universitaria”. En particular, los miembros de la comunidad universitaria incluyen alumnos, graduados, docentes y no docentes.

Por ejemplo, los gobiernos universitarios prestan servicios públicos a los alumnos, al brindar información sobre el calendario académico; a los graduados, proponiéndoles cursos de posgrado; a los docentes, proveyéndoles concursos para sus cargos; y a los no docentes, registrando su asistencia.

La prestación de los servicios se realiza a través de diferentes canales. Los canales de entrega se clasifican en: 1) tradicionales – como por ejemplo, mostrador, teléfono, o fax; y 2) electrónicos – como por ejemplo, un sitio web, mail, dispositivos móviles, y redes sociales, entre otros. Dada la alternativa de canales, es importante destacar que no todos los servicios pueden ser entregados a través de todos los canales y que la selección del canal influye en el costo de entrega del servicio, el tiempo de respuesta para la entrega del servicio, así como en la satisfacción del receptor del servicio.

2.3 Niveles de Madurez de Servicios Universitarios

El nivel de madurez de los servicios públicos puede ser evaluado en base al grado de automatización y al soporte que dan las TICs al proceso que entrega el servicio. Naciones Unidas propone un modelo de madurez de cuatro niveles¹:

- 1) *Emergente* – Los sitios de gobierno proveen información de políticas públicas, gobernanza, leyes, reglamentaciones y los tipos de servicios públicos que se proveen. Poseen links a ministerios, departamentos y otras ramas de gobierno. Los ciudadanos pueden obtener información actualizada del gobierno nacional y sus ministerios y pueden seguir links a información archivada.
- 2) *Mejorado* – Los sitios de gobierno entregan servicios mejorados de una vía o de simples comunicaciones de dos vías en línea entre el gobierno y el ciudadano, tales como descarga de formularios para servicios de gobierno y aplicaciones. Los sitios tienen recursos de audio y video, y son multilingüe. Algunos servicios electrónicos

¹ <https://publicadministration.un.org/egovkb/en-us/Reports/UN-E-Government-Survey-2014>

limitados permiten a los ciudadanos enviar pedidos para formularios no digitales o información personal.

- 3) *Transaccional* – Los sitios de gobierno involucran a los ciudadanos en una comunicación de dos vías, incluyendo solicitud y recepción de información en políticas de gobierno, programas, regulaciones, etc. Se requiere alguna forma de autenticación electrónica de la identidad del ciudadano a fin de completar satisfactoriamente el intercambio. Los sitios de gobierno procesan transacciones no-financieras, por ejemplo, completar formularios de impuestos, solicitar certificados, licencias y permisos. Asimismo, manejan transacciones financieras – es decir, transferencia de dinero en una red segura.
- 4) *Integrado* – Los sitios de gobierno cambian la forma en que los gobiernos se comunican con sus ciudadanos. Son proactivos en pedir información y opiniones de los ciudadanos usando herramientas de la Web 2.0 u otras herramientas interactivas. Los servicios electrónicos cruzan los departamentos y los ministerios de una manera fluida, información, datos y conocimiento son transferidos entre agencias de gobierno a través de aplicaciones integradas. Los gobiernos se han convertido de un enfoque “centrado-en-el-gobierno” a uno “centrado-en-el-ciudadano”, donde los servicios electrónicos están dirigidos a los ciudadanos a través de eventos del ciclo de vida y grupos segmentados para proveer servicios hechos a medida.

Adaptando las definiciones anteriores al alcance de los servicios EGOV-U, definimos el siguiente modelo de madurez para estos servicios:

- 1) *Emergente* – Los sitios de la universidad proveen información de políticas universitarias, gobernanza, reglamentaciones y los tipos de servicios universitarios que se proveen. Poseen links a sus unidades académicas, administrativas y otras universidades. El público puede obtener información actualizada de la universidad y sus dependencias y puede seguir links a información archivada.
- 2) *Mejorado* – Los sitios de la universidad entregan servicios mejorados de una vía o de simples comunicaciones de dos vías en línea entre la universidad y los miembros de su comunidad, tales como descarga de formularios para servicios universitarios y aplicaciones. Los sitios tienen recursos de audio y video, y son multilingüe. Algunos servicios electrónicos limitados permiten a los miembros de la comunidad universitaria enviar pedidos para formularios no digitales o información personal.
- 3) *Transaccional* – Los sitios de la universidad involucran a los miembros de la comunidad universitaria en una comunicación de dos vías, incluyendo solicitud y recepción de información en políticas universitarias, programas académicos, regulaciones, etc. Se requiere alguna forma de autenticación electrónica de la identidad de la persona a fin de completar satisfactoriamente el intercambio. Los sitios de gobierno procesan transacciones no-financieras, por ejemplo, completar formularios de inscripción, solicitar certificados, licencias y permisos. Asimismo, manejan transacciones financieras, como transferencias de dinero por red segura.
- 4) *Integrado* – Los sitios de la universidad cambian la forma en que la universidad se comunica con los miembros de su comunidad. Son proactivos en pedir información y opiniones a los miembros de la comunidad universitaria usando herramientas de la Web 2.0 u otras herramientas interactivas. Los servicios electrónicos cruzan las

unidades académicas y administrativas de una manera fluida, información, datos y conocimiento son transferidos entre dependencias a través de aplicaciones integradas. Las universidades se han convertido de un enfoque “centrado-en-la institución” a uno “centrado-en-el-usuario”, donde los servicios electrónicos están dirigidos a los miembros de la comunidad universitaria a través de eventos del ciclo de vida y grupos segmentados para proveer servicios hechos a medida.

3 Trabajos Relacionados

En [4], se presenta el estado del arte en temas de gobierno electrónico y su aplicación al ámbito universitario. El estudio se basa en el análisis de publicaciones científicas incluidas en la base de datos de Scopus. Para ello, se realizó una búsqueda utilizando los siguientes términos - “e-government”, “electronic government”, y “digital government”. Se buscaron publicaciones donde aparecieran esos términos en el título, resumen, o palabras claves, limitando los artículos que habían sido publicados entre el 2010 y el 2015. Los resultados se analizaron teniendo en cuenta la distribución en los años, las publicaciones por países, en particular los de la región, la relación con el área de ciencias de la computación y la calidad del software.

En base al resultado de esta búsqueda, se percibe que las publicaciones han logrado su auge en los años 2010-2011 con un poco más de 1000 publicaciones al año, decreciendo de forma constante hasta el 2015 con un poco más de 600 publicaciones. Los países que más publican en el área son China y Estados Unidos; y dentro de la región, Brasil. El 63,9% de las publicaciones, están producidas por investigadores de Ciencias de la Computación, lo que demuestra una importante relación entre el concepto del gobierno electrónico y los servicios informáticos.

Al especificar la búsqueda en los modelos de calidad y modelos de evaluación, se obtuvieron sólo 51 publicaciones que relacionaban el gobierno electrónico con los modelos de calidad y 33 que lo relacionaban con modelos de evaluación. Sólo 20 de estas publicaciones estaban relacionadas con temas de las Ciencias de la Computación.

Los modelos de calidad propuestos por los diferentes autores, [5]–[11] coinciden en definir las características de evaluación basados en diferentes agrupaciones de los clásicos criterios de Mc Call [12]. Sólo [9] y [12] mencionan la utilización de estándares ISO como modelo de calidad; y [13], [14] refieren a otros modelos preexistentes. En [15] se propone el uso de lógica difusa para la evaluación. Ocho de las publicaciones [5]–[9], [13], [14], [16] hacen mención al objetivo principal de la calidad, que es obtener la satisfacción del ciudadano; incluso algunos autores describen los conceptos más relevantes para lograr el objetivo. El principal objeto de estudio de la aplicación de los modelos de calidad han sido entidades gubernamentales, como por ejemplo en organismos públicos [15], [17], [18], pero ninguno relacionado con el ámbito universitario.

Se puede apreciar que hay escasa información de gobierno electrónico aplicado en el ámbito universitario. Esto presenta un importante desafío para la realización de un estudio de diagnóstico sobre la calidad y madurez de los servicios universitarios.

4 Servicios EGOV-U

Esta sección presenta la metodología para evaluar el nivel de madurez de los servicios universitarios. Primero, se realiza una identificación de servicios prestados por una UA para cada tipo de receptor. Luego, se explica paso a paso las tareas a realizar para el análisis de los mismos. Por último, se discute la aplicación de la metodología en la Facultad de Informática de la UNLP.

4.1 Identificación de Servicios

Las UAs de una universidad prestan una importante cantidad de servicios a los diferentes miembros de su comunidad. A fin de evaluar el nivel de madurez de los mismos, se identificó para cada uno de los posibles receptores, un subconjunto que se considera indispensable a los objetivos de una UA. Por ejemplo, servicios identificados para los alumnos incluyen: brindar información sobre el calendario académico, permitir la inscripción de alumnos a cursar asignaturas y elegir a representantes de claustro, entre otros. La Figura 1 muestra la lista completa de servicios identificados para cada receptor.

4.2 Análisis de Servicios

Los pasos para el análisis incluyen: 1) determinación de los servicios, 2) descripción de los servicios, 3) documentación de los servicios, 4) evaluación de la calidad, y 5) determinación del nivel de madurez. Las etapas se explican a continuación.

Determinación de los servicios

En el marco de los servicios universitarios, se considera UA a aquella dependencia universitaria (universidad, facultad, escuela, departamento, etc.) que tenga autonomía para regular las políticas de los servicios que ofrece.

El objetivo de esta actividad es identificar los servicios que cada UA ofrece a los diferentes miembros de su comunidad. Cada UA, debe seleccionar de la lista de servicios identificados, los que presta en la actualidad y los que considera relevantes para su comunidad.

Descripción de los servicios

Las UAs deberán brindar una breve descripción para cada servicio seleccionado en el paso anterior y el nombre del responsable por la entrega del mismo. La información provista en esta etapa permite identificar los servicios a analizar por cada UA y los responsables que deberán proveer una documentación más detallada del servicio. En la Tabla 1 podemos ver la descripción de los servicios 1-2 y 8 que reciben los alumnos de la Facultad de Informática de la UNLP.

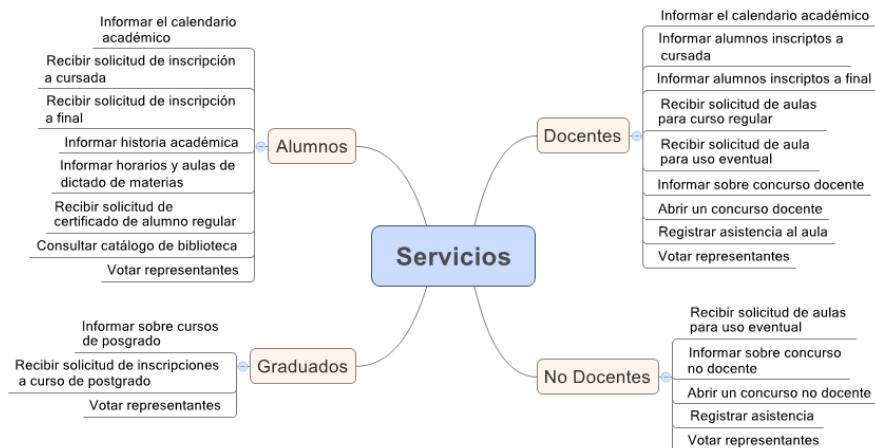


Fig. 1. Servicios identificados para cada miembro de la comunidad universitaria

Tabla 1. Ejemplo de descripción de servicios

Servicio	Descripción	Responsable	
1	Informar el calendario académico	La UA pone a disposición de los alumnos el calendario académico para el ciclo lectivo aprobado por el HCD	Secretaría Académica
2	Recibir solicitud de inscripción a cursadas	La UA recibe la solicitud de un alumno a inscribirse a una asignatura, se analiza si el alumno cumple las condiciones para lo solicitado y se acepta o rechaza la inscripción. La UA le informa el resultado de la solicitud	Oficina de Alumnos
8	Votar a representantes	La UA informa los periodos de elección y publica las listas de candidatos. Realiza la elección según las reglamentaciones vigentes e informa el resultado.	Junta Electoral

Documentación de los servicios

Para cada uno de los servicios descritos por la UA, el responsable de su prestación deberá indicar todos los canales de entrega del mismo. En caso que el proceso de entrega tenga algún nivel de automatización, se indicará la aplicación de software que se utiliza. En caso que el proceso sea manual, se completará si existiera, el tipo de soporte que se utiliza. Adicionalmente y en caso que corresponda, se debe indicar la regulación que reglamenta la prestación del servicio.

La información que se provee en esta etapa permite determinar el máximo grado de automatización que puede alcanzar el servicio en cuestión, separándolo del grado de automatización en el cual se presta el servicio actualmente. En algunos casos el grado de automatización está limitado principalmente por el cumplimiento de regulaciones legales que requiere la prestación del servicio. Por ejemplo, la Tabla 2 muestra la documentación para el servicio de informar el calendario académico a los alumnos.

Tabla 2. Documentación del servicio – Informar calendario académico

1 – Informar el calendario académico		
Descripción	La UA pone a disposición de los alumnos el calendario académico para el ciclo lectivo aprobado por el HCD	
Regulación	Calendario Académico	
Canal	Medio de Acceso	Descripción
UA	Cartelera	
Web	http://www.info.unlp.edu.ar/	Se publica el calendario en detalle
Mail	No	
IM	No	
APP	InfoAPP	Desde la app de la facultad se puede consultar el calendario académico
Facebook	Facultad de Informática UNLP	Se publica al inicio del año
Twitter	@informaticaUNLP	Se publica al inicio del año

Evaluación de la calidad de la prestación de los servicios

El receptor del servicio completa un formulario para evaluar la prestación del servicio, indicando: el canal usado para recibir el servicio; si ha podido resolver su inquietud en el momento, o quedó a la espera de una respuesta, y en este caso, el plazo en que ha obtenido la misma; y el nivel de satisfacción que ha percibido en la utilización del servicio. La Tabla 3 muestra un formulario para evaluar la calidad del servicio.

Determinación del nivel de madurez de los servicios

En base a la información completada en los pasos previos, en esta etapa se determina cual es el nivel de madurez deseado (máximo a obtener) y el actual para cada servicio, considerando la automatización alcanzada en todos los canales de entrega. Los niveles se determinan en base a la clasificación presentada en la Sección 2. A modo de ejemplo, la Tabla 4 ilustra la determinación del nivel de madurez de tres servicios.

4.3 Aplicación de la Metodología

La metodología propuesta se aplicó en la Facultad de Informática de la UNLP. Se evaluaron un total de 25 servicios - 8 para alumnos, 3 para graduados, 9 para docentes, y 5 para no docentes. En función del análisis, 15 servicios podrían llegar como máximo a nivel Mejorado y 10 al nivel Transaccional. De la evaluación realizada, se concluyó que 17 servicios llegaron al nivel máximo, asimismo, 6 tenían como nivel deseado el Transaccional y alcanzaron el nivel Mejorado; 1 se esperaba a nivel Mejorado y se encuentra en nivel Emergente; y 1 que debía estar en nivel Mejorado, no usa ningún canal electrónico para su entrega.

Tabla 3. Evaluación de la Calidad de la Prestación del Servicio

1 - Información del calendario académico								
Unidad académica: Facultad de Informa - UNLP	Receptores: Alumnos							
1 - Por favor seleccione los medios por los que recibió información sobre el calendario académico	<table border="1"> <tr> <td>En UA</td> <td>Web</td> <td>Mail</td> <td>IM</td> <td>APP</td> <td>Facebook</td> <td>Twitter</td> </tr> </table>	En UA	Web	Mail	IM	APP	Facebook	Twitter
En UA	Web	Mail	IM	APP	Facebook	Twitter		
1.1 Si la recibió por diferentes medios seleccione cual le fue de mayor utilidad	<table border="1"> <tr> <td>En UA</td> <td>Web</td> <td>Mail</td> <td>IM</td> <td>APP</td> <td>Facebook</td> <td>Twitter</td> </tr> </table>	En UA	Web	Mail	IM	APP	Facebook	Twitter
En UA	Web	Mail	IM	APP	Facebook	Twitter		
2 - Por favor indique su nivel de satisfacción al utilizar el canal seleccionado	1 - 2 - 3 - 4 - 5 1 Malo 2 Regular 3 Bueno 4 Muy bueno 5 Excelente							
3 - La operación se resolvió en el momento (recibió respuesta automáticamente de la operación realizada)	Si - No							
3.1 - En el caso de no haber obtenido una respuesta en el momento por favor indique cuanto demora y porque medio le comunicaron la respuesta	Demora: ___Hs o ___Días <table border="1"><tr><td>En UA</td><td>Web</td><td>Mail</td><td>IM</td><td>APP</td><td>Facebook</td><td>Twitter</td></tr></table>	En UA	Web	Mail	IM	APP	Facebook	Twitter
En UA	Web	Mail	IM	APP	Facebook	Twitter		
4 - Indique su nivel general de utilización del servicio	1 - 2 - 3 - 4 - 5 1 Malo 2 Regular 3 Bueno 4 Muy bueno 5 Excelente							

Tabla 4. Determinación del Nivel de Madurez de Servicios

	Servicio	Nivel Deseado	Nivel Actual
1	Informar el calendario académico	Mejorado El objetivo del servicio es brindar información actualizada por diferentes canales. No es necesario realizar transacciones para este servicio.	Mejorado Posee la información actualizada por varios canales, permite realizar descargas, contactarse por mail y realizar búsquedas.
2	Recibir solicitud de inscripción a cursadas	Transaccional El objetivo del servicio es registrar la recepción de la solicitud, analizar si es aceptada o rechazada.	Transaccional El sistema permite a los alumnos solicitar inscripciones a asignaturas, en función de las correlatividades. Las acepta o rechaza automáticamente.

5 Conclusiones

EGOV-U aplica los conceptos de gobierno electrónico al ámbito universitario. En este trabajo, se introdujeron conceptos de EGOV-U y se presentó una metodología que permite hacer un diagnóstico de la madurez de los servicios que brindan las universidades, en el contexto de EGOV-U. La metodología se aplicó a la Facultad de Informática de la UNLP. Para realizar el análisis, se seleccionaron 25 servicios, considerados indispensables para la Facultad. Las UA documentaron los servicios en función a las reglamentaciones vigentes, definiendo el nivel máximo de automatización que puede alcanzar el servicio y se determinó el nivel de madurez obtenido (actual).

Del estudio realizado, el 60 % de los servicios prestados por la Facultad podrían llegar a un nivel Mejorado y el 52% de los servicios obtuvo ese nivel. El 40% podría llegar a un nivel máximo Transaccional y sólo el 40% de estos lo ha alcanzado. El resto de los servicios poseen un nivel de madurez inferior al deseado. Trabajos futuros incluyen el análisis de servicios prestados por unidades administrativas y el estudio de mecanismos de innovación en la prestación de servicios universitarios.

6 Referencias

- [1] S. de P. U. SPU, "Síntesis de Información Estadísticas Universitarias Argentina." 2015.
- [2] Republica Argentina, "Ley Nro 24521 - Ley de Educacion Superior," 1995.
- [3] UNLP, "Estatuto de la UNLP," 2008.
- [4] A. Pasini and P. Pesado, "Quality Model for e-Government Processes at the University Level: a Literature Review," *Proc. 9th Int. Conf. Theory Pract. Electron. Gov.*, pp. 436–439, 2016.
- [5] D. Bhattacharya, U. Gulla, and M. P. Gupta, "E-service quality model for Indian government portals: citizens' perspective," *J. Enterp. Inf. Manag.*, vol. 25, no. 3, pp. 246–271, 2012.
- [6] X. Papadomichelaki and G. Mentzas, "Analysing e-government service quality in Greece," *Electron. Gov. an Int. J.*, vol. 8, no. 4, p. 290, 2011.
- [7] S. Funilkul, W. Chutimaskul, and V. Chongsuphajaisiddhi, *Electronic Government and the Information Systems Perspective*, vol. 6866. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.
- [8] P. Saha, A. Nath, and E. Salehi-Sangari, *Electronic Government*, vol. 6228. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.
- [9] M. A. Alanezi, A. K. Mahmood, and S. Basri, "A proposed model for assessing e-government service quality: An E-S-QUAL approach," in *2012 International Conference on Computer & Information Science (ICIS)*, 2012, vol. 1, pp. 130–135.
- [10] A. Sivaji, N. Abdollah, S. S. Tzuaan, C. N. Khean, Z. M. Nor, S. H. Rasidi, and Y. S. Wai, "Measuring public value UX-based on ISO/IEC 25010 quality attributes: Case study on e-Government website," in *2014 3rd International Conference on User Science and Engineering (i-USEr)*, 2014, pp. 56–61.
- [11] E. Ziemba, T. Papaj, and D. Descours, "Assessing the quality of e-government portals – the Polish experience," in *2014 Federated Conference on Computer Science and Information Systems, FedCSIS 2014*, 2014, pp. 1259–1267.
- [12] J. a. McCall, P. K. Richards, and G. F. Walters, "Factors in Software Quality," *Nat'l Tech. Inf. Serv.*, vol. 1, 2 and 3, no. ADA049055, 1977.
- [13] S. C. Misra and J. Chatterjee, "Applying Gap Model for Bringing Effectiveness to e-Government Services:," *Int. J. Electron. Gov. Res.*, vol. 9, no. 3, pp. 43–57, Jan. 2013.
- [14] S. W. Liang, H. P. Lu, and T. K. Kuo, "A study on using the kano two-dimensional quality model to evaluate the service quality of government websites," *J. Internet Technol.*, vol. 15, no. 2, pp. 149–162, 2014.
- [15] G. Yucel and A. F. Ozok, "Quantifying ergonomic quality of governmental websites," *Electron. Gov. an Int. J.*, vol. 7, no. 3, p. 233, 2010.
- [16] F. Sa, A. Rocha, and M. P. Cota, "Quality models for online e-Government services," in *2014 9th Iberian Conference on Information Systems and Technologies (CISTI)*,

2014, pp. 1–5.

- [17] E. Loukis, K. Pazalos, and A. Salagara, “Transforming e-services evaluation data into business analytics using value models,” *Electron. Commer. Res. Appl.*, vol. 11, no. 2, pp. 129–141, Mar. 2012.
- [18] E. Loukis, K. Pazalos, and A. Salagara, “Structuring e-services evaluation based on multi-level value flow models,” in *Proceedings of the European, Mediterranean and Middle Eastern Conference on Information Systems - Informing Responsible Management: Sustainability in Emerging Economies, EMCIS 2011*, 2011, pp. 552–570.

Computer Aided Verification of Relational Models

Claudia M. Necco¹, José N. Oliveira², Joost Visser³, and Roberto Uzal¹

cnecco@gmail.com jno@di.uminho.pt j.visser@sig.eu ruzal@uolsinectis.com.ar

¹ Dep. de Informática, Univ. Nacional de San Luis, San Luis, Argentina

² HASLab/INESC TEC & University of Minho, Portugal

³ Software Improvement Group & CWI, Amsterdam, The Netherlands

Abstract. Binary relational algebra provides semantic foundations for major areas of computing, such as database design, state-based modeling and functional programming. Remarkably, static checking support in these areas fails to exploit the full semantic content of relations. In particular, properties such as the simplicity or injectivity of relations are not statically enforced in operations such as database queries, state transitions, or composition of functional components.

When data models, their constraints and operations are represented by point-free binary relational expressions, proof obligations can be expressed as inclusions between relational expressions. We developed a type-directed, strategic term rewriting system that can be used to simplify relational proof obligations and ultimately reduce them to tautologies. Such reductions can be used to provide extended static checking for design constraints commonly found in software modeling and development. .

Keywords: Models verification, Symbolic execution; Abstract model verification; Extended static checking; Strategic term rewriting

1 Introduction

Software design is error-prone. The negative impact of programming errors on software productivity can be limited by catching them early. Static checkers (e.g. syntax and type checkers) are tools which catch errors at compile-time, i.e. before running the program. Examples of such errors are unmatched parentheses (wrong syntax) and adding integers to booleans (wrong typing). Errors such as null dereferencing, division by 0, and array bound overflow, are not caught by standard static checking; detecting their presence requires extensive testing, and if their presence can not be excluded with certainty, they must be handled at run-time via exception mechanisms.

Software formalists will argue that error checking in the coding phase is too late: first a formal model should be written, queried, reasoned about, and possibly animated (using e.g. a symbolic interpreter). Formal modeling relies on “rich” datatypes such as finite mappings, finite sequences, and recursive data structures, which abstract from much of the complexity found in common imperative programming languages (e.g. pointers, loop boundaries). However, such rich structures are not able to capture *all* properties, meaning that additional constraints need to be added to models such as invariants (attached to types) and pre-conditions (attached to operations). Checking such

constraints is once again a process which falls outside standard static type-checking, leading to a so-called *dynamic* type checking process, typical of model animation tools such as the VDMTools system [8].

Static checking of formal models involving such constraints is a complex process, relying on generation and discharge of proof obligations [11]. While proof obligations can be generated mechanically, their discharge is in general above the *decidability ceiling* in requiring full-fledged formal verification (theorem proving) [16]. Between these two extremes of standard, cheap, decidable static checking and costly theorem proving, *extended static checking* (ESC) [9] aims to catch more errors at compile-time at the relatively moderate cost of adding annotations to the code which record *design decisions* which were lost throughout the programming process (if ever explicitly recorded).

Extended static checking tools have been developed for imperative programming languages such as Java (ESC/Java [9]). At the heart of these tools we find a verification condition generator and the Simplify theorem prover [7]. Verification conditions are predicates in first-order logic which are computed in weakest precondition style. Theorem proving is performed by a combination of techniques, including SAT solvers, matching algorithms, and heuristics to guide proof search.

In the current paper we follow the spirit of this approach but intend to apply it much earlier in the design process: we wish to perform extended static checking at abstract model level to catch errors higher on the semantic scale.

The main novelty of our approach resides in the chosen method of proof construction, whereby first-order proof obligations are subject to the *PF-transform* [18] before they are reasoned about. (See reference [18] for the theory behind this blending of ESC with the PF-transform, suggestively referred to as the ESC/PF proof obligation calculus.) Such a transformation eliminates quantifiers and bound variables and reduces complex formulæ to algebraic expressions which are more agile to calculate with (see Fig. 1 for details). As shown in [18], ESC proof obligations can be discharged at PF-level, leading to the so-called ESC/PF calculus. In the current paper we move from “paper and pencil” ESC/PF reasoning to mechanical calculation using a Haskell implementation of strategic term rewriting [20, 15, 14].

ϕ	$PF \phi$	
$\langle \exists a :: b R a \wedge a S c \rangle$	$b(R \cdot S)c$	<i>In analogy to the well-known Laplace transform [12], the PF-transform takes expressions from a mathematical problem space, in this case first order logic formulæ, into a mathematical solution space, in this case relational algebra expressions [2]. The PF-transform eliminates quantifiers and bound variables (so-called points), resulting in a pointfree notation which is more agile to calculate with.</i>
$\langle \forall a, b :: b R a \Rightarrow b S a \rangle$	$R \subseteq S$	
$\langle \forall a :: a R a \rangle$	$id \subseteq R$	
$b R a \wedge c S a$	$(b, c)\langle R, S \rangle a$	
$b R a \wedge b S a$	$b(R \cap S) a$	
$b R a \vee b S a$	$b(R \cup S) a$	
$b = a$	$b id a$	
TRUE	$b \top a$	
FALSE	$b \perp a$	

Fig. 1. The PF-transform.

In Section 2 we will motivate our extended static checking approach with a small modeling example. In Section 3 we recapitulate binary relation theory which can be

used to capture the semantics of models with rich data structures and their operations. In Sections 4 and 5 we will demonstrate how the algebraic laws of the theory can be harnessed in a strategic term rewriting system, implemented in the functional programming language Haskell. In Section 6 we revisit the model operations of our example to show how our rewriting system is capable of generating the appropriate proof obligations and simplify or discharge them. Section 7 discusses related work and Section 8 concludes.

2 Motivating example

The UML class diagram in Fig. 2 depicts a simplified model of a system for trading non-consumable (uniquely identifiable) items. A user can put an item for sale for a given price, and other users can express their interest in these items for a price they are willing to pay. If a match between a seller and a buyer is established, this leads to a deal with an agreed price.

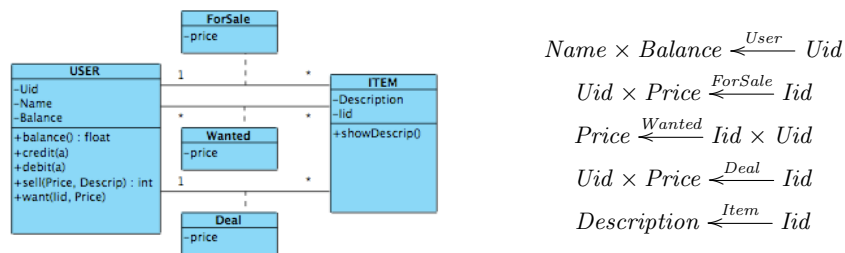


Fig. 2. Simplified UML model of a trading system and the corresponding binary relational model. The relations in this model are finite and simple (explained in Section 3). This is loosely based on a formal model (written in Haskell) for a real estate exchange market, which has been developed for a digital city consortium.

The specification of queries, predicates, and transformations on this model may present some pitfalls. Suppose the following operations are desired:

$$\begin{aligned}
 listWantedItems &:: Wanted \rightarrow Map\ Id\ Price \\
 putBatchForSale &:: (Uid, Map\ Id\ Price) \rightarrow ForSale \rightarrow ForSale \\
 settleDeal &:: (Id, Uid, Price) \rightarrow Deal \rightarrow Deal
 \end{aligned}$$

The *listWantedItems* query produces a map of item identifiers together with the price that has been offered for them. The transformation *putBatchForSale* adds a batch of items belonging to a given user to the *ForSale* relation. The *settleDeal* transformation adds an entry to the *Deal* collection.

When specifying these operations, the designer could benefit from the feedback of an extended static checker. For example, the checker should tell her/him that query *listWantedItems* should only return a map if the *Wanted* collection contains no two offers for the same item with different prices. Rather than adding a precondition to that effect, he will likely decide to change the return type to a general relation $Rel\ Id\ Price$ or, equivalently, to $Set\ (Id, Price)$. In case of the *settleDeal* operation, to ensure that

pre-existing deals do not get lost the checker should indicate that a precondition is needed that either no deal yet exists for the given item, or that it exists with the same buyer identifier and price.

3 Overview of relation theory

In this section we provide a brief introduction to the theory of binary relations [2].

Relations. Let $B \xleftarrow{R} A$ denote a binary relation R on data-types A (source) and B (target). We write bRa to mean that pair (b, a) is in R . The underlying partial order on relations is written $R \subseteq S$, with the usual semantics of the subset relation between sets of pairs. In relational terms, it means that S is more defined or less deterministic than R , that is, $R \subseteq S \equiv bRa \Rightarrow bSa$ for all a, b . $R \cup S$ denotes the union of two relations and \top is the largest relation of its type. Its dual is \perp , the smallest such relation. The identity id relates every element to itself. Equality on relations can be established by \subseteq -antisymmetry: $R = S \equiv R \subseteq S \wedge S \subseteq R$.

Three more operators are introduced to combine relations: composition ($R \cdot S$), converse (R°) and meet ($R \cap S$). R° is such that $a(R^\circ)b$ iff bRa holds. Meet corresponds to set-theoretical intersection and composition is defined in the usual way: $b(R \cdot S)c$ holds wherever there exists some mediating a such that $bRa \wedge aSc$.

Coreflexives. An endo-relation $A \xleftarrow{R} A$ is referred to as *reflexive* iff $id \subseteq R$ holds, and as *coreflexive* iff $R \subseteq id$ holds. Coreflexive relations, which we denote by Greek letters (Φ, Ψ , etc.), are fragments of the identity relation that model predicates or sets. A predicate p is modeled by the coreflexive $\llbracket p \rrbracket$ such that $b\llbracket p \rrbracket a \equiv (b = a) \wedge (p a)$ holds, that is, the relation that maps every a which satisfies p onto itself. Negation is modeled by $\neg\Phi = id - \Phi$. A set $S \subseteq A$ is modeled by $\llbracket \lambda a. a \in S \rrbracket$, that is $b\llbracket S \rrbracket a \equiv (b = a) \wedge a \in S$.

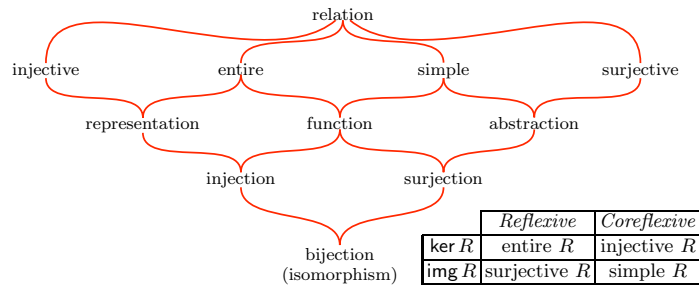


Figure 1: Binary relation taxonomy
Fig. 3. Binary relation taxonomy

Taxonomy. To establish a fundamental taxonomy of relations (illustrated in Fig. 3), let us first define the *kernel* of a relation, $\ker R = R^\circ \cdot R$ and its dual, $\text{img } R = \ker (R^\circ) = R \cdot R^\circ$, called the *image* of R . A relation R is said to be *entire* (or total) iff its kernel

Table 1. Some laws of the binary relational algebra.

<i>comp_assoc</i> $(R \cdot S) \cdot T = R \cdot (S \cdot T)$	<i>inv_comp</i> $(R \cdot S)^\circ = S^\circ \cdot R^\circ$
<i>comp_id</i> $R \cdot id = R ; id \cdot R = R$	<i>inv_inv</i> $(R^\circ)^\circ = R$
<i>comp_empty</i> $R \cdot \perp = \perp ; \perp \cdot R = \perp$	<i>inv_union</i> $(R \cup S)^\circ = R^\circ \cup S^\circ$
<i>union_fusion</i> $(R \cup S) \cdot T = (R \cdot T) \cup (S \cdot T)$	<i>corefl_symm</i> $\Phi^\circ = \Phi$
$T \cdot (R \cup S) = (T \cdot R) \cup (T \cdot S)$	<i>corefl_trans</i> $\Phi \cdot \Phi = \Phi$
<i>dom_elim</i> $R \cdot \delta R = R$	<i>const_fusion</i> $\underline{k} \cdot R = \underline{k} \cdot \delta R$
<i>neg_co_comp</i> $\neg \Phi \cdot \Phi = \perp ; \Phi \cdot \neg \Phi = \perp$	<i>not_dom_cancel</i> $R \cdot \neg (\delta R) = \perp$
<i>incl_empty</i> $\perp \subseteq R \Leftrightarrow True$	<i>incl_refl</i> $R \subseteq R \Leftrightarrow True$
<i>monotonicity</i> $R \cdot \Phi \cdot S \subseteq T \Leftrightarrow R \cdot S \subseteq T$	
<i>union_univ</i> $R \cup S \subseteq T \Leftrightarrow (R \subseteq T \wedge S \subseteq T)$	
<i>shunt_fun</i> $f \cdot R \subseteq S \Leftrightarrow R \subseteq f^\circ \cdot S$	
<i>shunt_fun_inv</i> $R \cdot f^\circ \subseteq S \Leftrightarrow R \subseteq S \cdot f$	
<i>shunt_map_inv</i> $R \cdot M^\circ \subseteq S \Leftrightarrow R \cdot \delta M \subseteq S \cdot M$	
<i>shunt_map</i> $M \cdot R \subseteq S \Leftrightarrow \delta M \cdot R \subseteq M^\circ \cdot S$	
<i>img_def</i> $img R = R \cdot R^\circ$	<i>ker_def</i> $ker R = R^\circ \cdot R$

is reflexive; and *simple* (or functional) iff its image is coreflexive. Simple relations are denoted with capital letters M, N , etc. Dually, R is *surjective* iff $img R$ is reflexive, and R is *injective* iff $ker R$ is coreflexive. This terminology is recorded in the summary table in Fig. 3. The coreflexive fragments of kernel and image are named *domain* (δ) and *range* (ρ).

Functions. As the taxonomy indicates, a relation is a *function* iff it is both simple and entire. Functions will be denoted by lowercase letters (f, g , etc.) and are such that bfa means $b = f a$. The constant function which maps every value of its domain to the value k is denoted by \underline{k} .

Algebraic properties. A rich set of algebraic properties is available for the various operators of relational algebra [2], of which a small sample is listed in Table 1. Of particular interest for the current paper are the various *shunting laws*. They allow the ‘shunting’ of relations (functions and simple relations in the listed cases) from one side of the inclusion to the other, similar to the shunting rules we learned in high school, such as $x - y \leq z \Leftrightarrow x \leq z + y$. The utility of such laws will become evident below.

4 Rewriting relational expressions and propositions

We developed a type-safe, type-directed rewriting system for normalization of relational expressions that harnessed the various algebraic laws of binary relations presented in Table 1 in the functional programming language Haskell. In this section and the next, we will provide a high-level description independent of the programming language.

Terms. The terms to be rewritten by our term rewriting system will be the expression of binary relational calculus with some additional annotations. The following outlines the grammar:

$$\begin{aligned}
 P &:= R \subseteq_T R \mid True \mid P \wedge P \\
 R &:= id \mid R \cdot_T R \mid R^\circ \mid V_L \mid \underline{A}_T \mid \dots
 \end{aligned}$$

$T := 1 \mid Int \mid Bool \mid String \mid [T] \mid T \times T \mid \dots$
 $L := \text{entire} \mid \text{simple} \mid \text{injective} \mid \text{surjective} \mid \text{reflexive} \mid \text{coreflexive}$
 $V := \text{variable names}$
 $A := \text{values}$

Thus, some of the relation operators are annotated with type information (shown as subscript). Relation variables are annotated with properties as they appear in the leafs of the taxonomy of Figure 3. For example, a function f is an entire and simple relation and is therefore annotated as $f_{[\text{entire, simple}]}$. Also, endo-relations can be annotated to be (co-)reflexive.

Predicates on relations. The first ingredient into our rewriting system are predicates for testing the various properties that relations may have, such as simplicity, surjectivity, etc. The various properties declared on relational variables propagate through relational operators. For example, the composition of two surjective relations is surjective, and the inverse of an injective relation is simple. This gives rise to predicates on relations that inductively check their properties. For example:

$isSimple(id) = True$
 $isSimple(r_l) = \text{simple} \in l$
 $isSimple(r^\circ) = isInjective(r)$
 $isSimple(s \cdot_b r) = isSimple(r) \wedge isSimple(s)$
 \dots
 $isSimple(r) = False$

Similar predicates are supplied for the remaining properties. These predicates test for properties by induction over the structure of relational expressions, but do not attempt to derive proofs for the properties. In this sense, they are approximations and may fail to discover that a certain relational expression enjoys particular properties.

Type-directed and property-aware rewriting rules. The predicates above are used in the definition of rewrite rules. Each rewrite rule encodes particular laws of the relational calculus. Since our rewrite system is type-directed, rewrite rules are annotated with types. Here is an encoding of the *inv_comp* law, applied in the left-to-right direction:

$inv_comp : (r \cdot_b s)^\circ \mapsto_{(c \leftarrow a)} s^\circ \cdot_b r^\circ$

Pattern matching is performed on a relational expression and, on successful match, a resulting expression is returned.

The *const_fusion* rule provides an example of rewriting directed by properties:

$const_fusion : (s \cdot_b r) \mapsto_{(c \leftarrow a)} (\underline{s}_c)$
 $\quad \text{if } isConstant(s) \wedge \neg (isCoreflexive(r)) \wedge isEntire(r)$
 $const_fusion : (s \cdot_b r) \mapsto_{(c \leftarrow a)} ((\underline{s}_c) \cdot_a (\delta_b r))$
 $\quad \text{if } isConstant(s) \wedge (\neg (isCoreflexive(r)))$

The rule works on a composition and, if the first argument s constant as required by the guarding predicate, then it replaces the second argument r by its domain. If the second argument r is entire, $\delta r = id$ then the rule return just the first (constant) argument. When r is coreflexive, the rule does not trigger, because the domain of a coreflexive is that relation itself.

An example of a rewrite rule on the level of relational propositions is offered by the shunting rule for functions:

$$\text{shunt_fun_inv} : ((x \cdot_b f^\circ) \subseteq_{(c \leftarrow a)} y) \mapsto_{(c \leftarrow a)} (x \subseteq_{(c \leftarrow b)} (y \cdot_a f))$$

if $\text{isEntire}(f) \wedge \text{isSimple}(f)$

Note the use of a guarding predicate that tests whether the relation f is indeed a function (entire and simple).

Combinators for strategic rewriting. To compose rewriting systems out of individual rewrite rules, we employ the following set of rule combinators known from strategic term rewriting ⁴:

```

nop :: Rule -- identity rule
(▷) :: Rule → Rule → Rule -- sequential comp.
(⊕) :: Rule → Rule → Rule -- choice (based on mplus)
(⊗) :: Rule → Rule → Rule -- choice (bas. on mcatch)
all :: Rule → Rule -- map on all children
one :: Rule → Rule -- map on one child
run :: Rule → R r → (R r, Derivation) -- top-level app.

```

The implementation of each of these combinators is straightforward, and omitted here for brevity. The top-level application function *run* takes the result of rewriting and the derivation (proof trace) out of the *Rewrite* monad; in case of failure it returns the original term and an empty derivation.

Using the basic rule combinators, more sophisticated ones can be defined:

```

many r      = (r ▷ (many r)) ⊗ nop -- repeat until failure
once r      = r ⊗ one (once r) -- apply once, at any depth
innermost r = all (innermost r) ▷ ((r ▷ innermost r) ⊗ nop)

```

The derived combinator *innermost* performs exhaustive rewrite rule application according to the leftmost innermost rewriting strategy.

5 Rewriting strategies

Having defined individual rules and rule combinators, we can proceed to the composition of rewrite systems for various purposes.

Normalization of relational expressions. The following definitions express that a relational expression can be normalized by exhaustive application of individual association, desugaring, and normalization rules:

```

simplify = innermost simplify1
simplify1 = comp_assoc ⊗ desugar1 ⊗ applylaw1
desugar1 = ker_def ⊗ img_def ⊗ ...
applylaw1 = inv_comp ⊗ inv_inv ⊗ comp_id ⊗ comp_empty ⊗ dom_elim ⊗
           coreft_symm ⊗ const_fusion ⊗ not_dom_cancel ⊗ ...

```

We use the convention of postfixing the names of single-step rule combinations with 1 in order to distinguish them from rule combinations that rewrite repetitively until a fixpoint is reached. Note that the *comp_assoc* rule is employed to bring relational compositions into left-associative form. Since the normalization rules together form a

⁴ These rules and our representation technique are inspired on the *2LT* system [6].

confluent and terminating rewrite system, the left-catching combinator \circledast is sufficient to combine them — no need for backtracking.

For example, the following derivation is constructed when applying the *simplify* strategy to $(N \cdot (\neg (\delta N))^\circ \cdot M^\circ)^\circ$, where N and M are simple relations:

$$\begin{aligned}
& (N \cdot (\neg (\delta N))^\circ \cdot M^\circ)^\circ \\
&= \{ \text{corefl_symm} \} \\
& (N \cdot (\neg (\delta N)) \cdot M^\circ)^\circ \\
&= \{ \text{not_dom_cancel} \} \\
& (\perp \cdot M^\circ)^\circ \\
&= \{ \text{comp_empty} \} \\
& \perp^\circ \\
&= \{ \text{corefl_symm} \} \\
& \perp
\end{aligned}$$

This normalization proof trace demonstrates that the original expression is equal to \perp . (Recall that proof traces are generated by our *Rewrite* monad.)

Deriving proofs and proof obligations. We define a more sophisticated strategy to simplify or dispatch proof obligations:

```

derive = simplify ▷ all_and process_conjunct ▷ innermost and_true
  where
    process_conjunct = (shunt_conjunct ⊕ strengthen_conjunct) ∘ nop
    shunt_conjunct   = shunt ▷ derive
    strengthen_conjunct = strengthen ▷ derive ▷ qed
    shunt = (shunt_fun_inv ∘ shunt_map_inv) ⊕ (shunt_fun ∘ shunt_map)
    strengthen = corefl_cancel
    all_and :: Rule → Rule -- apply arg. rule on all conjs
    qed     :: Rule -- test whether the current exp. is True

```

The initial application of *simplify* brings a given proposition into conjunctive normal form, where each conjunct is a normalized relational inclusion. The *all_and* combinator applies *process_conjunct* to all conjuncts. After processing each conjunct separately, *and_true* ($p \wedge \text{True} \Leftrightarrow \text{True} \wedge p \Leftrightarrow p$) is applied to absorb the propositions that have been rewritten to *True*. The processing of each conjunct makes a non-deterministic choice (using the backtracking operator \oplus) between starting with a shunting step (*shunt_conjunct*) or starting with a strengthening step (*strengthen_conjunct*); the conjunct is left unchanged if neither is possible (*nop*). When starting with shunting, the choice between shunting a left-composed relation or shunting a right-composed converse of a relation is again made non-deterministically (*shunt*). After the shunting step, a recursive call is made to the overall *derive* strategy. When starting with strengthening, the subsequent recursive call to *derive* is required to lead to a full proof (*qed*), since we are interested in strengthened propositions only for the purpose of discharging proof obligations.

The use of backtracking entails that several results may be obtained or the same result through different derivations. In the implementation, lazy evaluation is employed to ensure that only a single derivation is actually constructed.

6 Application scenarios

We now explain how our rewriting system can be used in concrete scenarios, such as the ones in our motivation example (Section 2). The overall operation of the developed tool is based on transforming and rewriting PF-relational expressions using the ESC/PF calculus described in [18].

List wanted items. The operation *listWantedItems* can be specified in binary relational terms as $listWantedItems = Wanted \cdot \pi_1^\circ$, where π_1 is the first projection on pairs, i.e. $\pi_1(a, b) = a$. Note that we leave the argument *Wanted* implicit in the definition of the operation. Regarding *Wanted* as a set of pairs, the definition converts to the pointwise $\{ (p, i) \mid (p, (i, u)) \in Wanted \}$, where p, i, u range over *Price*, *Id* and *Uid*, respectively. Clearly, this won't be a simple relation in general, even if *Wanted* is so, because dropping u from the input may lead to the same i related to different p . Since this operation is specified to produce a finite map (thus simple), it gives rise to the proof obligation $img(Wanted \cdot \pi_1^\circ) \subseteq id$, which in turn leads to the following derivation when applying our *derive* strategy:

$$\begin{aligned}
& img(Wanted \cdot \pi_1^\circ) \subseteq id \\
\Leftrightarrow & \{img_def\} \\
& Wanted \cdot \pi_1^\circ \cdot (Wanted \cdot \pi_1^\circ)^\circ \subseteq id \\
\Leftrightarrow & \{inv_comp\} \\
& Wanted \cdot \pi_1^\circ \cdot (\pi_1^\circ)^\circ \cdot Wanted^\circ \subseteq id \\
\Leftrightarrow & \{inv_inv\} \\
& Wanted \cdot \pi_1^\circ \cdot \pi_1 \cdot Wanted^\circ \subseteq id \\
\Leftrightarrow & \{shunt_map_inv\} \\
& Wanted \cdot \pi_1^\circ \cdot \pi_1 \cdot \delta \cdot Wanted \subseteq id \cdot Wanted \\
\Leftrightarrow & \{comp_id\} \\
& Wanted \cdot \pi_1^\circ \cdot \pi_1 \cdot \delta \cdot Wanted \subseteq Wanted \\
\Leftrightarrow & \{shunt_map\} \\
& \delta \cdot Wanted \cdot \pi_1^\circ \cdot \pi_1 \cdot \delta \cdot Wanted \subseteq Wanted^\circ \cdot Wanted
\end{aligned}$$

What does the last line above mean? We simply have to apply the rules of the PF-transform the other way round and find the corresponding, more descriptive logic expression:

$$\begin{aligned}
& \forall x, y. x \in dom(Wanted) \wedge y \in dom(Wanted) \wedge \pi_1(x) = \pi_1(y) \\
& \Rightarrow Wanted(x) = Wanted(y)
\end{aligned}$$

This formula expresses that query *listWantedItems* only returns a finite map if the *Wanted* collection contains no two offers for the same item with different prices. This feedback should lead the designer to broaden the output type of the operation to general binary relations.

Settle deal. Using singleton relation notation as described in Section 3, we can define $settleDeal(i, u, p) = Deal \cup \underline{(u, p)} \cdot \underline{i}^\circ$. (Again we leave the old value of *Deal* implicit in the definition.) Checking the simplicity of its output gives rise to the following derivation (condensed):

$$\begin{aligned}
& img(Deal \cup \underline{(u, p)} \cdot \underline{i}^\circ) \subseteq id \\
\Leftrightarrow & \{img_def, various\ union\ laws\}
\end{aligned}$$

$$\begin{aligned}
& Deal \cdot Deal^\circ \subseteq id \wedge Deal \cdot \underline{i} \cdot \underline{(u, p)}^\circ \subseteq id \wedge \\
& \underline{(u, p)} \cdot \underline{i}^\circ \cdot Deal^\circ \subseteq id \wedge \underline{(u, p)} \cdot \top \cdot \underline{(u, p)}^\circ \subseteq id \\
& \Leftrightarrow \{ \text{various shunting laws, dom_elim} \} \\
& \delta Deal \cdot \underline{i} \subseteq Deal^\circ \cdot \underline{(u, p)} \wedge \\
& \underline{i}^\circ \cdot \delta Deal \subseteq \underline{(u, p)}^\circ \cdot Deal
\end{aligned}$$

Thus, the simplification of this proof obligation leads to an intermediate conjunction of four proof obligations, of which two are subsequently discharged. The remaining two obligations actually express the same property (they can be converted into each other by taking their inverse). Conversion back to pointwise notation gives the following precondition:

$$i \in \text{dom}(Deal) \Rightarrow (u, p) = Deal(i)$$

Note that the proof obligation we derived is weaker than the over-defensive precondition that is typically added to an operation such as *settleDeal*, namely that $i \notin \text{dom}(Deal)$.

Batch addition of items to sell. Once PF-transformed, our last function is defined by $putBatchForSale(u, m) = ForSale \dagger x$, where $x = withUser\ u\ m$ and $withUser\ u\ m = \langle \underline{u}, m \rangle$. This model illustrates the use of two other useful binary operators on relations, *override* ($\cdot \dagger \cdot$) and *split* ($\langle \cdot, \cdot \rangle$) [19]. The latter pairs the outputs of two relations (recall Fig. 1) and the former overrides one relation by another. Checking the simplicity of the output of *putBatchForSale* leads to a 32-step derivation of which we show only the starting and closing steps, the latter condensed for space economy:

$$\begin{aligned}
& \text{img}(n \dagger x) \subseteq id \\
& \Leftrightarrow \{ \text{override_def} \} \\
& \text{img}(n \cup x \cdot \neg(\delta x)) \subseteq id \\
& \Leftrightarrow \{ \text{img_def} \} \\
& (n \cup (x \cdot \neg(\delta(n)))) \cdot (n \cup (x \cdot \neg(\delta(n))))^\circ \subseteq id \\
& \dots \\
& ((True \wedge True) \wedge (True \wedge x \cdot \neg(\delta(n)) \subseteq x)) \\
& \Leftrightarrow \{ \text{and_true, monotonicity} \} \\
& (True \wedge x \cdot id \subseteq x) \\
& \Leftrightarrow \{ \text{and_true, comp_id} \} \\
& x \subseteq x \\
& \Leftrightarrow \{ \text{incl_refl} \} \\
& True
\end{aligned}$$

Thus the proof obligation is discharged completely. In this case extended static checking validates the user model and no changes are needed. The 32-step derivation took 0.14 seconds to run with version 6.8.2 of the Haskell interpreter (GHCi) on a MacBook Pro (1.83 GHz Intel Core Duo processor).

7 Related work

Extended static checking. Extensive progress has been achieved on extended static checking (for review see [16]), resulting in practical tools for imperative languages [9].

These tools rely on theorem provers to find counter examples of verification conditions [7], using a combination of techniques such as backtracking search, matching algorithms for universally quantified formulæ, and heuristics. As alternative or supplemental technique, we have explored proof construction through rewriting of pointfree relational expressions. The absence of quantifiers and variables in these expressions promises to allow a more effective proof search and to enlarge the scope of properties that can be practically checked for, such as those arising in software modeling using rich data structures.

Relational programming (symbolic). MacLennan pioneered relational programming and proposed it as a more general substitute for functional programming [17]. He keeps a separation between finite relations representing data structures, and infinite relations representing operations. Cattrall and Runciman built on his work to develop compilation support for relational programming, where finite and infinite relations are mixed, and where relational expressions are made compilable by rewriting them according to algebraic properties [3].

Relation-algebraic analysis (finite). Modeling and analysis of systems based on *finite* relational representations is supported by systems such as Grok [10] and RelView [1] which are, however, very different from our approach: Grok is a calculator for *finite* relational algebra expressions and RelView uses BDDs to implement relations in an efficient way.

Typed strategic rewriting. Strategic programming [14] was first supported in the non-typed setting of the Stratego language [20]. A strongly-typed combinator suite was introduced as a Haskell library by the Strafunski system [15] and later generalized into the so-called ‘scrap-your-boilerplate’ generic programming library [13]. We developed GADT-based strategic combinator suites, similar to the one presented here, for two-level data transformation [5] and transformation of pointfree and structure-shy functions [4].

8 Concluding remarks

We have implemented a type-directed strategic rewrite system for normalization of pointfree relational expressions and simplification or discharge of relational propositions. We have demonstrated the utility of the system in the context of extended static checking of common model and program properties.

So far, we have limited ourselves to rewriting of pointfree expressions, relying on manual transformation of logic formulæ into relational algebra expressions and back. We intend to also automate this pointfree transform.

The suite of operators and laws implemented in the system is currently under study with respect to minimality, confluence and termination.

The strategy for proof search is likely to further evolve as well, for instance to include short cut derivations for special common cases or to eliminate duplication of proof obligations due to converse inclusions. A thorough analysis of the formal properties of the rewriting system we are building is one of our current concerns.

When achieving a good degree of maturity, an assessment will be needed as to whether this approach can indeed be an alternative or supplement to existing ESC approaches based on theorem proving. A good test will be to try and discharge complex ESC/PF proof obligations such as those arising from the Verified File System project [18]. Besides ESC, we envision to apply our relational algebra rewriting system to areas such as program optimization, program verification, relational programming, and more.

References

1. Rudolf Berghammer and Frank Neumann. *RelView – An OBDD-Based Computer Algebra System for Relations*, pages 40–51. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.
2. R. Bird and O. de Moor. *Algebra of Programming*. Series in Computer Science. Prentice-Hall International, 1997. C.A.R. Hoare, series editor.
3. D. Cattrall and C. Runciman. Widening the representation bottleneck: a functional implementation of relational programming. In *Proc. Func. Prog. Lang. and Comp. Arch.*, pages 191–200. ACM Press, 1993.
4. A. Cunha and J. Visser. Transformation of structure-shy programs, applied to XPath queries and strategic functions. In *PEPM’07, ACM SIGPLAN*, 2007.
5. Alcino Cunha, José Nuno Oliveira, and Joost Visser. *Type-Safe Two-Level Data Transformation*, pages 284–299. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
6. Alcino Cunha and Joost Visser. Strongly typed rewriting for coupled software transformation. *Electron. Notes Theor. Comput. Sci.*, 174(1):17–34, 2007.
7. D. Detlefs, G. Nelson, and J.B. Saxe. Simplify: a theorem prover for program checking. *J. ACM*, 52(3):365–473, 2005.
8. J. Fitzgerald and P.G. Larsen. *Modelling Systems: Practical Tools and Techniques for Software Development*. Cambridge University Press, 1st edition, 1998.
9. C. Flanagan, K.R.M. Leino, M. Lillibridge, G. Nelson, J.B. Saxe, and R. Stata. Extended static checking for Java. In *PLDI*, pages 234–245, 2002.
10. R.C. Holt. Structural manipulations of software architecture using Tarski relational algebra. In *WCRE ’98*, page 210. IEEE Comp. Soc., 1998.
11. C.B. Jones. *Systematic Software Development Using VDM*. Series in Computer Science. Prentice-Hall International, 1986. C.A. R. Hoare.
12. E. Kreyszig. *Advanced Engineering Mathematics*. J.Wiley & Sons, Inc., 1988.
13. R. Lämmel and S. Peyton Jones. Scrap your boilerplate: a practical design pattern for generic programming. *ACM SIGPLAN Notices*, 38(3):26–37, March 2003.
14. R. Lämmel, E. Visser, and J. Visser. The Essence of Strategic Programming. Available at <http://www.cwi.nl/~ralf>, 2003.
15. R. Lämmel and J. Visser. A Strafunski Application Letter. In V. Dahl et al., editors, *PADL’03*, volume 2562 of *LNCS*, pages 357–375. Springer, 2003.
16. K.R.M. Leino. Extended static checking: A ten-year perspective. In *Informatics - 10 Years Back. 10 Years Ahead.*, pages 157–175. Springer-Verlag, 2001.
17. B.J. MacLennan. Overview of relational programming. *SIGPLAN Not.*, 18(3):36–45, 1983.
18. J.N. Oliveira. Extended static checking by calculation using the PF-transform, Jul. 2008. LerNET’08 post-workshop tutorial paper submitted for publication.
19. J.N. Oliveira and C.J. Rodrigues. Transposing relations: from *Maybe* functions to hash tables. In *MPC’04*, volume 3125 of *LNCS*, pages 334–356. Springer, 2004.
20. Eelco Visser. *Stratego: A Language for Program Transformation Based on Rewriting Strategies System Description of Stratego 0.5*, pages 357–361. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.



XIII Workshop Bases de Datos y Minería de Datos (WBDMD)

Comité Científico:

Alfredo Simón Cuevas (Cuba)
Antonio Fariña (España)
Claudia Deco (Argentina)
Cristina Bender (Argentina)
Diego Seco (Chile)
Edgar Chávez (México)
Edilma Olinda Gagliardi (Argentina)
Francisco Pascual Romero (España)
Guillermo Leguizamón (Argentina)
Jesús Serrano (España)
José Ángel Olivas Varela (España)
José Ramón Paramá Gabía (España)
Karina Figueroa (México)
Laura Lanzarini (Argentina)
Leonardo Corbalán (Argentina)
Marcelo Errecalde (Argentina)
Miguel R. Luaces (España)
Nieves R. Brisaboa (España)
Norma Herrera (Argentina)
Óscar Pedreira (España)
Osvaldo Sposito (España)
Regina Motz (Uruguay)
Rodrigo Paredes (Chile)
Waldo Hasperue (Argentina)



XIII Workshop Bases de Datos y Minería de Datos (WBDMD)

ID	Trabajo	Autores		
9114	Diseño de Bases de Datos Basadas en Documento: Modelo de Interrelación de Documentos	Gerardo Rossel (UBA),	Andrea Manna (UBA)	662
9158	Construcción de una Vista Minable para aplicar Minería de Datos Secuenciales Temporales	Oscar Eduardo Quinteros (UNCa),	Ana Funes (UNSL), Hernàn Ahumada (UNCa)	672
9167	Clustering de un flujo de datos usando MapReduce	María José Basgall (UNLP),	Waldo Hasperué (UNLP), César Estrebou (UNLP), Marcelo Naiouf (UNLP)	682
9170	Estimación del Plano Anestésico en Elefante Marinos del Sur Utilizando Técnicas de Machine Learning.	Marcos Zarate (UNPSJB),	Mirtha Lewis (CENPAT)	692
9177	On the Assessment of Information Quality in Spanish Wikipedia	Guido Urquiza (UNSL),	Matías Soria (UNSL), Sebastián Perez Casseignau (UNSL), Edgardo Ferretti (UNSL), Sergio Alejandro Gómez (UNS), Marcelo Errecalde (UNSL)	702
9184	Data Matching and Deduplication Over Big Data Using Hadoop Framework	Pablo Adrián Albanese (UBA),	Juan María Ale (UBA)	712
9225	Proceso de Descubrimiento de Reglas de Caracterización de Grupos Espacialmente Referenciados	Giovanni Róttoli (UTN-FRBA),	Hernan Merlino (UNLA), Ramon Garcia Martinez (UNLA)	721
9227	Proceso de Descubrimiento de Patrones de Co-Localización alrededor de Tipos de Eventos de Referencia	Giovanni Róttoli (UTN-FRBA),	Hernan Merlino (UNLA), Ramon Garcia Martinez (UNLA)	731
9229	Clasificación de Distintos Conjuntos de Datos Utilizados en Evaluación de Métodos de Extracción de Conocimiento Creados para la Web	Santiago Bianco (UNLA),	Sebastian Martins (UNLA), Ramon Garcia Martinez (UNLA)	739
9231	Estudio Comparativo de Algoritmos de Minería de Subgrafos Frecuentes	Santiago Bianco (UNLA),	Sebastian Martins (UNLA), Ramon Garcia Martinez (UNLA)	748



ID	Trabajo	Autores	
9248	Algoritmo de Clustering Basado en el Concepto de Densidad Atómica	Oscar Andrés Monroy Medina (UA)	758
9254	Wild Cetacean Identification using Image Metadata	Débora Pollicelli (UNS), Mariano Coscarella (UNS), Claudio Delrieux (UNS)	765
9255	Búsqueda por Similitud de Posiciones de Ajedrez	Diego Gonzalez (UTN), Andrés Pascal (UTN), Anabella De Battista (UTN), Norma Edith Herrera (UNSL)	775
9289	Vector-based word representations for sentiment analysis: a comparative study	María Paula Villegas (UNSL), María José Garcarena Ucelay (UNSL), Juan Pablo Fernandez (UNSL), Miguel A. Alvarez-Carmona (OTRA), Marcelo Errecalde (UNSL), Leticia Cagnina (UNSL)	785
9306	LSA64: An Argentinian Sign Language Dataset	Franco Ronchetti (UNLP), Facundo Quiroga (UNLP), César Estrebou (UNLP), Laura Lanzarini (UNLP), Alejandro Rosete-Suárez (ISPJAE)	794
9317	Distal Dynamic Spatial Approximation Forest	Edgar Chávez (CICESE), María E. Di Genaro (UNSL), Hernàn Ahumada (UNCa), María José Basgall (UNLP)	804
9320	Propuesta de detección de datos anómalos y ruido en declaraciones juradas públicas	Rodrigo López-Pablos (UNLaM), Horacio Kuna (UNaM)	814
9322	Approximate Nearest Neighbor Graph via Index Construction	Edgar Chávez (CICESE), Verónica Ludueña (UNSL), Nora Reyes (UNSL), Fernando Kasián (UNSL)	824
9327	An exploratory analysis of methods for extracting credit risk rules	Patricia Jimbo Santana (UCE), Augusto Villa-Monte (UNLP), Laura Lanzarini (UNLP), Aurelio F. Bariviera (Universidad de Rovira i Virgili)	834

Diseño de Bases de Datos Basadas en Documento: Modelo de Interrelación de Documentos

Gerardo Rossel, Andrea Manna

Departamento de Computación. FCEyN.
Universidad de Buenos Aires
{grossel, amanna}@dc.uba.ar

Resumen Las bases de datos no relacionales han experimentado en los últimos años un importante crecimiento, particularmente las bases de datos basadas en documento. A partir de la amplia adopción de este tipo de modelo de almacenamiento, aumenta la necesidad de contar con herramientas de modelización adecuadas. En este trabajo presentamos una metodología de modelización conceptual y de diseño de este tipo de bases de datos, para lo cual proponemos además la realización de un tipo de diagrama que denominamos *diagrama de Interrelación de documentos* o *DID*.

Keywords: NoSQL, Bases de datos basadas en documentos, modelización conceptual de datos, diagrama de diseño de documentos, BigData, JSON

1 Introducción

El enorme crecimiento de lo que se ha denominando *Big Data*, entendido como el análisis, procesamiento y almacenamiento de grandes cantidades de datos, que se ha producido en los últimos años ha impactado fuertemente en la tecnología de almacenamiento de datos. Entre los desafíos que se plantean se encuentran: la necesidad de escalar en forma horizontal, el trabajo con fuentes de datos diversas, la falta de esquema o estructura de los datos con los que se trabaja, etc.. Estas demandas, junto a la necesidad de alcance global y disponibilidad permanente, tuvieron como respuesta el surgimiento de una familia de bases de datos, que no se referencian en el modelo relacional, conocidas como NoSQL (en algunos contextos también denominadas *NoSQL datastores*).

Si bien hay una enorme variedad de bases de datos *NoSQL* podemos clasificarlas, entre otras cosas, por la forma de almacenar y recuperar la información[4][5]:

- Clave/Valor (Key-Value).
- Basadas en Documento.
- Familia de columnas.
- Basadas en Grafos.

En este artículo trabajaremos sobre una de las categorías más utilizadas: las bases de datos NoSQL basadas en documentos, una de cuyos exponentes (MongoDB) se ha ubicado en el cuarto lugar de popularidad entre las bases de datos, siendo la más popular entre los sistemas NoSQL[6]. Este tipo de base de datos utilizan un enfoque similar a las bases clave/valor pero con importantes diferencias: los valores son almacenados como documentos en un formato standard como *Extensible Markup Language* (XML) o principalmente como *JavaScript Object Notation* (JSON) y es posible realizar consultas basadas en el contenido de los documentos[5].

Las metodologías tradicionales de diseño y construcción de bases de datos en el mundo relacional han sido ampliamente estudiadas y refinadas por décadas de investigación [1][2][3]. En la Fig. 1 se muestra un flujo simplificado de los pasos que se siguen en el diseño tradicional.

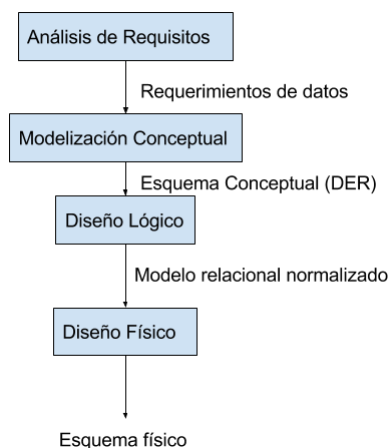


Fig. 1: Pasos generales en el diseño de bases de datos relacionales

La modelización conceptual se presenta como una descripción de los requerimientos respecto de los datos y es realizada usualmente utilizando alguna forma de diagrama de entidad-relación (DER) [2]. En esta fase también se analizan los requerimientos de manipulación de datos de alto nivel que a su vez sirven para la validación del modelo. Posteriormente, se sigue el mapeo del modelo conceptual al modelo relacional y la normalización de éste. Estas tareas forman parte del diseño lógico y tiene aspectos independientes y aspectos dependientes del sistema de gestión base de datos. Los principios de diseño utilizados en el diseño lógico no son aplicables a las bases de datos basadas en documentos. Estas últimas conllevan un conjunto de decisiones basadas en los patrones de consulta que implica muchas veces redundancia y *desnormalización*. En este trabajo proponemos un método de modelización lógica adecuada para este tipo de bases de datos y para ello incorporamos un nuevo tipo de diagrama que llamamos *diagrama de inter-*

relación de documentos o *DID*. Utilizamos la modelización conceptual basada en los diagramas de entidad-relación y sumamos los patrones de consulta que surgen del análisis de requisitos. El resto de este artículo es como sigue: la sección 2 provee un fundamento básico del modelo de documentos, la sección 3 detalla la modelización conceptual y el rol de los patrones de consulta, la sección 4 desarrolla la forma en que a partir del modelo de entidad-relación y los patrones de consulta se construye el modelo de diseño de documentos utilizando el *DID*.

2 Modelo de documentos

La palabra *documento* en el contexto de una base de datos basada en documentos significa una estructura jerárquica organizada de datos. La utilización de *JSON* (o incluso *Binary JSON* o *BSON*) para el almacenamiento de documentos es la forma dominante.

Similar a lo que ocurre con las bases de datos *clave/valor*, se almacenan los documentos en la parte del *valor*. Una de las características comunes en este tipo de base de datos es que son agnósticas respecto al esquema de los documentos. Los documentos similares se agrupan en colecciones, cómo las tuplas en una base de datos relacional se agrupan en tablas. A diferencia de un esquema relacional donde todas las tuplas de una tabla tienen la misma estructura los documentos de una colección pueden diferir en su estructura. Esto permite que a la flexibilidad para almacenar documentos con diferentes estructuras en la misma colección se suma la posibilidad de incorporar cambios en los datos sin necesidad de reestructurar la base de datos. En el listado 1 vemos dos documentos con diferente estructura que pueden ser guardados en la misma colección

Listado 1: Dos documentos con diferente estructura

```
{
  "_id": 1
  "nombre": "Napoleon",
  "apellido": "Solo",
  "genero": "masculino",
  "fecha_nacimiento": "1960-08-03T12:57:32.706000",
  "Direccion": "Calle 2 CABA ARGENTINA",
  "Telefono": "555-5555"
}
{
  "_id": 2
  "nombre": "Jhon",
  "apellido": "Doe",
  "genero": "masculino",
  "fecha_nacimiento": "2003-08-03T12:57:32.706000",
  "Direccion": "Calle 1 CABA ARGENTINA",
  "idiomas": ["ingles", "español"]
}
```

Lo que en principio parece sencillo se complica cuando se debe decidir la forma de relacionar documentos. A diferencia del modelo relacional, no es posible realizar una junta como consulta. Más adelante detallamos los diferentes patrones para modelar la relación entre documentos.

3 Modelización conceptual y diseño de documentos

Nuestra propuesta comienza por la realización de un modelo conceptual de los datos que nos permite obtener conocimiento de los datos que deben administrarse. Para ello optamos por la utilización del diagrama de entidad-relación usando la notación de Chen [2] que además de ser ampliamente conocido en la comunidad de base de datos es una notación independiente de la tecnología e incluso del modelo relacional [7].

La modelización conceptual vinculada al diseño de bases de datos NoSQL es un área que está en desarrollo. Como antecedentes podemos citar la metodología para Apache Cassandra propuesta en [7] que si bien utiliza el DER, está orientada a ese motor específicamente por lo que no es genérica. Otra propuesta sobre modelo conceptual y diseño de NoSQL es la descrita en [10] donde se plantea utilizar los aspectos comunes de las diversas bases NoSQL para realizar una metodología general, donde la modelización conceptual utiliza un modelo abstracto de datos denominado NoAM.

El modelo conceptual en el diseño tradicional es mapeado a un esquema de implementación que se expresa, en general, como un conjunto de esquemas relacionales. En nuestro caso el modelo conceptual debe ser mapeado a un modelo de diseño de documentos. En dicho modelo se representa la estructura general de los documentos en base a colecciones. Para realizar el modelo de diseño de documentos incorporamos un nuevo tipo de diagrama que denominamos diagrama de diseño de documentos y se representa como un extensión del diagrama de entidad-relación al que se le agregan las relaciones propias entre documentos.

Básicamente hay dos formas de relacionar documentos: referenciar o embeber (anidar). La facilidad de embeber documentos permite al diseñador almacenar datos relacionados como un documento simple. De esta manera, se puede romper con lo que se denomina *impedance mismatch*, es decir con la diferencia entre las estructuras de datos en memoria y la forma en que éstos son almacenados [5]. La decisión de cuando embeber o referenciar es una decisión de diseño que debe ser guiada por la forma en que los datos son manipulados. Datos que se acceden regularmente juntos deberían agruparse.

Para poder realizar el diagrama de diseño de documentos a partir del DER, adecuadamente, es necesario ampliar nuestro modelo conceptual con los patrones de consulta y acceso a los datos. Ambas cosas son esenciales y no deben ser subestimadas. Un diseño conceptual equivocado o patrones de consulta errados llevan a un mal diseño de la base de datos. Los patrones de consulta pueden ser escritos en lenguaje informal o en un lenguaje más formal como ERQL [8]. Junto a la especificación de las consultas se debe establecer el peso de éstas en el global. Es decir, no es lo mismo una consulta que se realiza esporádicamente que una

que es realizada con mayor frecuencia. Además, es necesario determinar cuales son las operaciones de actualización/inserción más frecuentes y si la aplicación será de lectura intensiva o no. El análisis transaccional puede ayudar también ya que en este tipo de base de datos la unidad atómica de almacenamiento es el documento. Por último el ciclo de vida del documento pasa a ser de interés en un diseño optimo. Conocerlo nos ayuda a reservar espacio en determinados casos como cuando se almacenan series de tiempo.

El diagrama de la Fig.2 muestra un esquema simplificado de como se modifican los pasos del diseño tradicional .

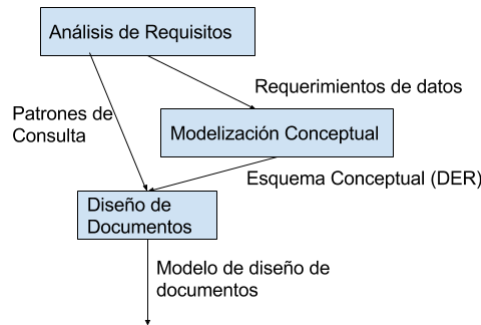


Fig. 2: Pasos en la modelización de documentos

Analizaremos, caso por caso, como llevar el DER al modelo de diseño de documentos en función de los tipos de interrelaciones, su cardinalidad y tomando los patrones de consultas como herramienta para decidir. De este diseño surgen las colecciones que finalmente integrarán la base de datos.

4 Modelo de Diseño de Documentos

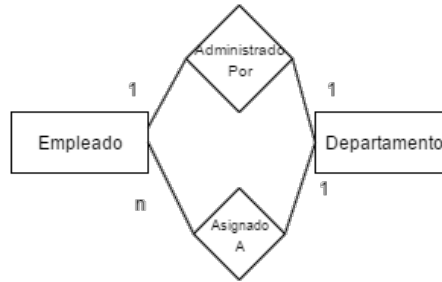
En esta sección describiremos como realizar el modelo de diseño de documentos a partir del DER. Algunos de los patrones que seguiremos son conocidos pero en nuestro caso la decisión de diseño queda expresada en el *DID*. El alcance del artículo no permite una descripción exhaustiva de todos los casos por lo que intentaremos mostrar la semántica y sintaxis del *DID* más allá de la cantidad de variantes de diseño que pueden encontrarse.

4.1 Interrelaciones 1 a 1

En la Fig. 5a vemos un DER, tomado del libro de Jeffrey D. Ullman[9]. Se modelan como entidades separadas los empleados y los gerentes (se omiten los atributos por claridad). En este caso es muy posible que, a menos que los patrones de consulta indiquen otra cosa, lo correcto sea embeber un documento en otro y



(a) Gerente y Empleado como entidades separadas



(b) Gerente modelado por la interrelación

Fig. 3: DER Interrelaciones 1 a 1

como además los departamentos tienen otra interrelación lo ideal es embeber al gerente en el departamento. En la Fig. 5b vemos otro DER que modela a los gerentes como empleados diferenciados por participar de la relación "administrado por".

En definitiva, tenemos varios modelos posibles para el diseño de documentos. La notación utilizada en *DID* mantiene al DER pero las entidades pasan a ser documentos e indicamos la forma de relacionarse. Para el caso de embeber, usamos un símbolo específico en forma de llave que lo indica. La Fig. 4 muestra que el documento *Gerente* es embebido en *Departamento*.



Fig. 4: DID. 1 a 1 embebiendo

4.2 Interrelaciones 1 a N

En las interrelaciones 1 a N los elementos de una entidad E_1 se interrelacionan con uno o más elementos de otra entidad E_2 , mientras que los elementos de E_2 sólo se interrelacionan a un elemento E_1 . En la Fig. 3 vemos que un departamento tiene muchos (N) empleados asignados y un empleado puede ser asignado a

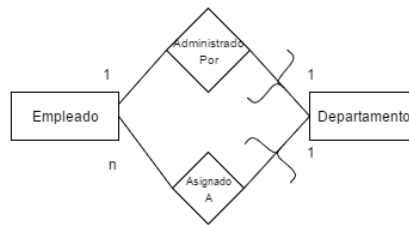
sólo un (1) departamento. También podría haber participación parcial, es decir que existan departamentos que no tienen empleados asignados o que existan empleados que no tengan departamento asignado (esto se indica con una O sobre la línea que vincula la entidad correspondiente con la interrelación).



(a) Empleados Embebidos en Departamento



(b) Departamento embebido en Empleado



(c)

Fig. 5: DID Interrelaciones 1 a N Embebiendo

El diseño de este tipo de interrelación tiene algunas variantes. Un diseño implicaría embeber el documento que representa a E_1 en el documento que representa a E_2 (en el ejemplo sería embeber el documento de Departamento en el documento de Empleado). También es posible que el departamento tenga una lista de documentos que representan Empleados ¿Cuál solución es mejor?. Dependerá del patrón de consultas. La Fig. 5 nos muestra el *DID* con diferentes formas de embeber. Además de embeber se puede referenciar y también es posible combinar ambas. En la Fig. 6 vemos las referencias indicadas como una flecha abierta. En este caso, los departamentos tendrán una lista de referencias a sus empleados y cada documento empleado una referencia a un departamento. La cardinalidad de la interrelación indica la existencia de una lista o no.



Fig. 6: DID. Referencias

4.3 Interrelaciones N a M

En las interrelaciones N a M (muchos a muchos) los elementos de una entidad E_1 se interrelacionan con uno o más elementos de otra entidad E_2 y viceversa. Nuevamente puede también existir una participación parcial u opcional. En la Fig. 7, se muestra el DID (7b) correspondiente a un DER con una interrelación N a M (7a).



(a) DER con interrelación N a M



(b) DID con N a M usando referencias

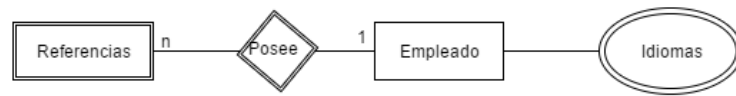
Fig. 7: DID Interrelaciones M a N

4.4 Entidades débiles y atributos multivaluados

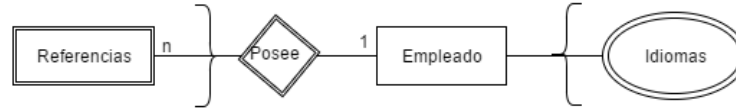
El caso de los atributos multivaluados, aquellos atributos que pueden tener muchos valores para el mismo elemento, y las entidades débiles, que no contienen una clave por sí mismas y dependen de otra entidad para ser identificadas, puede ser modelado embebiendo como muestra en las figuras 8a y 8b

4.5 Desnormalización

La normalización es una técnica de diseño utilizada ampliamente en las bases de datos relacionales. La normalización permite obtener esquemas de relación que minimizan la redundancia y evitan anomalías en la actualización e inserción [9]. El costo de la normalización es la necesidad de utilizar juntas para recuperar la información. En los modelos de documentos se utiliza lo que se denomina desnormalización que implica agregar redundancia para optimizar la lectura a costa de trasladar al programador la responsabilidad de evitar inconsistencias. Hasta aquí hemos mostrado cómo se representan en el modelo de documentos la desnormalización más trivial que es cuando un documento se embebe en otro. Una forma menos trivial y que depende fuertemente del patrón de consultas es embeber sólo algunos atributos de un documento en otro. Es necesario que el



(a) DER. Entidades Débiles y Atributos Multivaluados

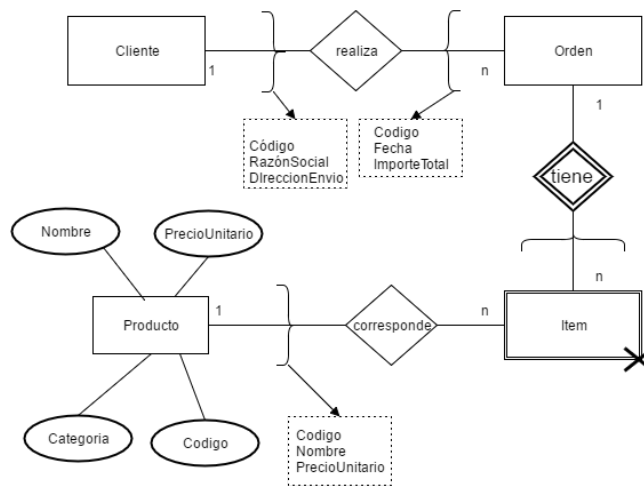


(b) DID. Entidades Débiles y Atributos Multivaluados

Fig. 8: Entidades Débiles y Atributos multivaluados

diagrama permita modelar esta situación lo cual se logra asignando una lista de campos a embeber en el símbolo correspondiente.

Al embeber un documento completo debe decidirse si el documento que se embebe tendrá además una existencia independiente. Esa decisión se indica en el modelo de la siguiente manera: para indicar que un documento *sólo* se almacena embebido y no en forma independiente, se marca con una cruz en el diagrama. En la Fig. 9a se puede ver un diagrama donde se muestra como modelar lo descrito.



(a) Denormalización

Se indica que el documento *Item* solamente será almacenado como embebido en el documento *Orden* y que además contendrá sólo el código, el nombre y el precio unitario del producto. Se ven dos embebidos parciales más, el *Cliente*

mantendrá un vector de ordenes con únicamente el código, el precio y el importe total, mientras que la *Orden* sólo mantendrá el código, la razón social y la dirección de envío del cliente. La figura omite los atributos para facilitar la lectura.

5 Conclusiones

En este artículo introdujimos una nueva forma de llevar a cabo el diseño de bases de datos NoSQL basadas en documentos. El enfoque modifica el método tradicional incorporando los patrones de consulta y nuevo modelo de diseño expresado en un diagrama de diseño de documentos. Hemos mostrado como representar en el diagrama los diferentes esquemas de diseño que pueden realizarse al construir una base de datos basada en documentos.

Se observan dos aspectos destacables:

- Es el primer trabajo sobre modelización conceptual y modelización de documentos utilizando un diagrama específico para modelar documentos.
- La simplicidad del diagrama presentado facilita la comunicación en el grupo de desarrollo y la discusión sobre las decisiones de diseño que se tomen.

En el futuro pretendemos refinar este trabajo a partir de la experiencia de su uso en diversos proyectos de desarrollo.

References

1. E. F. Codd. A relational model of data for large shared data banks. *Commun. ACM* 13, 6 1970, 377-387
2. Peter P. S. Chen. 1975. The entity-relationship model: toward a unified view of data. In *Proceedings of the 1st International Conference on Very Large Data Bases (VLDB '75)*. ACM, New York, NY, USA
3. Peter P. Chen. 2009. Thirty Years of ER Conferences: Milestones, Achievements, and Future Directions. In *Proceedings of the 28th International Conference on Conceptual Modeling (ER '09)*, Alberto H. Laender, Silvana Castano, Umeshwar Dayal, Fabio Casati, and José Palazzo Oliveira (Eds.). Springer-Verlag, Berlin, Heidelberg
4. Adam Fowler: *The State of NoSQL 2016*. Adam Fowler; 1 edition(2016)
5. Pramod J. Sadalage, Martin Fowler. 2012. *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence* (1st ed.). Addison-Wesley Professional.
6. solid IT, DB-Engines Ranking <http://db-engines.com/en/ranking>
7. Artem Chebotko, Andrey Kashlev, Shiyong Lu, A Big Data Modeling Methodology for Apache Cassandra, *IEEE International Congress on Big Data (BigData'15)*, pp. 238-245, New York, USA, 2015.
8. M. Lawley, R. W. Topor. A query language for EER schemas in *Proceedings of the 5th Australasian Database Conference, 1994*, pp. 292-304.
9. Jeffrey D. Ullman. . *Principles of Database and Knowledge-Base Systems, Vol. I*. Computer Science Press, Inc., New York, NY, USA. 1988
10. Francesca Bugiotti, Luca Cabibbo, Paolo Atzeni, Riccardo Torlone. *Database Design for NoSQL Systems*. *International Conference on Conceptual Modeling, Oct 2014, Atlanta, United States*. pp.223 - 231, 2014.

Construcción de una Vista Minable para aplicar Minería de Datos Secuenciales Temporales

Oscar Eduardo Quinteros¹, Ana Funes², Hernán César Ahumada¹

¹ Facultad de Tecnologías y Ciencias Aplicadas - Universidad Nacional de Catamarca,
Catamarca, Argentina,

{oequinteros,hcahumada}@tecnoc.unca.edu.ar

² Universidad Nacional de San Luis

San Luis, Argentina

afunes@unsl.edu.ar

Resumen Sobre datos de exámenes rendidos y aprobados de las materias del Ciclo Común de Articulación (CCA) de las carreras de Ingeniería de la Facultad de Tecnología y Ciencias Aplicadas, se propone conformar la vista minable apropiada para la aplicación de métodos de minería de secuencias temporales, como parte de un proyecto de extracción de conocimiento. El proceso de elaboración de esta vista minable se lleva a cabo siguiendo las actividades de la fase de selección y preparación de datos, según la metodología CRISP-DM. Una vez definidos los elementos de la secuencia, Identificador, Tiempo y Evento, se genera una vista minable y se realiza un estudio de frecuencias en las secuencias de aprobación de materias.

Keywords: Minería de Datos, Minería de Secuencia Temporal, Vista Minable, Metodología CRISP-DM, Frecuencias de secuencias

1. Introducción

La Minería de Secuencias [1] consiste en la búsqueda de patrones secuenciales frecuentes en una base de datos de eventos con fecha y hora [9] [4]. Existen además, diversos métodos para minar secuencias. Ahola [2] explicita varios, tales como SPADE [12], o PrefixSpan [8], entre otros. Como antecedente en la aplicación de minar secuencias sobre datos provenientes de entornos educativos se tiene el trabajo de Guerra [5], donde usan los métodos SPAM [3] y PexSPAM [6] para descubrir patrones comunes en secuencias de ejercicios parametrizados usados en herramientas de aprendizaje on-line.

Los algoritmos de minería de secuencias, requieren hacer una transformación de los datos para llevarlos al formato adecuado. Una *Vista Minable* es la consolidación en una única tabla de todas las observaciones y los atributos sobre los que se aplicarán los algoritmos de minería de datos.

La metodología CRISP-DM (CRoss Industry Standard Process for Data Mining) [11], respecto a otras metodologías, posee ventajas comparativas según Moine [7]. CRISP-DM es una metodología para el descubrimiento de conocimiento

en bases de datos, estructurada en un proceso jerárquico, compuesto por tareas descriptas en cuatro niveles diferentes de abstracción, que van desde lo general a lo específico. CRISP-DM propone, en el nivel más alto, seis fases para el proceso de minería de datos: *1. Entendimiento del negocio*, *2. Entendimiento de los datos*, *3. Selección y Preparación de los datos*, *4. Modelado*, *5. Evaluación* y *6. Implementación*. Cada fase plantea tareas generales que se proyectan a tareas específicas, que describen las acciones que deben ser desarrolladas para situaciones específicas en procesos de extracción de conocimiento a partir de datos.

Las fases 1 y 2 fueron planteadas en el trabajo “Extracción de Conocimiento en el Cursado del Ciclo Común de Articulación de Carreras de Ingeniería” [10] con el objetivo de caracterizar y analizar el recorrido académico de exámenes rendidos por alumnos del Ciclo Común de Articulación (CCA) de las Carreras de Ingeniería en la Facultad de Tecnologías y Ciencias Aplicadas (FTyCA) de la Universidad Nacional de Catamarca (U.N.CA.)

En el presente trabajo se lleva a cabo la fase *3. Selección y Preparación de Datos*, que consta de dos sub-fases: *Comprensión de los Datos* (Apartado 3.1) y *Preparación de los Datos* (Apartado 3.2). En la primera de ellas se realizan las actividades de *Recolección de datos iniciales*, *Descripción de los datos* y *Exploración inicial de los datos*. En la segunda, se ejecutan las actividades de *Seleccionar los datos*, *Limpieza de los datos*, *Construcción de los datos*. Se busca de este modo, a partir de registros de actas de exámenes, construir la vista minable que sea apta para ser procesada mediante técnicas de minería de secuencias.

En la Sección 4 se explica el proceso de transformación de los datos para generar la vista minable. Luego, en la Sección 5, se muestran resultados del análisis frecuencial sobre la vista minable. Finalmente, la Sección 6 cierra el trabajo con las conclusiones y trabajo futuro.

2. Minería de Secuencias Temporales

La Minería de Secuencias Temporales consiste en encontrar patrones de secuencias, generalmente bajo la forma de asociaciones del tipo: *cuando ocurre A, entonces ocurre B dentro de algún lapso de tiempo* [2]. El proceso de descubrir patrones secuenciales involucra dos etapas: representar las secuencias y la aplicación del algoritmo que encontrará patrones frecuentes en las secuencias.

La formulación del problema de minería de secuencias frecuentes involucra los siguientes elementos básicos [12]:

- Alfabeto: conjunto de ítems. $I = \{i_1, i_2; \dots; i_m\}$
- Evento: n-upla no ordenada de ítems. $\alpha_i = (i_1; i_2; \dots; i_n)$
- Secuencia: lista ordenada de eventos $\alpha = (\alpha_1 \rightarrow \alpha_2 \rightarrow \dots \rightarrow \alpha_k)$

Una secuencia α está compuesta de uno o más eventos α_i que a su vez incluyen n ítems del alfabeto I .

El tamaño de un evento está dado por la cantidad n de ítems que lo integran.

El ancho de la secuencia lo determina el máximo tamaño de los eventos α_i que componen dicha secuencia. El largo de una secuencia está determinado por la cantidad k de eventos que componen la misma.

El primer paso para aplicar un método de minería de secuencias es representar los datos a procesar bajo el formato de *transacciones*. Una transacción o secuencia se identifica mediante un número (Id de transacción) y está compuesta por una serie de eventos.

Las transacciones o secuencias temporales incorporan además información sobre el momento de ocurrencia de cada evento (variable tiempo). Por ello en una secuencia temporal, la serie de eventos se ordena según la variable tiempo y se consigna el tiempo de ocurrencia del evento.

El total de transacciones constituye la base de datos D para realizar la minería de secuencias. Por lo tanto D es una colección de secuencias de entrada. Cada secuencia de entrada tiene un identificador único (S_{id}) y, a su vez, cada evento de una secuencia tiene también un identificador único (E_{id}). Además, cada evento de una secuencia tiene asociado un valor de la variable tiempo ($t(E_{id})$), donde no puede haber 2 eventos con el mismo valor de tiempo. Por lo tanto, $t(E_{id})$ se puede usar como identificador del evento. Dentro de una transacción, los eventos se ordenan de manera ascendente según la variable tiempo. Es decir, si el evento α_i ocurre antes que el evento α_j , entonces $t(\alpha_i) < t(\alpha_j)$.

En el presente trabajo, se obtiene el conjunto de datos inicial y se lo procesa para conformar un conjunto D de secuencias de exámenes aprobados que tenga las variables (S_{id}), (E_{id}) y $t(E_{id})$, que requiere una vista minable de minería de secuencia temporal. Luego, como primer estudio de las secuencias, se realizan diversos análisis frecuenciales sobre el conjunto de secuencias D .

3. Selección y Preparación de Datos

3.1. Comprensión de los Datos

Esta fase de comprensión de los datos, comprende la recolección inicial de datos, con el objetivo de establecer un primer contacto con el problema, familiarizándose con ellos, identificar su calidad y establecer las relaciones más evidentes que permitan definir futuras acciones.

3.1.1. Recolección de datos iniciales. Se definió como unidad observacional los exámenes rendidos por alumnos de las carreras de Ingeniería de la Facultad de Tecnología y Ciencias Aplicadas, particularmente los exámenes correspondientes a las materias del CCA, acotando la consulta de información a los exámenes hasta el 02/06/2016 de alumnos de las cohortes entre los años 2004 y 2013 inclusive.

3.1.2. Descripción de los datos La información extraída del sistema SIU-GUARNI fue presentada en un archivo en formato Microsoft Excel con la totalidad de exámenes rendidos por los alumnos de las carreras de ingeniería. La consulta consta de 14166 registros (exámenes), con los siguientes campos:

- Carrera: el nombre de la carrera de Ingeniería que se dicta en la FTyCA.

- **IdAlumno**: el número de legajo que identifica unívocamente a cada alumno.
- **Cohorte**: el año de ingreso a la Carrera del alumno.
- **Materia**: Código alfanumérico que identifica a cada materia del CCA.
- **Fecha**: fecha en la cual el alumno ha rendido el examen de la materia.
- **Resultado**: resultado evaluatorio que ha obtenido el alumno en un examen.
- **Forma**: forma en la que se ha tomado el examen ya sea rindiendo el mismo o solicitando equivalencia.

En la tabla 1 se puede apreciar la descripción de las variables extraídas.

Tabla 1. Descripción de las variables extraídas

Atributo	Tipo	Valores Posibles
Carrera	Catógorica	[MINAS,ELECTRONICA,AGRIMENSURA,INFORMATICA]
IdAlumno	Entero	
Cohorte	Catógorica	2004-2005-2006-2007-2008-2009-2010-2011-2012-2013
Materia	Catógorica	M1-M2-M3-M4-M5-M6-M7-M8-M9-M10-M11
Fecha	Fecha	[01/01/2004 al 31/12/2013]
Resultado	Catógorica	[APROBADO - AUSENTE - REPROBADO]
Forma	Catógorica	[EXAMEN - EQUIVALENCIA]

3.1.3. Exploración inicial de los datos. Los datos recolectados se han explorado con el software WEKA para observar las distribuciones de las variables catógoricas. No se contabilizaron frecuencias de las variables “*Fecha*” y “*IdAlumno*”. La variable “*Fecha*” no se evaluará en esta etapa porque más adelante será transformada. La variable “*IdAlumno*” no se considera porque no aporta información al problema de estudio, por ser sólo un identificador.

De los 14436 registros de exámenes, se tiene que en 5660 de ellos el **Resultado** es *APROBADO*, en 3991 es *REPROBADO* y 4785 con *AUSENTE*.

En la Figura 1 se observa la distribución de las variables catógoricas del conjunto de datos considerando como variable de clase a la variable “*Resultado*”. El bloque inferior representa los exámenes con resultado *Aprobado*, el bloque medio aquellos con resultado *Reprobado* y el bloque superior a los exámenes con resultado *Ausente*.

En la tabla 2 se detalla la cantidad de exámenes que existen por materia.

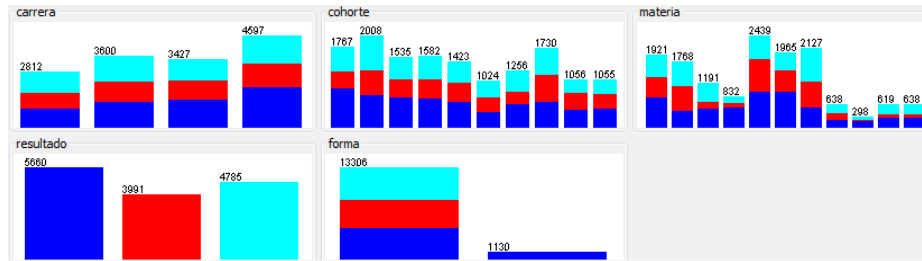


Figura 1. Distribución de las variables categóricas del conjunto de datos con variable de clase “Resultado”

Tabla 2. Cantidades de Exámenes por Materia

Materia	Codigo de Materia	Cantidad
FUNDAMENTOS DE INFORMÁTICA	M1	1921
QUÍMICA	M2	1768
FÍSICA I	M3	1191
SISTEMA DE REPRESENTACIÓN	M4	832
ÁLGEBRA	M5	2439
GEOMETRÍA ANALÍTICA	M6	1965
ANÁLISIS MATEMÁTICO I	M7	2127
ANÁLISIS MATEMÁTICO II	M8	638
CÁLCULO AVANZADO	M9	298
PROBABILIDAD Y ESTADÍSTICA	M10	619
FÍSICA II	M11	638

3.2. Preparación de los datos

En esta fase y una vez efectuada la recolección inicial de datos, se procede a su preparación para adaptarlos a las técnicas de Minería de Datos que se utilicen posteriormente.

3.2.1. Seleccionar los datos. Se descarta para este estudio el atributo “Carrera”, porque se considera que no aporta información en el análisis de secuencia deseado ya que el conjunto de materias del CCA es común en todas las carreras.

3.2.2. Limpieza de los datos. Se eliminaron todas las observaciones de aquellos alumnos que hubieran aprobado al menos una materia bajo la forma de aprobación *EQUIVALENCIA*. Los exámenes rendidos de esta forma no permiten determinar la fecha original de su aprobación. También se eliminaron las observaciones donde el resultado del examen fuera *Reprobado* o *Ausente*, debido a que el objeto de estudio es la secuencia de exámenes aprobados.

3.2.3. Construcción de los datos. Para poder aplicar los algoritmos de minería de secuencia temporal se necesita que la vista minable tenga un atributo que exprese el tiempo del evento. El atributo “*Fecha*” podría ser candidato para marcar el tiempo del evento, pero tiene la dificultad de que estamos considerando en el conjunto de datos a alumnos de diferentes cohortes con los cual sus exámenes no se han rendido en el mismo intervalo de tiempo. Resulta necesario normalizar los valores del atributo “*Fecha*” en una escala común a todos los alumnos, y para ello se decidió normalizar el valor de la variable “*Fecha*” sobre una variable que represente cuántos semestres académicos transcurrieron desde el ingreso a la carrera hasta la fecha del aprobación del examen. Como el año académico va de Abril a Marzo, se considera como un semestre académico al período Abril-Septiembre y el otro semestre comprende los meses Octubre-Marzo. Para determinar en qué semestre aprobó las diferentes materias cada alumno, se inicia en 1 el conteo a partir del año de ingreso a la carrera indicado el valor de la variable “*Cohorte*”. Luego, se incrementa en 1 por cada semestre que avanza en el tiempo de la vida académica del alumno. En la tabla 3 se muestra un ejemplo de la normalización de la fecha. En el caso particular del alumno 1 cuya cohorte es 2006 y fecha de examen 01/12/2006, el valor del semestre se calcula sumando la cantidad de semestres transcurridos desde 01/04/Cohorte hasta el semestre en el que corresponde la fecha de examen, en este caso es [Abril-2006 a Septiembre-2006] + [Octubre-2006 a Marzo-2007] = 2 semestres.

Tabla 3. Ejemplo de normalización de la variable “*Fecha*” en la variable “*Semestre*”

IdAlumno	Cohorte	fecha del examen	Semestre
1	2006	04/07/2006	1
1	2006	01/12/2006	2
2	2004	02/07/2004	1
2	2004	23/11/2005	4
2	2004	20/02/2006	4
3	2005	02/12/2005	2
3	2005	30/11/2006	4

4. Modelado de la Vista Minable

Luego de las etapas de limpieza y construcción de datos se ha obtenido un conjunto de datos a analizar de 4198 registros de exámenes aprobados. En la Tabla 4 se muestra un ejemplo del conjunto de datos así obtenido, donde las filas se ordenan según la fecha de examen.

Tabla 4. Ejemplo de representación de datos de actas de examen ordenadas por fecha de examen.

fecha del examen	semestre	IdAlumno	Materia
02/07/2004	1	2	M4, M5
23/11/2005	4	2	M1, M3
02/12/2005	2	3	M1
20/02/2006	4	2	M6
04/07/2006	1	1	M6
30/11/2006	4	3	M2, M7
01/12/2006	2	1	M3

Para generar la vista minable apta para la aplicación de minería de secuencias temporales, el siguiente paso consiste en agrupar para cada alumno, las actas de exámenes de aquellas materias que haya aprobado. Para ello, se reordenan las filas y columnas de la Tabla 4, quedando conformada la estructura que se muestra en Tabla 5.

Tabla 5. Ejemplo de detalle cronológico de materias aprobadas por cada alumno.

IdAlumno	Fecha del examen	Semestre	Materia
1	04/07/2006	1	M6
1	01/12/2006	2	M3
2	02/07/2004	1	M4, M5
2	23/11/2005	4	M1, M3
2	20/02/2006	4	M6
3	02/12/2005	2	M1
3	30/11/2006	4	M2, M7

Un conjunto de datos secuenciales se caracteriza por contar con tres columnas: *Objeto* (S_{id}), *Tiempo* $t(E_{id})$, *Ítems*. Se considera como variable *Tiempo del Evento* $t(E_{id})$ a la variable *Semestre* de la Tabla 5.

En la Tabla 6 se muestran las secuencias temporales construidas a partir de la Tabla 5, cuyo formato es el apropiado para ser procesado con algoritmos de *Minería de Secuencias*. Dicho formato no permite que una secuencia tenga eventos en un mismo tiempo. Por ello, para el alumno 2 que en la Tabla 5 tiene 2 eventos en el semestre 4, en la Tabla 6 ambos eventos se unifican.

Así la Tabla 6 tiene los elementos básicos descriptos en la Sección 2:

- **Identificador único de secuencia:** S_{id} establecido según el atributo “*Legajo*”.
- **Tiempo del evento:** $t(E_{id})$ es el atributo “*Semestre*” definido en el Apartado 3.2.3.
- **Ítems del evento:** representa la n-upla de ítems, atributo “*Materia*” aprobadas por un alumno en un semestre académico en particular.

Tabla 6. Ejemplo de secuencia temporal de materias aprobadas por cada alumno.

IdAlumno (S_{id})	Tiempo del evento $t(E_{id})$	Ítems del evento
1	1	(M6)
1	2	(M3)
2	1	(M4, M5)
2	4	(M1, M3, M6)
3	2	(M1)
3	3	(M2, M7)

5. Análisis Frecuencial de Secuencias Temporales

Para realizar un estudio de la frecuencias de eventos en la secuencia temporal, se utilizó el software *R*, ejecutando funciones del paquete *arulesSequences* sobre la vista minable construida previamente.

Aplicando la función *summary* a las transacciones generadas mediante la función *read.basket*, se obtiene la cantidad eventos según su tamaño, es decir la cantidad de ítems que componen cada evento. En la Tabla 7 se detalla la cantidad de veces que se aprobaron n materias en un semestre académico. La primera columna informa que 2078 veces se aprobó solo una materia en un semestre, mientras que la última columna expresa que 15 veces se aprobaron 6 materias en un semestre académico.

Tabla 7. Cantidad de eventos según la cantidad de ítems

Cant. Ítems	1	2	3	4	5	6
Cant. Eventos	2078	642	143	48	25	15

La Tabla 8, obtenida aplicando la función *timeFrequency*, muestra la cantidad de eventos ocurridos para cada valor de la variable tiempo. Representa para cada semestre académico en particular (valor de tiempo) cuántas veces se aprobó al menos una materia del CCA, por ejemplo durante el semestre 7 han ocurrido 144 eventos de aprobación de exámenes (cada evento incluye una o mas materias).

Tabla 8. Cantidad de eventos según valor de tiempo $t(E_{id})$

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
402	684	283	410	174	262	144	177	81	99	44	48	32	36	16	23	11	10	7	2	4	1	1

La función *timeFrequency(vista_minable, "gaps")* calcula los valores de la Tabla 9, con la cantidad de veces que eventos consecutivos de las secuencias ocurren con una cierta brecha (*gap*) de tiempo (semestres). La brecha (*gap*) entre 2

eventos consecutivos es la diferencia en tiempo de los mismos [2]. En símbolos: $t(\alpha_j) - t(\alpha_{j-1})$. Por ejemplo, en 89 oportunidades la distancia o brecha entre exámenes aprobados es de 4 semestres.

Tabla 9. Cantidad de eventos cada cierto intervalo de tiempo

Brecha	1	2	3	4	5	6	7	8	9	10	11	13	14	15	16
Eventos	1123	579	168	89	35	29	11	10	10	3	3	1	1	2	1

La función $timeFrequency(vista_minable, "span")$ calcula los valores de la Tabla 10, que muestra la cantidad de secuencias (alumnos) según el lapso ($span$) de tiempo (semestres). El lapso es la duración neta (en semestres) del recorrido académico de aprobación de materias. El lapso ($span$) de una secuencia se determina restando el tiempo del último evento menos el tiempo del primer evento de la misma [2]. En símbolos: $span(S_k) = \max(t(\alpha_j(S_k))) - \min(t(\alpha_i(S_k)))$. Por ejemplo, son 66 las secuencias (alumnos) que entre la primera y la última materia aprobada tienen un lapso de 4 semestres. Con $lapso = 0$ hay 242 secuencias (alumnos), es decir que esas secuencias tienen un único evento.

Tabla 10. Cantidad de secuencias por lapso de tiempo

Lapso	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	22
Secuencias	242	101	68	57	66	57	41	57	38	35	17	19	18	19	14	13	6	8	4	4	1	1

La Tabla 11 muestra la cantidad de secuencias (alumnos) según la cantidad de ítems de cada secuencia (exámenes). Por ejemplo con 11 exámenes aprobados existen 101 alumnos.

Tabla 11. Cantidad de secuencias por cantidad de ítems

Cant. Ítems	1	2	3	4	5	6	7	8	9	10	11
Cant. Secuencias	166	134	129	77	66	49	56	49	40	19	101

6. Conclusiones y Trabajo Futuro

Siguiendo la metodología CRISP-DM se pudo construir una vista minable apta para algoritmos de minería de secuencias, en datos del contexto educativo.

El análisis frecuencial realizado a las secuencias permite conocer diferentes características del fenómeno de aprobación de materias del CCA. Lo más común es que se apruebe una sola materia en un semestre académico (Tabla 7). Los alumnos aprueban mayor cantidad de materias del CCA durante el segundo semestre de su historia académica (Tabla 8). Lo usual es que se aprueben materias

con un semestre de diferencia (Tabla 9). La mayoría de los alumnos registra un sólo evento de aprobación de materias (Tabla 10). De los 886 alumnos considerados, sólo 101 (el 11 %) de ellos han aprobado las 11 materias del CCA (Tabla 11).

El resultado de este trabajo posibilita que a futuro se cumpla con las fases 4 y 5 de CRISP-DM, utilizando algoritmos específicos para encontrar patrones frecuentes de aprobación de materias y relacionarlos con el grado de avance en la carrera y los plazos que demanda aprobar las materias del CCA. De esta manera se espera contar con información estratégica y novedosa, la cual no es brindada por las herramientas habituales de informes y que se encuentra oculta en los datos y solo puede ser obtenida por técnicas de minería de datos.

Referencias

1. Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*, pages 3–14. IEEE, 1995.
2. Jussi Ahola. Mining sequential patterns. 2001.
3. Jay Ayres, Jason Flannick, Johannes Gehrke, and Tomi Yiu. Sequential pattern mining using a bitmap representation. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 429–435. ACM, 2002.
4. Dante Conti and Fco Javier Martínez de Pisón Ascacibar. Reglas de asociación en series temporales: panorama referencial y tendencias.
5. Julio Guerra, Shaghayegh Sahebi, Yu-Ru Lin, and Peter Brusilovsky. The problem solving genome: Analyzing sequential patterns of student work with parameterized exercises. In *Educational Data Mining 2014*, 2014.
6. Joshua Ho, Lior Lukov, and Sanjay Chawla. Sequential pattern mining with constraints on large protein databases. In *Proceedings of the 12th International Conference on Management of Data (COMAD)*, pages 89–100, 2005.
7. Juan Miguel Moine. *Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo*. PhD thesis, Facultad de Informática, 2013.
8. Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Mei-Chun Hsu. Mining sequential patterns by pattern-growth: The prefixspan approach. *Knowledge and Data Engineering, IEEE Transactions on*, 16(11):1424–1440, 2004.
9. Prof Pinkal Shah and AK Dua. Algorithm for sequence mining using gap constraints. *International Journal of Engineering Research and Development*, pages 37–49, 2014.
10. Oscar Eduardo Quinteros, Ana Funes, and Hernán César Ahumada. Extracción de conocimiento en el cursado del ciclo común de articulación de carreras de ingeniería. In *XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina)*, 2016.
11. Rüdiger Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, pages 29–39. Citeseer, 2000.
12. Mohammed J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Mach. Learn.*, 42(1-2):31–60, January 2001.

Clustering de un flujo de datos usando MapReduce

María José Basgall ^{1,2}, Waldo Hasperué ², Cesar Estrebou ², Marcelo Naiouf ²
{mjbasgall, whasperue, cesarest, mnaiouf}@lidi.info.unlp.edu.ar

¹ UNLP, CONICET, III-LIDI, La Plata, Argentina

² Instituto de Investigación en Informática (III-LIDI)
Facultad de Informática - Universidad Nacional de La Plata

Resumen Las técnicas de agrupamiento (*clustering*) sobre flujo de datos (*data stream*) son una poderosa herramienta para determinar las características que tienen en común los datos provenientes del flujo. Para obtener buenos resultados es necesario almacenar gran parte de éste en una ventana temporal. En este artículo medimos una técnica que maneja el tamaño de la ventana temporal de manera dinámica utilizando un algoritmo de clustering implementado en el framework MapReduce. Los resultados obtenidos demuestran que esta técnica alcanza una ventana de gran tamaño logrando así que cada dato del flujo se utilice en más de una iteración del algoritmo de clustering permitiendo conseguir resultados similares independientemente de la velocidad de los datos del flujo. Los centroides resultantes de cada flujo de datos son semejantes a los que se consiguen haciendo un clustering sobre el conjunto de datos completo.

Keywords: Big Data, Stream Processing, MapReduce, Clustering

1. Introducción

El procesamiento de flujos de datos (*stream processing*) es un área muy estudiada en los últimos años. Stream processing permite llevar a cabo tareas mediante el análisis de un flujo continuo y potencialmente infinito de datos [1][2][3][4][5][6][7][8][9].

El objetivo de stream processing es permitir que las tareas analicen los flujos de datos de forma online, brindando respuestas en tiempos muy cercanos al tiempo real. La principal característica de este tipo de procesamiento es que los datos del flujo llegan a una velocidad tal que no es posible almacenarlos en su totalidad y, si se pueden almacenar, el volumen de datos es tan grande que presenta la dificultad de analizarlo en tiempos de respuesta cortos [10].

Para poder realizar análisis a partir de grandes volúmenes de datos resulta sumamente útil hacer uso de tecnologías multiprocesador así como de las técnicas de paralelización de algoritmos que tomen ventaja de las mismas. Actualmente es posible encontrar diversos aportes donde se aplican conceptos de Cómputo de Altas Prestaciones (*High Performance Computing, HPC*) en el área de stream

processing; de igual manera, es posible encontrar trabajos realizados en entornos de computación en la nube (*Cloud Computing*). Estos últimos introducen un nuevo foco desde el punto de vista de la computación de altas prestaciones, ya que brindan un soporte “a medida” para la ejecución de aplicaciones sin la necesidad de adquirir el hardware [11][12][13][14].

Una de las herramientas más utilizadas para el procesamiento de grandes cantidades de datos es Hadoop MapReduce [15][16][17], construida sobre los principios bien conocidos del procesamiento paralelo y distribuido. MapReduce trabaja sobre el sistema de archivos HDFS [18], el cual ofrece eficiencia, es distribuido, brinda almacenamiento tolerante a fallos y es adecuado para aplicaciones que utilizan grandes volúmenes de datos ya que proporciona un alto rendimiento de acceso a los mismos. El framework MapReduce realiza las tareas de distribución y paralelización de manera automática y transparente al programador lo que lo convierte en una poderosa herramienta para implementar soluciones distribuidas.

Una de las tareas más comunes al realizar stream processing es la de clustering. Esta tarea consiste en agrupar los datos provenientes del flujo en diferentes grupos con la finalidad de encontrar características similares en los datos. En los últimos años se han propuesto varios trabajos que realizan clustering sobre un flujo de datos. Estos aportes se pueden dividir básicamente en dos categorías, los que no almacenan datos utilizando únicamente un conjunto de descriptores por grupo (*cluster*) y los que utilizan una pequeña ventana temporal del flujo de tamaño fijo. La principal desventaja de estos algoritmos es que al no poder trabajar con todos los datos, ya sea porque no los almacenan o solo almacenan una pequeña porción del flujo, no pueden garantizar que los grupos de datos obtenidos reflejen la verdadera distribución de los datos. Este problema puede ser disminuido almacenando la mayor cantidad de datos del flujo posible y para ello es necesario contar con algoritmos que sean capaces de ejecutarse en entornos de trabajo distribuidos y paralelos [19].

Recientemente se ha propuesto una técnica que permite el manejo de una ventana temporal dinámica del flujo utilizando el framework MapReduce [20]. Esta técnica permite almacenar una gran cantidad de datos provenientes del flujo administrando el tamaño de la ventana de manera dinámica en función de la frecuencia de llegada de los datos del flujo y del tiempo que le lleva al algoritmo que realiza el análisis de los datos en completar una etapa de su tarea.

En el presente trabajo se midió el rendimiento de la técnica presentada en [20] aplicando una tarea de clustering sobre un flujo de datos proveniente de Twitter. La tarea de clustering utilizada está basado en el algoritmo K-means [21] al cual se le realizaron las modificaciones necesarias para poder ejecutarlo en el framework Hadoop MapReduce. Se eligió el algoritmo K-means por su simplicidad en la implementación pero esta técnica puede ser aplicada a cualquier algoritmo de clustering.

Se midió la respuesta final de los centroides obtenidos con esta técnica utilizando un flujo de datos proveniente de Twitter, empleando diferentes frecuencias de llegada de los datos. Se compararon los resultados obtenidos en cada

experimento con el resultado final producto de ejecutar de manera secuencial el algoritmo K-means convencional sobre el conjunto completo de los datos del flujo recolectado.

El artículo está organizado como sigue. La sección 2 se refiere al clustering sobre un flujo de datos y los diferentes enfoques. En la sección 3 se presenta el método utilizado. En la sección 4 se muestran los experimentos realizados y los resultados obtenidos. Finalmente en la sección 5 se presentan las conclusiones y los trabajos a futuro.

2. Clustering sobre un flujo de datos

El área de minería de flujo de datos es la encargada de procesar flujos de datos con la intención de extraer conocimiento útil, en la mayoría de los casos, para la toma de decisiones. Dentro de la minería de datos el clustering es una tarea muy utilizada para descubrir cómo se relacionan los datos analizados entre sí. Cuando el volumen de datos a analizar es muy grande aparecen nuevas técnicas para realizar clustering sobre flujos de datos.

La tarea de clustering sobre flujos de datos consiste en agrupar los datos disponibles en un número de grupos preestablecido utilizando para ello alguna métrica de similitud. Los algoritmos convencionales de clustering, que trabajan sobre una base de datos, actualizan los correspondientes centroides de los clusters utilizando los vectores n -dimensionales de aquellos datos que están más cerca del propio centroide. Así, para cada dato d , se busca el centroide más cercano y se asigna a éste el dato d . Al finalizar una iteración del algoritmo, cada cluster actualiza su centroide utilizando todos los datos que le fuera asignado. Este proceso continúa un número significativo de iteraciones hasta conseguir un cierto equilibrio entre los centroides. En un escenario donde los datos vienen de manera continua este enfoque no es posible llevar a cabo ya que el volumen de información a almacenar es muy grande y potencialmente infinito. Por ello, se han propuesto diferentes variantes para poder realizar clustering sobre un flujo de datos.

La mayor parte de las técnicas propuestas para el procesamiento de flujos de datos se pueden enmarcar en dos enfoques principales. Uno de ellos son los algoritmos que, ante la imposibilidad de almacenar todos los datos y utilizarlos para calcular el nuevo centroide, hacen uso de descriptores de clusters. Los centroides se actualizan inmediatamente ante la llegada de un nuevo dato o bien ante la acumulación de alguna evidencia de que el centroide debe ser actualizado. Estos algoritmos son modelos que procesan cada dato una única vez [1][22][23][24].

El otro enfoque es utilizar ventanas temporales y deslizantes de tamaño fijo donde almacenan los n últimos datos recibidos, o los n más representativos [25][26] [27] [28][29]. La principal desventaja de estos algoritmos es que el tamaño de ventana se establece a priori y que éste depende fuertemente de la velocidad del flujo de datos. Cuando no se conoce de antemano la frecuencia de llegada de los datos o cuando ésta puede variar a lo largo del tiempo, elegir un tamaño de ventana adecuado representa un desafío.

2.1. Trabajos relacionados

En el trabajo presentado en [30] aplican la idea de microclusters, el objetivo es realizar muchos clusters de tamaño reducido para luego utilizar alguna técnica aglomerativa y determinar cómo quedan formados los clusters a partir de la información recolectada por los microclusters. En este trabajo proponen que cada microcluster solo guarde el vector centroide y un peso que mide la densidad del área o volumen del propio microcluster.

Esa misma idea es utilizada en [31] que además explota el uso del paradigma Hadoop MapReduce para llevar a cabo la tarea. Este trabajo utiliza una segunda fase para realizar un pulido de los microclusters y eliminar aquellos que ya no son relevantes. En [32] se profundiza en esta idea almacenando en cada cluster, el vector centroide, la cantidad de elementos pertenecientes al cluster, una medida de distorsión del cluster y el timestamp del último elemento incorporado en el cluster. Un trabajo de características similares se puede encontrar en [33], que además de almacenar tres descriptores por cluster también presenta un algoritmo de dos etapas, una online y otra offline de pulido de información.

Todos estos trabajos tienen el mismo problema que cada cluster sólo guarda información resumida de todos los elementos pertenecientes en él. Ello representa un gran inconveniente al momento de realizar una actualización de los clusters ya que solo poseen parte de la información y cada cluster está sesgado básicamente por un centro y un radio, sin poder guardar la topología de los datos que pertenecen a un cluster.

Por otro lado, en [34] se utiliza más información de los datos aplicando una ventana deslizante de tamaño fijo mientras que en [35] se emplea support vector machine (SVM) para realizar el clustering tratando el flujo en porciones de tamaño fijo.

Cualquier algoritmo de clustering consigue mejores resultados si cuenta con todos los datos para poder actualizar los centroides. En un entorno de tratamiento de un flujo resulta útil poder contar con un gran número de datos el mayor tiempo posible. En este trabajo se implementó el algoritmo K-means [21] el cual fue adaptado para ser ejecutado en el framework Hadoop MapReduce utilizando la técnica del manejo de ventana temporal presentado en [20].

3. Clustering sobre MapReduce

En [20] se presenta una técnica que gestiona de manera dinámica una ventana de datos recolectados de un flujo. Esta técnica permite que en cada iteración del algoritmo de clustering cuente con la mayor cantidad de datos posible. El tamaño de la ventana temporal es adaptada en función de la frecuencia del flujo, la cantidad de datos almacenados para su procesamiento y el tiempo que le lleva a la tarea de clustering lograr el resultado parcial con el lote de datos actual, manteniendo el mayor tamaño posible del flujo disponible dentro de la ventana. De esta manera, el algoritmo que realiza la tarea de clustering utiliza cada dato durante un número significativo de iteraciones.

La técnica propuesta en [20] consiste en dos tareas. La primera se basa en la captura del flujo de datos y su correspondiente almacenamiento, que se ejecuta de manera continua. El data stream es leído de forma online, y los datos recolectados se van guardando en un buffer el cual es almacenado en un archivo dentro del directorio de trabajo en el HDFS.

La segunda tarea lleva a cabo el procesamiento del algoritmo de *stream mining* (en el caso puntual de este trabajo, una tarea de clustering) la cual realiza su ejecución en el framework MapReduce utilizando como entrada todos los archivos que están dentro del directorio de trabajo en el HDFS. A medida que se colectan datos del flujo, estos son almacenados en un nuevo archivo. Cuando el tamaño total de los mismos supera un umbral, los archivos más antiguos son eliminados, generando así el corrimiento de la ventana de datos.

El funcionamiento de un trabajo MapReduce está basado en dos fases. La primera etapa (Map) consiste en asociar cada dato leído con una clave, es decir, toma un conjunto de datos de entrada y los convierte en otro conjunto de datos donde los elementos son convertidos en tuplas (par de clave/valor). Finalizada esta etapa se ejecuta la segunda fase (Reduce) donde cada proceso reductor (reducer) recibe todos los valores asociados con una misma clave para finalmente realizar la tarea propiamente dicha, la cual es específica del problema a resolver y escrita por el programador.

En este trabajo, en la etapa de MapReduce se realiza el procedimiento de una iteración del algoritmo de clustering el cual está basado en el algoritmo K-means. El número de clusters a encontrar es determinado al comienzo del proceso y los centros iniciales son seleccionados al azar.

La ejecución del trabajo MapReduce comienza cuando el buffer del stream almacena los primeros b datos del flujo. En ese momento se guardan todos los datos acumulados en un archivo dentro de un directorio de trabajo en el HDFS y se lanza su ejecución.

El conjunto de centroides C es pasado al proceso Map y éste busca para cada vector v leído cuál es el centroide cb más cercano. En la salida escribe pares clave-valor donde la clave es cb y como valor el propio vector v .

Para aprovechar al máximo la capacidad del cómputo paralelo se crean K reducers, los cuales cada uno recibe todos los vectores correspondientes a un mismo cluster y con ellos calcula el nuevo centroide del mismo, que resulta de calcular el vector promedio entre todos los vectores recibidos. Cada reducer escribe como salida el nuevo centroide calculado además de todos los vectores que pertenecen al cluster. Esto último permite tener información de cómo están conformados los clusters al finalizar una etapa MapReduce.

Cuando el trabajo MapReduce finaliza, se lee del directorio de trabajo el nuevo conjunto de centroides C . Luego se ejecuta un nuevo trabajo MapReduce para ir refinando el procedimiento de clustering y de esa manera lograr que los centroides converjan. Este proceso iterativo continúa hasta ejecutar un número máximo de iteraciones, o hasta alcanzar un equilibrio donde los centroides no cambian significativamente de una iteración a otra.

Mientras el trabajo MapReduce ejecuta el algoritmo de clustering, el proceso de streaming continúa con la captura de datos llenando un nuevo buffer. Se guarda hasta b datos de este buffer en un archivo nuevo. Esta condición existe para evitar que se escriban archivos con gran cantidad de datos y que la ventana sea reemplazada en su totalidad, logrando así un desplazamiento leve de la misma.

4. Experimentos realizados y resultados obtenidos

Para la realización de los ensayos se utilizó un flujo de datos públicos proveniente de Twitter, el cual posee 778072 tweets en español recolectados durante el desarrollo de la Copa América Centenario el cual se llevó a cabo del 3 al 26 de junio del 2016 en Estados Unidos. La captura de los datos se realizó desde el 30 de mayo al 10 de julio del 2016, utilizando la librería de Java Twitter4J [36] para acceder a los datos mediante la API de Twitter Streaming [37]. Estos tweets han sido procesados para remover URLs, emojis, imágenes y símbolos de puntuación.

Los textos de los tweets fueron procesados para convertirlos en vectores binarios de 95 dimensiones. Debido a que los tweets tienen una longitud muy reducida (140 caracteres como máximo), cada elemento del vector representa la ausencia o presencia de una palabra en particular. En todos los ensayos se utilizó la distancia del coseno (ecuación 1) como medida de similitud para la ejecución del algoritmo K-means ya que es la más utilizada en estos tipos de problemas [38]. De manera arbitraria se eligió un $K=5$ para realizar el clustering.

$$\text{cosine_dist}(x, y) = 1 - \frac{x \cdot y}{|x| \cdot |y|} \quad (1)$$

La figura 1 muestra la frecuencia de llegada de los tweets. Dado que el flujo de datos recolectado duró 42 días, para llevar a cabo los experimentos se transformó el data stream reduciendo la duración entre el primer y el último tweet recolectado, cambiando sólo la duración de todo el flujo, manteniendo la frecuencia relativa de llegada de los tweets. Se realizaron cuatro modificaciones, las que se denominaron como stream50, stream100, stream250 y stream500, la duración final de cada flujo es de aproximadamente 20, 10, 4 y 2 horas, respectivamente. La cantidad de tweets manejados por el algoritmo en cada una de sus iteraciones puede verse en la figura 2.

Las pruebas se realizaron tomando una configuración de los parámetros mencionados en [20] obtenida de manera empírica a partir de la combinación de los mismos que mejor se adapta al flujo de datos analizados, teniendo en cuenta el número de etapas interrumpidas del algoritmo y los valores dado por el coeficiente de renovación (r) que se menciona en dicho trabajo. Esta configuración está dada por un tamaño de buffer inicial de 1000 tweets, un factor que determina el tamaño inicial de la ventana de 50, un factor de extensión del tamaño del buffer de 0.5, un umbral de determinación de convergencia de 0.01 y el número de iteraciones máximas por cada etapa de 4.

Se realizó la comparación de los centroides finales que resultaron de la ejecución del algoritmo de clustering luego de procesar cada uno de los flujos. Se

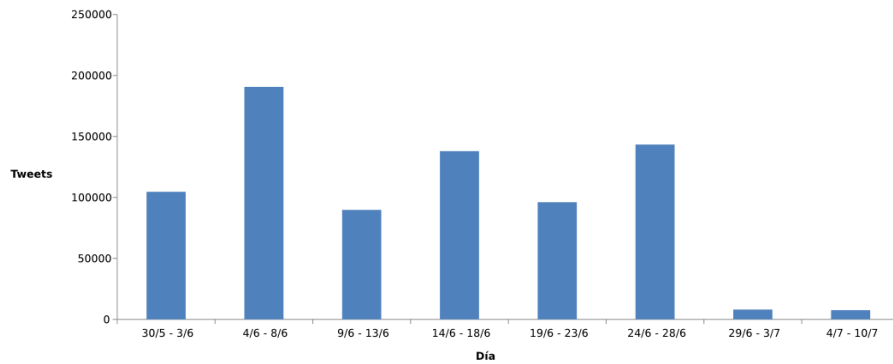


Figura 1: Frecuencia de llegada de los *tweets*

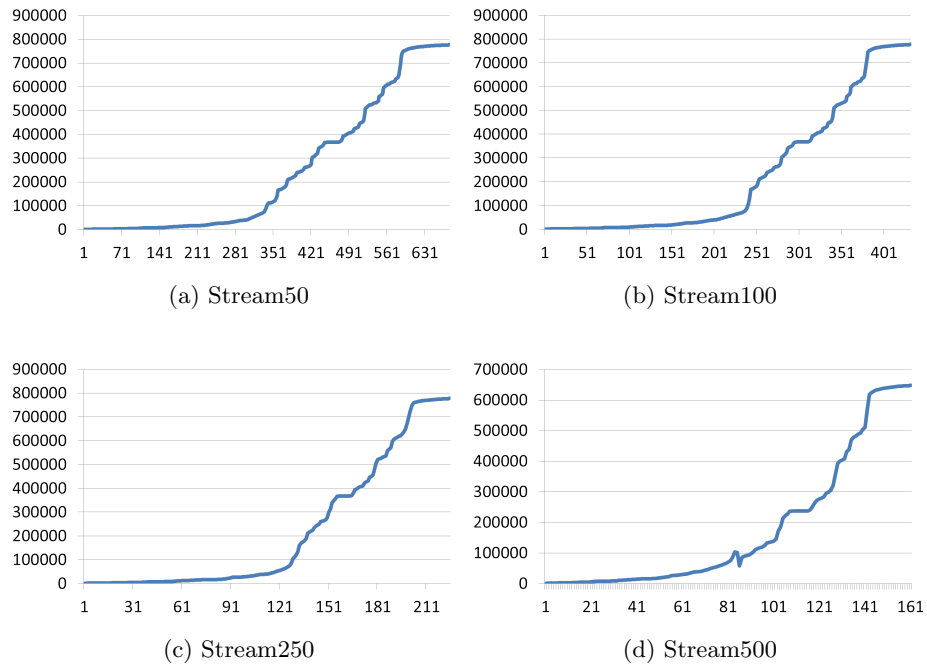


Figura 2: Cantidad de tweets manejados por el algoritmo en cada una de sus iteraciones

cotejaron los centroides finales de cada flujo todos entre sí, y también se hizo la comparación de cada uno de los flujos con el resultado obtenido por el algoritmo K-means secuencial (*Km-sec*) donde se utilizó todo el conjunto de datos en cada una de las iteraciones. La comparación entre los clusters de dos flujos se realizó midiendo la distancia coseno entre el centroide resultado de un flujo y el

centroide del segundo flujo. Esta distancia se midió en cada par de centroides, luego se sumaron todas las distancias para medir el grado de *corrimiento* entre los centroides de los dos flujos. El cuadro 1 muestra el grado de *corrimiento* de los centroides entre todos los flujos y el Km-sec. Se puede observar que la diferencia más grande ocurre entre el flujo stream500 y el Km-sec (en rojo), mientras que la menor diferencia ocurrió entre el flujo stream250 y stream50 (en azul).

	Stream250	Stream100	Stream50	Km-sec
Stream500	6.57E-11	3.28E-11	6.44E-11	6.81E-11
Stream250		3.51E-11	4.20E-12	6.53E-11
Stream100			3.00E-11	5.20E-11
Stream50				6.39E-11

Cuadro 1: Grado de *corrimiento* de los centroides

Al realizar la comparación entre los centroides individualmente se observa que la mayor diferencia ocurre entre el cluster 3 del flujo stream500 y el stream250 con una diferencia de 2.86E-11. La menor diferencia ocurre en el cluster 1 entre el flujo de stream250 y stream50 con una diferencia de -2.22E-16. La tabla con estos valores no está incluida por falta de espacio.

Para la experimentación se utilizó una hoja de un Blade de 8 hojas, con 2 procesadores quad core Intel Xeon e5405 de 2.0 GHz en cada una de ellas. Cada hoja posee 10Gb de RAM (compartido entre ambos procesadores) y cache L2 de 2 x 6Mb entre par de núcleos.

5. Conclusiones

En este trabajo se utilizó la técnica para el manejo dinámico de una ventana temporal de un flujo de datos presentado en [20], donde la característica es maximizar el tamaño de dicha ventana permitiendo que cada dato recolectado del flujo sea utilizado por el algoritmo de clustering la mayor cantidad de veces posibles.

Se midió el manejo del buffer utilizando una tarea de clustering con el algoritmo K-means [21] implementado en el paradigma MapReduce. El algoritmo se evaluó con un flujo de datos donde se aumentó la frecuencia de llegada de los datos en 50, 100, 250 y 500 veces. Los resultados obtenidos permiten observar un manejo eficiente de los datos produciendo resultados muy similares a los que se obtienen con el algoritmo de clustering secuencial utilizando el conjunto de datos completo.

Referencias

- [1] N. Takahashi y col. "A Parallelized Data Stream Processing System Using Dynamic Time Warping Distance". En: *2009 International Conference on Complex, Intelligent and Software Intensive Systems, CISIS 2009, Fukuoka, Japan, March 16-19, 2009*. 2009, págs. 1100-1105.
- [2] Y. Noh, D. Han e Y. Byun. "Real-Time Data Stream Processing for Ubiquitous Home Network Systems". En: *4th International Conference on Multimedia and Ubiquitous Engineering, MUE 2010, Cebu, Philippines, 11-13 August, 2010*.
- [3] C. Kuka. "Processing the uncertainty: Quality-aware data stream processing for dynamic context models". En: *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*. 2012, págs. 560-561.
- [4] D. Bonino y F. Corno. "spChains: A Declarative Framework for Data Stream Processing in Pervasive Applications". En: *Procedia Computer Science* 10 (2012). {ANT} 2012 and MobiWIS 2012. ISSN: 1877-0509.
- [5] J. Stefanowski, A. Cuzzocrea y D. Slezak. "Processing and mining complex data streams". En: *Inf. Sci.* 285 (2014), págs. 63-65.
- [6] R. Agerri y col. "Big data for Natural Language Processing: A streaming approach". En: *Knowledge-Based Systems* 79 (2015). ISSN: 0950-7051.
- [7] Y. Ma y col. "Remote Sensing Big Data Computing". En: *Future Gener. Comput. Syst.* 51.C (oct. de 2015), págs. 47-60. ISSN: 0167-739X.
- [8] P. ZareMoodi, H. Beigy y S. Kamali Siahroudi. "Novel Class Detection in Data Streams Using Local Patterns and Neighborhood Graph". En: *Neurocomput.* 158.C (jun. de 2015), págs. 234-245. ISSN: 0925-2312.
- [9] D. Desai y A. Joshi. "A Deviant Load Shedding System for Data Stream Mining". En: *Procedia Computer Science* 45 (2015). International Conference on Advanced Computing Technologies and Applications (ICACTA). ISSN: 1877-0509.
- [10] A. Rajaraman y J. D. Ullman. *Mining of Massive Datasets*. New York, NY, USA: Cambridge University Press, 2011. ISBN: 1107015359, 9781107015357.
- [11] G. Hager y G. Wellein. *Introduction to High Performance Computing for Scientists and Engineers*. ("Chapman and Hall/CRC Computational Science), CRC Press, 2010. ISBN: 97814398119241.
- [12] P. Pacheco. *An Introduction to Parallel Programming*. 1st. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011. ISBN: 9780123742605.
- [13] S. Zhang y col. "Cloud Computing Research and Development Trend". En: *Future Networks, 2010. ICFN '10. Second International Conference on*. 2010.
- [14] S. S. Saurabh Bilgaiyan y S. S. Sahu. "Cloud Computing: Concept, Terminologies, Issues, Recent Technologies". En: *Research Journal of Applied Sciences* 9 (2014), págs. 614-618.
- [15] Apache Hadoop. <https://hadoop.apache.org/>. Accedido en 07/2016.
- [16] MapReduce. <http://hadoop.apache.org/docs/current/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html>. Accedido en 07/2016.
- [17] J. Dean y S. Ghemawat. "MapReduce: Simplified Data Processing on Large Clusters". En: *Commun. ACM* 51.1 (ene. de 2008). ISSN: 0001-0782.
- [18] Apache Hadoop. <http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html>. Accedido en 07/2016.
- [19] C. C. Aggarwal. "A Survey of Stream Clustering Algorithms". En: *Data Clustering: Algorithms and Applications*. 2013, págs. 231-258.

- [20] M. J. Basgall, W. Hasperué y M. Naiouf. “Tratamiento de un flujo de datos usando ventanas deslizantes con MapReduce”. En: *IV Jornadas de Cloud Computing y Big Data, La Plata, Buenos Aires, Argentina* (2016).
- [21] J. MacQueen y col. “Some methods for classification and analysis of multivariate observations”. En: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, págs. 281-297.
- [22] S.-S. Kim y H.-K. Ahn. “An improved data stream algorithm for clustering”. En: *Computational Geometry* 48.9 (2015). ISSN: 0925-7721.
- [23] E. Lughofer y M. Sayed-Mouchaweh. “Autonomous Data Stream Clustering Implementing Split-and-merge Concepts - Towards a Plug-and-play Approach”. En: *Inf. Sci.* 304.C (mayo de 2015). ISSN: 0020-0255.
- [24] A. Sancho-Asensio y col. “Improving data partition schemes in Smart Grids via clustering data streams”. En: *Expert Systems with Applications* 41.13 (2014), págs. 5832 -5842. ISSN: 0957-4174.
- [25] Y. Li y col. “Incremental Entropy-based Clustering on Categorical Data Streams with Concept Drift”. En: *Know.-Based Syst.* 59 (mar. de 2014).
- [26] Z. Miller y col. “Twitter spammer detection using data stream clustering”. En: *Information Sciences* 260 (2014), págs. 64 -73. ISSN: 0020-0255.
- [27] R. Mythily, A. Banu y S. Raghunathan. “Clustering Models for Data Stream Mining”. En: *Procedia Computer Science* 46 (2015). Proceedings of the International Conference on Information and Communication Technologies, {ICICT} 2014, 3-5 December 2014, Kochi, India. ISSN: 1877-0509.
- [28] PhridviRaj, C. Srinivas y C. GuruRao. “Clustering Text Data Streams – A Tree based Approach with Ternary Function and Ternary Feature Vector”. En: *Procedia Computer Science* 31 (2014). 2nd International Conference on Information Technology and Quantitative Management, {ITQM} 2014.
- [29] M. Z. ur Rehman y col. “Hyper-ellipsoidal clustering technique for evolving data stream”. En: *Knowledge-Based Systems* 70 (2014). ISSN: 0950-7051.
- [30] M. Hahsler y M. Bolaños. “Clustering Data Streams Based on Shared Density between Micro-Clusters”. En: *IEEE Transactions on Knowledge and Data Engineering* 28.6 (2016), págs. 1449-1461. ISSN: 1041-4347.
- [31] W. Hu y col. “Research on Parallel Data Stream Clustering Algorithm Based on Grid and Density”. En: *Computer Science and Mechanical Automation (CSMA), 2015 International Conference on.* 2015, págs. 70-75.
- [32] X. Zhang y col. “Data Stream Clustering with Affinity Propagation”. En: *IEEE Transactions on Knowledge and Data Engineering* 26.7 (2014).
- [33] A. Amini, H. Saboohi y T. Y. Wah. “A Multi Density-Based Clustering Algorithm for Data Stream with Noise”. En: *2013 IEEE 13th International Conference on Data Mining Workshops.* 2013, págs. 1105-1112.
- [34] R. Fathzadeh y V. Mokhtari. “An ensemble learning approach for data stream clustering”. En: *21st Iranian Conference on Electrical Engineering (ICEE).* 2013.
- [35] C. D. Wang y col. “SVStream: A Support Vector-Based Algorithm for Clustering Data Streams”. En: *IEEE Transactions on Knowledge and Data Engineering* 25.6 (2013), págs. 1410-1424. ISSN: 1041-4347.
- [36] Twitter4J. <http://twitter4j.org/en/index.html>. Accedido en 07/2016.
- [37] The Streaming APIs. <https://dev.twitter.com/streaming/overview>. Accedido en 07/2016.
- [38] E. Rasmussen. “Information Retrieval”. En: ed. por W. B. Frakes y R. Baeza-Yates. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992. Cap. Clustering Algorithms, págs. 419-442. ISBN: 0-13-463837-9.

Estimación del Plano Anestésico en Elefante Marinos del Sur Utilizando Técnicas de Machine Learning.

Marcos Zárate^{1,2}, Mirtha Lewis¹

¹ Grupo de Estudio de Macroecología Marina (GEMMA), CENPAT-CONICET, Almirante Brown 2915, Puerto Madryn, Chubut.

² Laboratorio de Investigación en Informática (LINVI), Facultad de Ingeniería, Universidad Nacional de la Patagonia San Juan Bosco, Almirante Brown 3051, Puerto Madryn, Chubut.
{zarate,mirtha}@cenpat-conicet.gob.ar

Abstract. Prediction systems are techniques that build and study new forecasts through a branch of the artificial intelligence called Machine Learning. In this work we intend to estimate the time that remains anesthetized an southern elephant seal to which you have applied a combination of drugs (Zoletil[®]), the fundamental objective of anesthesia is to avoid risky situations to researchers studying this species. To know these times data mining techniques and algorithms used particular classification algorithms were compared *JA.8*, *SMO*, *Random Tree*, *NBTree* y *Naïve Bayes* with data mining tool Weka and a data set containing the records of 96 individuals undergoing anesthesia procedure. It is concluded that after tests Random Tree was the classification algorithm that best responded, making this an accuracy of 98.79%.

Keywords: Machine Learning, WEKA, Anaesthesia, Random Tree, Mirounga Leonina.

1 Introducción

El grupo de trabajo, bajo la denominación GEMM@ (acrónimo de Grupo de Estudio de Macroecología Marina) del CENPAT-CONICET¹ dedica sus esfuerzos al desarrollo de estudios sobre ecología del paisaje de especies marinas, utilizando herramientas informáticas e instrumentos tecnológicos para tal fin.

En este contexto la población de elefantes marinos del sur (Mirounga Leonina, EMS de aquí en adelante) es de gran interés debido a que dentro de las poblaciones de EMS del mundo, la colonia de Península Valdés (ubicada en el mar Argentino, en la provincia del Chubut, Argentina) es la única que presenta un marcado crecimiento [1]. Para la conservación y estudio de la especie, se requiere obtener muestras de sangre, tejido, peso del ejemplar, toma de medidas y realizar la colocación de equipos electrónicos para el posterior rastreo satelital una vez que dejan las costas y migran al mar durante varios meses. En particular, la implementación de dispositivos electrónicos miniaturizados en EMS (Fig. 1) proporciona perfiles oceanográficos de la columna de agua hasta profundidades de 2000 metros completando así vacíos de información o reforzando la existente [2] [3]. Estos instrumentos devuelven, a un

¹ <http://www.cenpat-conicet.gob.ar/>

menor costo, grandes volúmenes de datos hidrográficos en regiones que nunca se han estudiado en forma directa por boyas y/o buques oceanográficos.

Todas estas tareas requieren la inmovilización del individuo, para ello se recurre al uso de drogas anestésicas. El procedimiento consiste en inyectar por vía intramuscular una dosis de Zoletil®² (combinación de un agente anestésico disociativo, clorhidrato de Tiletamina y un tranquilizante del grupo de las benzodiazepinas, clorhidrato de Zolazepam). Esta combinación de drogas se utiliza como agente inductor, sedante y anestésico. Luego que el individuo entra en lo que se conoce como *plano de inducción*, es decir que se encuentra parcialmente anestesiado se procede a la colocación de una inyección intravenosa con la misma combinación de drogas para de esta manera anestesiarse completamente al individuo, esta etapa se conoce como *plano anestésico*. Normalmente la segunda dosis por vía intravenosa se realiza con individuos adultos debido al volumen corporal de los mismos, mientras que en individuos jóvenes basta con una dosis de la droga vía intramuscular para inducir el *plano anestésico*. Es en este momento cuando se pueden comenzar las tareas de recolección de muestras de sangre, tejido, y colocación de instrumentos electrónicos de rastreo, luego de un determinado periodo de tiempo el EMS recupera su estado de conciencia y una vez asimilada completamente la droga puede volver a su estado normal. Conocer el tiempo del *plano anestésico* es fundamental para los investigadores debido a que pueden ocurrir accidentes si el EMS intenta defenderse, considerando su volumen promedio, el daño que pueden ocasionar es un factor de riesgo que se debe tener siempre presente.

Ante la necesidad de establecer los tiempos seguros de trabajo, se hizo necesario acudir a los sistemas computacionales de aprendizaje automatizado con el objetivo de precisar, utilizando algoritmos de clasificación, los factores que más aportaron a la estimación de tiempos seguros de trabajo. El presente trabajo se estructura de la siguiente manera: la Sección 2 presenta la metodología utilizada para abordar el problema, la Sección 3 muestra las pruebas realizadas, la Sección 4 los resultados obtenidos y finalmente la Sección 5 las conclusiones.



Fig. 1. a) Instrumentos de rastreo satelital. b) Colocación de los instrumentos en el EMS, para realizar este procedimiento se requiere anestesiarse al animal. c) Instrumentos de rastreo satelital ya adheridos al cuerpo.

2 Materiales y Métodos

2.1 Investigación de la literatura sobre anestesia

Para comprender los diferentes métodos de inmovilización y las drogas que se utilizan habitualmente con la especie *Mirounga Leonina*, se hizo una recorrida por las

² <http://www.imvab.com.ec/fichasguaffy/zoletil50.html>

publicaciones que son tomadas como referencia en este ámbito. Primero se investigó los diferentes métodos de inmovilización utilizados en especies pinnípedas [4] luego se toma como referencia [5] [6] donde se describen las diferentes combinaciones de drogas y sus efectos en EMS y por último en [7] se muestra el efecto que tiene el Zoletil® en elefantes marinos de diferente tamaño, edad, etc.

2.2 Pre-procesamiento de los datos

Basado en los datos recolectados en los censos de elefantes marinos realizados en la Península Valdés durante los años 1998 a 2007 se cuenta con 96 registros de procedimientos de anestias de diferentes individuos, particularmente estos son del grupo Weaning Pup (cachorro destetado) y para cada ejemplar se consideran los siguientes atributos: Identificador del ejemplar (ID), Sexo, Días transcurridos desde el destete, Dosis en MG de Zoletil®, Efecto de la inducción en minutos, Tiempo de recuperación en minutos, Largo del individuo en cm, Circunferencia en cm y el peso calculado en base al largo y circunferencia [8].

Otros atributos como horas y fechas fueron quitados ya que no son relevantes, además se agregó un nuevo atributo (combinación de otros dos) que representa la relación que existe entre la dosis de Zoletil® suministrada y el peso del individuo.

Dado que los tiempos de recuperación se representan en minutos y para el set de datos que se toma en consideración, este intervalo va desde los 18 a los 65 minutos, se definieron cuatro intervalos de tiempo asignándoles un valor nominal que los representa, los intervalos de tiempo se pueden ver en la Tabla 1.

Tabla 1. Intervalos de tiempo definidos para realizar la clasificación.

Intervalo en minutos	Tiempo de recuperación
(-inf, 26]	RAPIDO
(26,34]	NORMAL
(34,42]	LENTO
(42, +inf)	MUY LENTO

El set de datos utilizado contaba con 14 atributos (Columnas) de los cuales se utilizaron los por cuestiones de simplicidad y para mejorar la estimación solo los que se observan en la Tabla 2.

Tabla 2. Atributos seleccionados luego del pre-procesamiento.

Nombre	Descripción
SEXO	Sexo del ejemplar (M, F) (Nominal).
DOSIS I (Mg.)	Droga inyectada en Mg (Numérico).
PESO (Kgr.)	Peso calculado en base al largo y la circunferencia del ejemplar (Numérico).
RELACION (dosis/peso)	Cociente entre la dosis suministrada y el peso total del ejemplar (Numérico).
TIEMPO REC (Min.)	Tiempo en que la droga hace su efecto (Nominal).

2.3 Metodología propuesta

Para alcanzar el objetivo se proponen los siguientes pasos:

- Pre-procesamiento de los datos de anestias disponibles.
- La aplicación de los diferentes algoritmos para estimar los tiempos del plano anestésico.
- Comparación de los resultados obtenidos y selección del mejor algoritmo de clasificación.

La estimación y la predicción pueden ser vistas como tipos de clasificación. El problema general es evaluar el modelo a través del conjunto de datos de entrenamiento y luego verificar el resultado utilizando un conjunto de datos de prueba. Para este trabajo se utilizó la herramienta WEKA (acrónimo de Waikato Environment for Knowledge Analysis), el mismo nace del esfuerzo de un grupo de investigadores del Machine Learning Laboratory de la Universidad de Waikato en Nueva Zelanda, como software de código abierto bajo los términos de la GNU GPL el cual es una suite completa de herramientas para aprendizaje automático y minería de datos [9] [10]. La Fig. 2 muestra la ventana principal de WEKA con todas las opciones disponibles.



Fig. 2. Ventana principal de WEKA, desde aquí se puede elegir la opción Explorer para trabajar con el set de datos y realizar las pruebas con los algoritmos de clasificación. Desde la opción Experimenter compararemos los algoritmos elegidos para ver en detalle su precisión.

3 Pruebas realizadas

El objetivo de este trabajo es comparar diferentes algoritmos de clasificación utilizando WEKA para predecir tiempos del plano anestésico luego de inyectada la droga. En este caso tenemos un archivo con datos que contienen 96 registros de anestias en formato csv, pero que fueron convertidos al formato arff³ utilizado por WEKA. Los algoritmos elegidos para realizar las pruebas fueron *J4.8 decision tree*, *SMO*, *Random Tree*, *NB tree* y *Naive Bayes*. En todos los casos se utilizó el mismo set de datos en formato arff, para nuestro caso anestias.arff.⁴ Para las pruebas se trabajó con la opción *Explorer* de WEKA, haciendo pruebas con los diferentes algoritmos de clasificación, en todos los casos elegimos la opción *Percentage Split* que por defecto es de 66% (esto quiere decir que usaremos el 66% de los datos para entrenar el

³ <http://www.cs.waikato.ac.nz/ml/weka/arff.html>

⁴ <https://drive.google.com/open?id=0ByFBIFdvMqVlemRqZUFEYzA5TUE>

algoritmo y el resto será para las pruebas), luego se seleccionó el atributo a ser clasificado, en este caso TIEMPO REC (Min). Para realizar la comparativa de los algoritmos escogidos utilizaremos la opción *Experimenter*, donde podremos ver con mayor nivel de detalle cada uno de los algoritmos y poder seleccionar el que mejor desempeño obtenga. Antes de comenzar a mostrar los resultados de cada uno de los algoritmos podemos ver que en la solapa *Preprocess* de la interfaz *Explorer* se visualizan los atributos que contiene el archivo arff, aquí se permite eliminar atributos, aplicar filtros, etc. La Fig. 3 muestra esta solapa, además podemos visualizar la distribución de frecuencia de todos los atributos al mismo tiempo haciendo clic en el botón *Visualize All*.

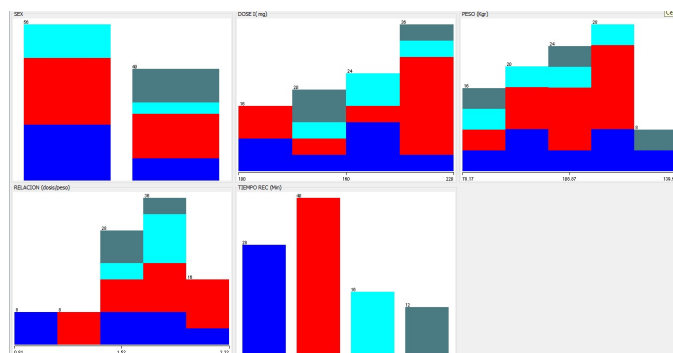


Fig. 3. Distribución de frecuencia de todos los atributos del dataset anestusias.arff, se observa que el atributo TIEMPO REC (Min) (tiempo en minutos en recuperarse totalmente del efecto de la droga) va desde los 18 a los 65 minutos incluyendo 28 ejemplares que se recuperaron en el intervalo RAPIDO, 40 en el intervalo NORMAL, 16 en el intervalo LENTO y 12 ejemplares en el intervalo MUY LENTO.

3.1 J48 Decision Tree

El algoritmo J48 implementado en WEKA es una versión del clásico algoritmo de árboles de decisión C4.5 [11]. Los árboles de decisión entran dentro de los métodos de clasificación supervisada, es decir, se tiene una variable dependiente o clase, y el objetivo del clasificador es determinar el valor de dicha clase para casos nuevos. El algoritmo J48 amplía las funcionalidades del C4.5, tales como permitir la realización del proceso de post-poda del árbol mediante un método basado en la reducción del error o que las divisiones sobre las variables discretas sean siempre binarias.

En la pestaña *Classify* buscamos el algoritmo J48 de entre los arboles de decisión que nos proporciona WEKA y procedemos a la ejecución. Como se puede apreciar una vez terminada la ejecución, se muestra la salida en formato de texto, (ver Fig. 4).

Los resultados obtenidos fueron los siguientes: se puede apreciar que se clasificaron correctamente el 90.9091 % de las instancias e incorrectamente el 9.0909 %, además se puede observar en la sección *Confusion Matrix* que los valores de la diagonal principal corresponden a las instancias correctamente clasificadas, mientras que los valores que están fuera de la misma son las instancias clasificadas incorrectamente, se puede ver que tres instancias fueron clasificadas como MUY LENTO, pero en realidad correspondían a NORMAL, para ver más detalles de los

errores en la clasificación solo deberemos posicionarnos sobre la ejecución realizada en la lista *Result list*, apretar el botón derecho y elegir la opción *Visualize classifiers errors*.

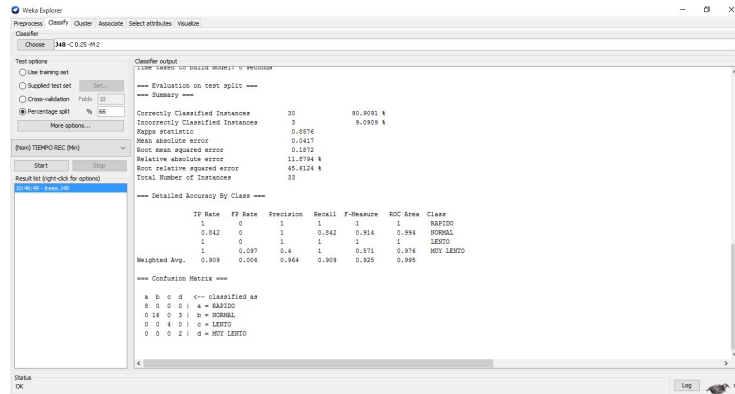


Fig. 4. Resultados obtenidos luego de ejecutar el algoritmo J48, se puede observar que tres instancias fueron clasificadas incorrectamente, las mismas correspondían a NORMAL, pero fueron clasificadas como MUY LENTO.

3.2 SMO (Sequential Minimal Optimization)

SMO implementa el algoritmo de optimización mínima secuencial para entrenar una máquina de soporte vectorial (SVM) [12]. Estos algoritmos extienden las características de los modelos lineales, ya que permiten distinguir entre clases que presentan límites de decisión no lineales. Para ello se transforman los datos originales transformándolos de forma no lineal en un nuevo espacio de mayor dimensión. En este nuevo espacio se construye un modelo lineal que pueda representar un límite de decisión no lineal en el espacio original. Luego de ejecutar el algoritmo los resultados obtenidos fueron los siguientes: se clasificaron correctamente el 42.4242 % de las instancias e incorrectamente el 57.5758 % como se puede apreciar en la Fig. 5.

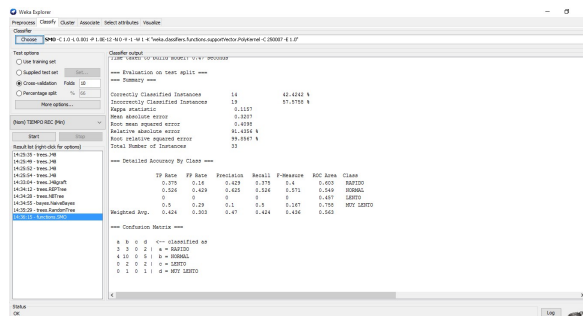


Fig. 5. Resultado obtenidos con el algoritmo SMO, el porcentaje de instancias correctamente clasificadas es mucho menor que con J48.

3.3 Random Tree

Este algoritmo mejora la precisión en la clasificación mediante la incorporación de aleatoriedad en la construcción de cada clasificador individual. Esta aleatorización puede introducirse en la partición del espacio (construcción del árbol), así como en la muestra de entrenamiento. El resultado obtenido luego de la ejecución del algoritmo, se muestra en la Fig. 6, en este caso se clasificaron correctamente el 100% de las instancias, con un 0% de instancias incorrectamente clasificadas.



Fig. 6. Resultado obtenidos con Random Tree, 100% de instancias correctamente clasificadas mejorando al J48 con un 10% más de exactitud.

3.4 Naïve Bayes

Naïve Bayes (NB) [13] es un clasificador probabilístico que parte de la premisa de que cada par parámetro-valor de un mismo ejemplo es independiente del resto. A cada par parámetro-valor se le asigna una probabilidad de pertenencia a una clase. Para ello se divide el número de ejemplos de cada clase en los que aparece ese par entre el número de ejemplos que pertenecen a esa clase. Para clasificar un caso nuevo se calcula la probabilidad de pertenencia de ese caso a cada clase, clasificándolo en la clase donde dicha probabilidad sea mayor, adoptando pues un criterio de estimación máxima a posteriori. El resultado obtenido luego de la ejecución del algoritmo, se muestra en la Fig. 7, en este caso se clasificaron correctamente el 48.4848% de las instancias, con un 51.5152% de instancias incorrectamente clasificadas.

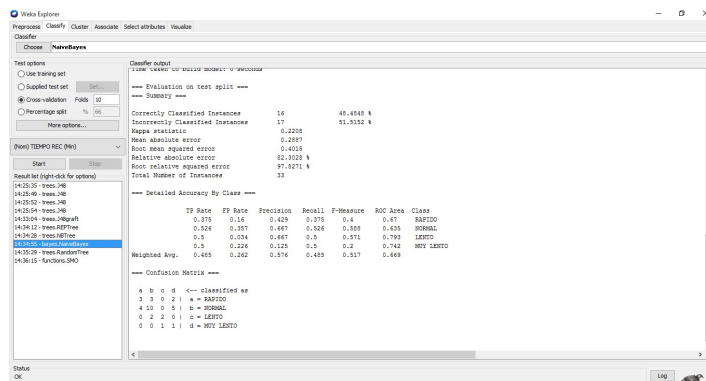


Fig. 7. Resultados obtenidos luego de ejecutar Naïve Bayes.

3.5 NB Tree

Este algoritmo [14] básicamente es un árbol de decisión con clasificadores Naïve Bayes en las hojas. El resultado obtenido luego de la ejecución del algoritmo, se muestra en la Fig.8, en este caso se clasificaron correctamente el 48.4848% de las instancias, con un 51.5152% de instancias incorrectamente clasificadas.

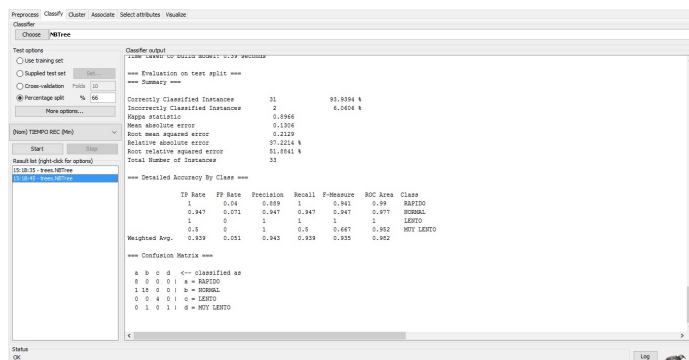


Fig. 8. Resultados obtenidos con el algoritmo NBTree.

4 Resultados obtenidos

Si bien en la sección anterior podemos tener una estimación bastante precisa de que algoritmo presenta mejores resultados, para poder corroborar esto haremos uso de la interfaz *Experimenter* de WEKA la que nos permite conocer de manera detallada el desempeño de cada uno de los algoritmos utilizados, este proceso consiste en seleccionar los 5 algoritmos que se utilizaron aplicados al data set *anestias.arff*, cada algoritmo es corrido 10 veces y se utiliza un 66% del data set para entrenamiento y el 34% restante para las pruebas. Una vez configurado el *Experimenter* se procede a ejecutar las pruebas desde la pestaña *Run*, los resultados de las pruebas se almacenan

en un archivo de extensión arff y se pueden visualizar desde la pestaña *Analyze* abriendo el archivo que se creó en *Setup* como se puede ver en la Fig. 9. El porcentaje correcto para cada uno de los cuatro algoritmos es: 90.82 para *J48*, 98.79 para *Random Tree*, 47.27 para *Naïve Bayes* y 83.84 para *NB Tree*. La notación "v" o "*" indica que un determinado resultado es estadísticamente mejor (v) o peor (*) que el algoritmo base (en este caso, *J48*) en el nivel de significación especificado (actualmente 0,05)

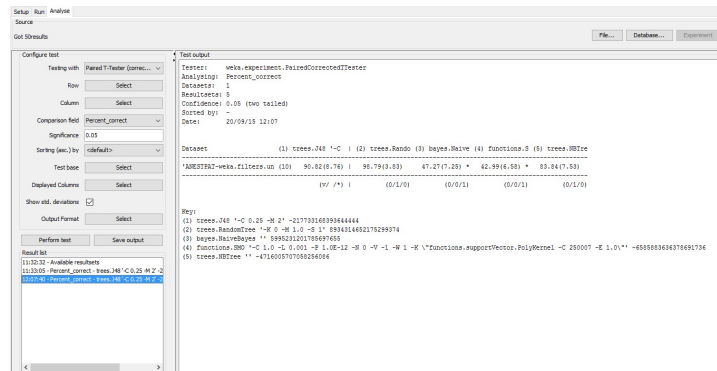


Fig. 9. Resultados obtenidos luego de comparar los 5 algoritmos, en este caso se contrastan 4 de los algoritmos frente al algoritmo *J48*.

Al pie de cada columna se puede ver una leyenda (v/ /*) esto es el número de veces que el algoritmo fue mejor (v) igual (/) o peor (*) al algoritmo base seleccionado (*J48*) teniendo en cuenta el número de data sets que fueron utilizados para el experimento (en este caso solo se usó uno). Los resultados muestran que *Random Tree* y *NB Tree* fueron iguales (0/1/0) que *J48*, mientras que *SMO* y *Naive Bayes* fueron peores (0/0/1) que *J48*. Teniendo en cuenta los resultados previos se puede establecer que el mejor algoritmo de clasificación de los 5 elegidos es el *Random Tree* debido a que su precisión con respecto a *J48* (90.82 %) fue de 98.79%

5 Conclusiones

El presente trabajo es utilizado para estimar los tiempos del plano anestésico luego de suministrada una dosis de Zoletil® por vía intramuscular a ejemplares de elefantes marinos del sur. En particular se utilizaron los datos de los trabajos realizados en las costas de Península Valdés con la categoría *Weaning Pup*. Para lograr este objetivo se propone construir un modelo de predicción utilizando técnicas de clasificación (Algoritmo *Random Tree*) para poder estimar nuevos tiempos de respuesta a la anestesia suministrada en base a una serie de atributos como (sexo, peso, relación dosis/peso, etc.), esto es de gran ayuda para los investigadores que trabajan en la conservación de esta especie, debido al riesgo que existe cuando se trabaja con estos individuos al tomar muestras biológicas. Los resultados obtenidos a través del entrenamiento y la evaluación de los resultados demostraron un buen desempeño en la clasificación de nuevos tiempos de recuperación utilizando el algoritmo *Random Tree* cuya tasa de exactitud alcanza un 98.79%. La Tabla 4 muestra con detalle los resultados obtenidos.

Tabla 4. Resultados obtenidos luego de la ejecución de los diferentes algoritmos.

Parámetro	Random Tree	NB tree	J48	SMO	Naïve Bayes
Tiempo en construir el modelo	0.01 s	0.89 s	0.01 s	0.69 s	0 s
Inst. correctamente clasificadas	100 %	93.93 %	90.90 %	42.42%	48.48%
Inst. incorrectamente clasificadas	0%	6.06 %	9.09 %	57.57%	51.51%
Kappa	1	0.8966	0.8576	0.1157	0.2208
Error Absoluto promedio	0	0.1306	0.0417	0.3207	0.2887
Error Cuadrático medio	0	0.2129	0.1872	0.4098	0.4015
Error Absoluto relativo	0%	37.22 %	11.87 %	91.43 %	82.30 %

Referencias

- Lewis, M. Campagna C. y F. Quintana. 1996. Site fidelity and dispersion of southern elephant seals from patagonia. *Marine Mammal Science* 12(1):138-147.
- Boehme, L., P. Lovell, M. Biuw, F. Roquet, J. Nicholson, S. E. Thorpe, M. P. Meredith, and M. Fedak, 2009: Technical note: Animal-borne CTD-satellite relay data loggers for real-time oceanographic data collection. *Ocean Sci.*, 5, 685–695.
- Roquet, F. et al. Estimates of the southern ocean general circulation improved by animal-borne instruments. *Geophys. Res. Lett.* 40, 6176–6180 (2013).
- Gales NJ. Chemical restraint and anaesthesia of pinnipeds: a review. *Mar Mamm Sci* 1989; 5:228-256.5.
- Woods R, McClean S, Nicol S, Burton H. A comparison of some cyclohexamine based drug combinations for chemical restraint of southern elephant seals (*Mirounga leonina*). *Vet Rec.* 1994; 10:412-429.
- Woods R, McClean S, Nicol S, Burton H. Antagonism of some cyclohex-amine-based drug combinations used for chemical restraint of southern elephant seals (*Mirounga leonina*). *Aust Vet J.* 1995; 72:165-17.
- McMahon, C. R., Burton, H., McLean, S., Slip, D., & Bester, M. (2000). Field immobilization of southern elephant seals with intravenous tiletamine and zolazepam. *Veterinary Record*, 146(9), 251-254.
- Haley, M. P., Deutsch, C. J., & Boeuf, B. J. L. (1991). A method for estimating mass of large pinnipeds. *Marine Mammal Science*, 7(2), 157-164.
- Garner, S. R. (1995, April). Weka: The waikato environment for knowledge analysis. In *Proceedings of the New Zealand computer science research students conference* (pp. 57-64).
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), 10-18.
- Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. Elsevier.
- J. Platt: Fast Training of Support Vector Machines using Sequential Minimal Optimization. In B. Schoelkopf and C. Burges and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, 1998.
- John, G. H., & Langley, P. (1995, August). Estimating continuous distributions in Bayesian classifiers. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence* (pp. 338-345). Morgan Kaufmann Publishers Inc.
- Kohavi, R. (1996, August). Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid. In *KDD* (pp. 202-207).

On the Assessment of Information Quality in Spanish Wikipedia

Guido Urquiza¹, Matías Soria¹, Sebastián Perez Casseignau¹, Edgardo Ferretti^{1,2}, Sergio A. Gómez³, and Marcelo Errecalde^{1,2}

¹ Universidad Nacional de San Luis (UNSL), San Luis - Argentina

² Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (UNSL)

³ Laboratorio de Investigación y Desarrollo en Inteligencia Artificial, Universidad Nacional del Sur (UNS), Bahía Blanca - Argentina

e-mails: {ferretti,merreca}@unsl.edu.ar, sag@cs.uns.edu.ar

Abstract. Featured Articles (FA) are considered to be the best articles that Wikipedia has to offer and in the last years, researchers have found interesting to analyze whether and how they can be distinguished from “ordinary” articles. Likewise, identifying what issues have to be enhanced or fixed in ordinary articles in order to improve their quality is a recent key research trend. Most of the approaches developed in these research trends have been proposed for the English Wikipedia. However, few efforts have been accomplished in Spanish Wikipedia, despite being Spanish, one of the most spoken languages in the world by native speakers. In this respect, we present a first breakdown of Spanish Wikipedia’s quality flaw structure. Besides, we carry out a study to automatically assess information quality in Spanish Wikipedia, where FA identification is evaluated as a binary classification task. The results obtained show that FA identification can be performed with an F1 score of 0.81, using a document model consisting of only twenty six features and AdaBoosted C4.5 decision trees as classification algorithm.

Keywords: Wikipedia, Information Quality, Featured Article Identification, Quality Flaws Prediction

1 Introduction

The online encyclopedia Wikipedia is one of the largest and most popular user-generated knowledge sources on the Web. Considering the size and the dynamic nature of Wikipedia, a comprehensive manual quality assurance of information is infeasible. Information Quality (IQ) is a multi-dimensional concept and combines criteria such as accuracy, reliability and relevance. A widely accepted interpretation of IQ is the “fitness for use in a practical application” [1], i.e. the assessment of IQ requires the consideration of context and use case. Particularly, in Wikipedia the context is well-defined by the encyclopedic genre, that forms the ground for Wikipedia’s IQ ideal, within the so-called *featured article criteria*.⁴ Having a formal definition of what constitutes a high-quality article, i.e. a

⁴ http://en.wikipedia.org/wiki/Wikipedia:Featured_article_criteria

featured article (FA), is a key issue; however, as indicated in [2], in 2012 less than 0.1% of the English Wikipedia articles were labeled as featured. At present, this ratio still remains, since there are 4 785 featured articles out of 5 187 923 articles on the English Wikipedia.⁵

Information quality assessment in Wikipedia has become an ever-growing research line in the last years [3–10]. A variety of approaches to automatically assess quality in Wikipedia has been proposed in the relevant literature. According to our literature review, there are three main research lines related to IQ assessment in Wikipedia, namely: (i) featured articles identification [5, 6, 10]; (ii) quality flaws detection [7–9]; and (iii) development of quality measurement metrics [3, 4]. In this paper we will concentrate on the first two research trends mentioned above.

All the above-mentioned approaches have been proposed for the English Wikipedia, which ranks among the top ten most visited Web sites in the world.⁶ With 1 265 961 articles, Spanish Wikipedia ranks ninth in the list after English, Swedish, Cebuano, German, Dutch, French, Russian and Italian languages. In spite of being one of the thirteen versions containing more than 1 000 000 articles,⁷ and despite being Spanish one of the most spoken languages in the world by native speakers, few efforts have been made to assess IQ on Spanish Wikipedia. To the best of our knowledge, [11] and [12] are the most relevant works related to IQ in Spanish Wikipedia, and [12] can be characterized as belonging to the third main research trend mentioned above.

In [11], Pohn et al. presented the first study to automatically assess information quality in Spanish Wikipedia, where FA identification was evaluated as a binary classification task. The research question which guided their experiments was to verify if successful approaches for the English version, like word count [5] and style writing [6], also work for the Spanish version, and if not, what changes were needed to accomplish a successful identification. Results showed that when the discrimination threshold is properly set, the word count discrimination rule performs well for corpora where average lengths of FA and non-FA are dissimilar. Moreover, it was concluded that character tri-grams vectors are not as effective for the Spanish version as they are for FA discrimination in the English Wikipedia; but Bag-of-Words (BOW) and character n -grams with $n > 3$ performed better in general. This may be because in Spanish many kind of adverbs are fully encompassed in 4-grams or 5-grams. The best F1 scores achieved were 0.8 and 0.81, when SVM is used as classification algorithm, documents are represented with a binary codification, and 4-grams and BOW are used as features, respectively.

The contribution of our work is twofold. On one hand, we report results on FA identification evaluated as a binary classification task, like in [11], but where the document model used is composed of static features rather than dynamic features. On the other hand, this paper also targets the investigation of quality

⁵ https://en.wikipedia.org/wiki/Wikipedia:Featured_articles

⁶ Alexa Internet, Inc., <http://www.alexa.com/siteinfo/wikipedia.org>

⁷ http://meta.wikimedia.org/wiki/List_of_Wikipedias

flaws. We have conducted an exploratory analysis similar to the original one proposed in [13], to reveal both the quality flaws that actually exist and the distribution of flaws in Spanish Wikipedia articles.

With this aim, in Sect. 2, we describe the experimental design and results obtained in the FA identification task. Then, Sect. 3 introduces the problem of predicting quality flaws in Wikipedia based on cleanup tags. Also, our findings are presented and discussed. Finally, Sect. 4 offers the conclusions.

2 Featured Articles Identification

Given the question: *is an article featured or not?* we have followed a binary classification approach where articles are modeled using a vector composed by twenty six features. All article features correspond to *content* and *structure* dimensions, as characterized by Anderka et al. [7]. We decided to implement these features based on the experimental results provided by Dalip et al. [14], which showed that the most important quality indicators are the easiest ones to extract, namely, textual features related to length, structure and style. The dataset used, was the one compiled in [11], which consists of two corpora, namely: a balanced corpus and an unbalanced corpus. It is worth noting that “balanced” means that FA and non-FA articles were selected with almost similar document lengths. In a similar manner, “unbalanced” refers to the fact that non-FA articles were randomly selected without considering their average lengths. Both corpora are balanced in the traditional sense, i.e. the positive (FA) and negative (non-FA) classes contain the same number of documents. In particular, the balanced corpus contains 714 articles in each category and the unbalanced one has 942 articles in each category as well. It is ensured that non-FA articles belonging to the balanced corpus have more than 800 words. The articles belong to the snapshot of the Spanish Wikipedia from 8th, July 2013.

Formally, given a set $A = \{a_1, a_2, \dots, a_n\}$ of n articles, each article is represented by twenty six features $F = \{f_1, f_2, \dots, f_{26}\}$. A vector representation for each article a_i in A is defined as $a_i = (v_1, v_2, \dots, v_{26})$, where v_j is the value of feature f_j . A feature generally describes some quality indicator associated with an article. A few differ slightly from one another, e.g., counts divided by the number of characters instead of words or ratios instead of a pure count. Table 1 shows the features composing our document model; for specific implementations details cf. [15].

Given the characteristics of these features, content-based features were implemented with AWK and shell-script programming using as input the plain texts extracted from the Wikipedia articles. By using the same programming languages, but using as input the wikitexts of Wikipedia articles, structure-based features were calculated. It is worth mentioning that wikitexts are not provided in the corpora of Pohn et al. and they were extracted from the corresponding Wikipedia dump.⁸ To perform the experiments we have used the WEKA Data

⁸ The updated corpora, including the wikitexts, can be downloaded from:
<https://dl.dropboxusercontent.com/u/35037977/Corpus.tar.gz>

Mining Software [16], including its SVM-wrapper for LIBSVM [17]. Notice that all the results discussed below are average values obtained by applying tenfold cross-validation.

2.1 Results

In the first place, we replicated the experimental setting of Pohn et al. [11], where Naive Bayes (NB) and Support Vector Machine (SVM) classification approaches were evaluated, for both corpora. For the unbalanced corpus, the best F1 scores achieved by Pohn et al. were 0.91 and 0.94, for NB and SVM, respectively. In both cases, character 4-grams were used as features (with full vocabulary size) in a binary document model (*bnn* codification from the SMART nomenclature [18]). In particular, the C parameter of SVM was set to 32, after experimentally deriving its value ranging in the set $\{2^{-5}, 2^{-3}, 2^{-1}, \dots, 2^{13}, 2^{15}\}$. For a linear kernel like the one used by Pohn et al., we achieved an F1 score of 0.92, and F1 = 0.94 was achieved by using an RBF kernel with $C = 2^9$ and $\gamma = 2^{-3}$; a configuration similar to the linear kernel given that γ value is close to zero. For the NB classifier, the F1 score obtained was 0.9. Hence, as it can be observed, the performance of both proposals are similar.

For the balanced corpus, a more challenging setting, the F1 scores reported by Pohn et al. for NB classifier were below 0.78 and the best F1 scores achieved were 0.8 and 0.81, for the SVM classifier with full and reduced vocabulary, respectively, using a binary document model. In our experiments, NB performed notably worse than in [11], given that this classifier achieved an F1 = 0.62. For SVM, the best F1 score achieved was 0.78, with an RBF kernel with parameters set to $C = 2^{11}$ and $\gamma = 2^{-3}$, respectively. As usual, these parameters were experimentally derived by a grid-search in the ranges $C \in \{2^{-5}, 2^{-3}, 2^{-1}, \dots, 2^{13}, 2^{15}\}$ and $\gamma \in \{2^{-15}, 2^{-13}, 2^{-11}, \dots, 2^1, 2^3\}$. Different configurations of polynomial kernels were also evaluated (with $d \in \{2, 3, 4, 5\}$ and $r \in \{0, 1\}$) but no better results were obtained than 0.78. It is well known that increasing γ and d parameters from the RBF and polynomial kernels allow for a more flexible decision boundary, but if they are increased too much, this might yield in principle an over-fitting of the model and hence obtaining a poor capability of generalization of the classifier.

Besides, we also evaluated other classification approach — Ada-boosted C4.5 decision trees. This approach has been used before in the context of Wikipedia IQ, but for quality flaws prediction task [19]. Using unpruned trees and one hundred boosting rounds achieved an F1 score of 0.81. This meta-algorithm, was also run with Pohn et al. document models and the performance achieved was F1 = 0.8. As it can be observed, both approaches have quite alike performances, with both classification methods. We believe that the advantage of our feature-engineering approach relies on the fact of having a fixed-size document model, that with only 26 features has a performance comparable to dynamic document models with thousand of features. This is not a minor issue, since having a classifier in a productive environment (like a Wikipedia bot⁹), also implies being able of computing document models efficiently, as in our case.

⁹ <https://en.wikipedia.org/wiki/Wikipedia:Bots>

Table 1. Features which comprise the document model.

Feature	Description
<i>Content-based</i>	
Character count	Number of characters in the plain text, without spaces
Word count	Number of words in the plain text
Sentence count	Number of sentences in the plain text
Word length	Average word length in characters
Sentence length	Average sentence length in words
Paragraph count	Number of paragraphs
Paragraph length	Average paragraph length in sentences
Longest word length	Length in characters of the longest word
Longest sentence length	Number of words in the longest sentence
Shortest sentence length	Number of words in the shortest sentence
Long sentence rate	Percentage of long sentences*
Short sentence rate	Percentage of short sentences*
<i>Structure-based</i>	
Section count	Number of sections
Subsection count	Number of subsections
Heading count	Number of sections, subsections and subsubsections
Section nesting	Average number of subsections per section
Subsection nesting	Average number of subsubsections per subsection
Lead length	Number of words in the lead section [†]
Lead rate	Percentage of words in the lead section
Image count	Number of images
Image rate	Ratio of image count to section count
Link rate	Percentage of links [‡]
Table count	Number of tables
Reference count	Number of all references using the <code><ref>...</ref></code> syntax (including citations and footnotes)
Reference section rate	Ratio of reference count to the accumulated section subsection and subsubsection count
Reference word rate	Ratio of reference count to word count

* A long sentence is defined as containing at least 30 words.

* A short sentence is defined as containing at most 15 words.

† A lead section is defined as the text before the first heading. Without a heading there is no lead section.

‡ Every occurrence of a link (introduced with two open square brackets) in the unfiltered article text is considered when computing the ratio of link count to word count in the plain text.

3 A Preliminary Breakdown of Quality Flaws

Despite the fact that FA identification is a useful task, assessing what kind of shortcomings of an article must be enhanced, would help writers to improve the article's quality. In this respect, cleanup tags are a means to tag flaws in Wikipedia. As shown in Fig. 1, they are used to inform readers and editors of specific problems with articles, sections, or certain text fragments. However, there is no single strategy to spot the entire set of all cleanup tags. Cleanup tags are realized based on templates, which are special Wikipedia pages that can be included into other pages.

Quality flaws prediction in Wikipedia was a research line started in 2011 by Anderka et al. [13] and evolved in seminal works like [2, 20, 21]. Particularly, in [2] an extensive exploratory analysis on Wikipedia's quality flaw structure is presented for the English version, whose approach consisted in creating a local copy of the Wikipedia database. Their results revealed that tagging work in Wikipedia mostly targets the encyclopedic content rather than pages used for content organization and user discussions. Based on this, we decided to use an alternative method, viz. a query retrieving approach on indexed documents with *Elasticsearch*, a search engine which provides scalable and real-time search.¹⁰

We hence introduced an extraction approach that consists of automatically querying the search engine with patterns representing maintenance templates.¹¹ These templates are organized into categories depending on the maintenance



Fig. 1. The Wikipedia article “Salto Base” (Base Jumping) with a cleanup tag indicating that certified references need to be included.

¹⁰ <https://www.elastic.co/>

¹¹ https://es.wikipedia.org/wiki/Wikipedia:Plantillas_de_mantenimiento

task required, but not all maintenance templates necessarily imply a quality flaw. For example, *notification* templates are used to inform Wikipedians to proceed in agreement with the policies and conventions of Wikipedia. Similarly, *protection* templates warn Wikipedians that a particular working space has been blocked for its proper restoration by a librarian due to violations on the policies and conventions of Wikipedia. Likewise, according to our analysis, the remaining categories, namely: *critic maintenance*, *content*, *style*, *fusion* and *development*, do contain templates which can be associated with quality flaws, as shown in Table 2. It is worth noting that, as stated in this table, this preliminary breakdown of quality flaws has been carried out on a recent Wikipedia snapshot, the dump corresponding to April 2016.¹²

The first column of Table 2 specifies the category where templates (second column) associated with a particular flaw type (third column) are organized. The fourth column presents the number of articles that were found containing these particular templates. The flaw type scheme used corresponds to the one proposed by Anderka et al. [2, 13]. Figure 2 shows, from among the tagged articles, how flaw types are distributed. As it can be observed, *verifiability* is by far the most extended flaw type, corresponding to approximately 70% of the tagged content. This finding agrees with the results reported in [2, 13].

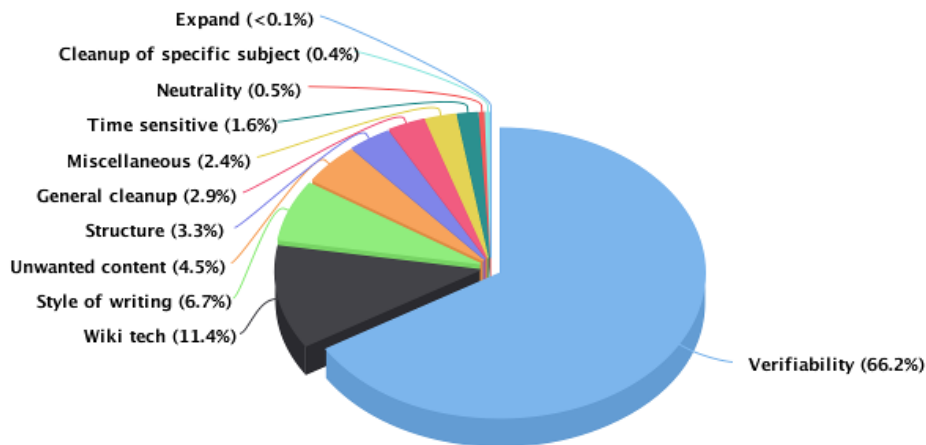


Fig. 2. Flaw types found and their distribution in the Spanish Wikipedia snapshot from April 2016. The percentages relate to the set of 111 072 tagged articles.

Besides, from Table 2, we can notice that the template *Referencias*¹³ represents 90% of the articles tagged with the flaw concerning verifiability. That means that most of the articles suffer from this flaw because they contain neither references nor footnotes. From the remainder 10%, template *Referencias*

¹² <https://dumps.wikimedia.org/eswiki/20160407/>

¹³ <https://es.wikipedia.org/wiki/Plantilla:Referencias>

Table 2. Flaw types breakdown for the Wikipedia snapshot corresponding to April 2016.

Category	Template	Flaw type	# articles
Content	Actualizar	Time sensitive	275
	CDI	Neutrality	4
	Complejo	Style of writing	107
	Desactualizado	Time sensitive	1515
	Discutido	Verifiability	610
	Documentación deficiente	Expand	18
	Ficticio	Cleanup of specific subject	464
	Fuentes no fiables	Verifiability	23
	Globalizar	Neutrality	92
	No neutralidad	Neutrality	341
	Problemas artículo	General cleanup	3218
	PVfan	Neutrality	60
	Referencias	Verifiability	66616
Critic maintenance	Artículo indirecto/esbozo	Wiki tech	7
	Bulo	Verifiability	701
	Contextualizar	Style of writing	162
	Fuente primaria	Verifiability	54
	Infraesbozo	Wiki tech	93
	Plagio	Unwanted content	27
	Posible copyvio	Verifiability	11
	Promocional	Unwanted content	141
	Sin relevancia	Verifiability	320
Development	Traducción	Miscellaneous	2612
Fusion	Fusión historiales	Unwanted content	19
	Fusionar	Unwanted content	2002
	Fusionar desde	Unwanted content	470
	Fusionar en	Unwanted content	740
	Posible fusionar	Unwanted content	73
Style	Categorizar	Wiki tech	99
	Copyedit	Style of writing	3641
	Excesivamente detallado	General cleanup	18
	Formato de cita	Wiki tech	546
	Huérfano	Wiki tech	223
	Identificador	Verifiability	1344
	Largo	Structure	3701
	Mal traducido	Style of writing	3082
	Mejorar redacción	Style of writing	113
	Publicidad	Unwanted content	1528
	Recentismo	Neutrality	5
	Referencias adicionales	Verifiability	3850
	Revisar traducción	Style of writing	342
	Traducción incompleta	Miscellaneous	8
Wikificar	Wiki tech	11731	
Total over all types			111072

adicionales comprise almost 5% and template *Identificador* represents almost 2%. This means that existing references are not enough or are difficult to be found since particular key features are missing in the references, like the ISBN in a book. From Fig. 2, we can also see that *Wiki tech* flaw type, ranks second with 11.4%. In [2, 13], this flaw type also ranked second with approximately 19% and 16%, respectively. In a similar manner as occur with verifiability flaw and the *Referencias* template, in this case, 92% of the articles tagged with the *Wiki tech* flaw type, correspond to template *Wikificar*; indicating that these articles notoriously do not comply to Wikipedia’s style manual. The remaining flaw types and their orderings, differ in [2, 13], as well as in our case; nonetheless, flaw types *Unwanted content*, *Style of writing* and *General cleanup*, are those having in general higher percentages after *Verifiability* and *Wiki tech*.

4 Conclusions

In this work, we have presented a first breakdown of Wikipedia’s quality flaw structure for the Spanish language, following the pioneering approach of Anderka et al. [2, 13]. As reported in these works, *verifiability* related flaws comprise approximately 70% of tagged articles, like found in our study. Without doubts, this preliminary report paves the way for the development and evaluation of existing approaches to predict quality flaws by means of machine learning techniques, like in [8, 9, 19].

Besides, we carried out a study to automatically assess information quality, where FA identification was evaluated as a binary classification task. The results obtained showed that FA identification can be performed with an F1 score of 0.81, using a document model consisting of only twenty six features and AdaBoosted C4.5 decision trees as classification algorithm. These results were compared to previous results reported by Pohn et al. [11], who used dynamic document models with thousand of features, and both approaches have quite alike performances. In our view, the advantage of our feature-engineering approach relies on the fact of having a fixed-size document model which can be efficiently computed in a productive environment, like a Wikipedia bot.

Acknowledgments

This work has been partially founded by PROICO 30312, Universidad Nacional de San Luis, Argentina. Sergio A. Gómez is supported by Secretaría General de Ciencia y Técnica, Universidad Nacional del Sur, Argentina. The authors also thank to PROMINF (*Sub-proyecto “Desarrollo conjunto de sistema inteligente para la Web, con alumnos y docentes de las Licenciaturas en Cs. de la Computación de la UNS y la UNSL”*), Plan Plurianual 2013-2016, SPU.

References

1. Wang, R., Strong, D.: Beyond accuracy: what data quality means to data consumers. *Journal of management information systems* **12**(4) (1996) 5–33

2. Anderka, M., Stein, B.: A breakdown of quality flaws in Wikipedia. In: 2nd joint WICOW/AIRWeb workshop on Web quality (WebQuality'12), ACM (2012) 11–18
3. Lih, A.: Wikipedia as participatory journalism: reliable sources? Metrics for evaluating collaborative media as a news resource. In: Proceedings of the 5th international symposium on online journalism. (2004) 16–17
4. Stvilia, B., Twidale, M., Smith, L., Gasser, L.: Assessing information quality of a community-based encyclopedia. In: 10th Intl. Conf. on Information Quality. (2005)
5. Blumenstock, J.: Size matters: word count as a measure of quality on Wikipedia. In: 17th international conference on World Wide Web, ACM (2008) 1095–1096
6. Lipka, N., Stein, B.: Identifying featured articles in Wikipedia: writing style matters. In: 19th international conference on World Wide Web, ACM (2010) 1147–1148
7. Anderka, M., Stein, B., Lipka, N.: Predicting Quality Flaws in User-generated Content: The Case of Wikipedia. In: 35rd annual international ACM SIGIR conference on research and development in information retrieval, ACM (2012)
8. Ferretti, E., Fusilier, D.H., Guzmán-Cabrera, R., y Gómez, M.M., Errecalde, M., Rosso, P.: On the use of PU learning for quality flaw prediction in wikipedia. In: CLEF (Online Working Notes/Labs/Workshop). (2012)
9. Ferretti, E., Errecalde, M., Anderka, M., Stein, B.: On the use of reliable-negatives selection strategies in the pu learning approach for quality flaws prediction in wikipedia. In: 11th Intl. Workshop on Text-based Information Retrieval. (2014)
10. Lex, E., Völske, M., Errecalde, M., Ferretti, E., Cagnina, L., Horn, C., Stein, B., Granitzer, M.: Measuring the quality of web content using factual information. In: 2nd joint WICOW/AIRWeb workshop on Web quality (WebQuality), ACM (2012)
11. Pohn, L., Ferretti, E., Errecalde, M.: Identifying featured articles in Spanish Wikipedia. In: Computer Science & Technology Series: XX Argentine Congress of Computer Science - selected papers. EDULP (2015) 171–182
12. Druck, G., Miklau, G., McCallum, A.: Learning to predict the quality of contributions to wikipedia. WikiAI 8 (2008) 7–12
13. Anderka, M., Stein, B., Lipka, N.: Towards Automatic Quality Assurance in Wikipedia. In: 20th intl. conference on World Wide Web, ACM (2011) 5–6
14. Dalip, D.H., Gonçalves, M.A., Cristo, M., Calado, P.: Automatic assessment of document quality in web collaborative digital libraries. Journal of Data and Information Quality 2(3) (December 2011) 1–30
15. Fricke, C.: Featured article identification in wikipedia. Bachelor Thesis, Bauhaus-Universität Weimar (2012)
16. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: An update. SIGKDD Explorations 11(1) (2009)
17. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27
18. Salton, G.: The Smart Retrieval System: Experiments in Automatic Document Processing. Prentice Hall (1971)
19. Ferschke, O., Gurevych, I., Rittberger, M.: FlawFinder: a modular system for predicting quality flaws in Wikipedia. In: Notebook papers of CLEF 2012 labs and workshops. (2012)
20. Anderka, M., Stein, B., Busse, M.: On the Evolution of Quality Flaws and the Effectiveness of Cleanup Tags in the English Wikipedia. In: Wikipedia Academy 2012, Wikipedia (July 2012)
21. Anderka, M.: Analyzing and Predicting Quality Flaws in User-generated Content: The Case of Wikipedia. PhD thesis, Bauhaus-Universität Weimar (June 2013)

Data Matching and Deduplication Over Big Data Using Hadoop Framework

Pablo Adrián Albanese, Juan M. Ale
palbanese@fi.uba.ar ale@acm.org
Facultad de Ingeniería, UBA

Abstract. Entity Resolution is the process of matching records from more than one database that refer to the same entity. In case of a single database the process is called deduplication. This article proposes a method to solve entity resolution and deduplication problem using MapReduce over Hadoop framework. The proposed method includes data preprocessing, comparison and classification tasks indexing by standard blocking method. Our method can operate with one, two or more datasets and works with semi structured or structured data.

Keywords: entity resolution, data matching, hadoop, mapreduce, standard blocking, indexing

1 Introduction

The Entity Resolution problem, also known as Data Matching or Record Linkage refers to identify, relate and merge records corresponding to the same entity stored in different databases. A special case arises when we analyze duplicates of the same database, this problem is known as duplicate detection. The Entity Resolution process involves the following tasks: 1. Perform data preprocessing, 2. Index records by one or more key fields, 3. Compare records 4. Establish a classification between them (i.e. coincidence, coincidence or not possible match), and 5. Deliver results.

Comparing records is a task that grows exponentially because, for instance, if we have 2 databases of 1,000 records each, the number of comparisons we have to do is $1.000^2 = 1,000,000$. In recent decades, the community has developed indexing algorithms order to reduce the number of comparisons.

In the last five years there have been several studies of MapReduce techniques to compare large databases, especially in data matching. These solutions focus in making a better use of parallelism by balancing the records and comparing them using different computational nodes. This problem is called data skew and occurs when the databases contain a non-uniform data distribution which hinders the distribution tasks in different computational nodes used for MapReduce.

In [1] a method is proposed to distribute the load among different computing nodes in order to solve the entity resolution problem with MapReduce using standard blocking.

There are numerous articles where similar methods are proposed. However, most of these articles focus on optimizing the blocking algorithm leaving important questions related to the data matching problem unanswered, such as: How should the preprocessing be included inside MapReduce? Is it possible to use this technique to analyze duplicates of the same database or more than two database? How to perform indexing if we have more than one key field?

This paper extends the proposed method [1] in order to include preprocessing tasks indexed by more than one key field, and record comparison using one, two or more data sets.

The structure of the paper is organized as follows. Related work with an outline of original method is given in Section 2. In Section 3 we present our proposal: a technique to include the preprocessing task and for indexing by more than one key field. We explain where to include matching and classification tasks. Experimental results are presented in Section 4. Conclusions and ideas for future extensions are discussed in Section 5.

2 Related works

In [2] a method for entity resolution on Hadoop that works with semi structured data, including preprocessing tasks, indexing, comparisons and ranking results. It explains how to use different classification techniques and its results to improve future comparisons. The authors in [3] created a method called multi-sig-er, which supports structured and unstructured data type and the tasks contemplate preprocessing of data and reducing comparisons. Another algorithm derived from standard blocking is proposed in [4], however no practical results are provided. Different techniques preprocessing of large amounts of data in MapReduce have been discussed and testing in [5].

The method to extend in this presentation, was presented in [1]. Authors proposed a simple, effective solution to efficiently distribute the calculation of similarity between two entities in entity resolution over MapReduce using two Jobs which the output of the first Job is the input of the second Job. For each Job we must define the Map, Partitioner and Reducer processes. Job 1 is responsible for preparing the information in order for Job 2 can evenly distribute comparisons avoiding the problem of data skew, therefore obtaining a method that is not sensitive to information distribution, so its performance depends only upon the size of the data sets to compare.

3 Our proposal

Many databases contain information with poor quality, incomplete or erroneous information. This problem results in high organizational cost. There are different parameters to measure data quality, the most important are: accuracy, completeness, consistency, timeliness and believability. Accuracy and consistency are the most important in Data Matching. In addition, with Data Matching we face the problem

that data from different databases usually have different formats. When it comes to large amounts of records, we must consider that preprocessing entire databases, consume considerable time and resources. In [5] an experimented of 1 TB data set, preprocessed with MapReduce with 8 nodes, took approximately 63 hours.

For this reason we need a unique method where the data preprocessing be performed in a selectively and efficiently way, because it is not necessary to perform preprocessing to all fields of all records as there are many records that may not need to be compared. The technique proposed consists of reusing the same two Jobs to perform preprocessing task of one field only when needed.

3.1 Distributed Preprocessing Between Job 1 and Job 2

In the proposed technique the task Map of Job 1 processes only key fields, while for the rest of the fields we proceed to apply the acceptance rules. Followed by Job 2 Reducer task where non-key fields are pre-processed in two steps: Step 1. Comparing only against involved fields. Step 2. Comparing against the rest of the fields in case we need to generate that specific record.

We obtain an efficient method that only performs preprocessing tasks as required at each stage of the process.

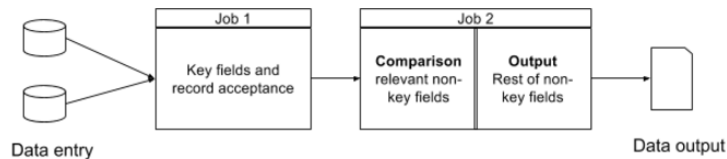


Fig. 1. On map task Job 1 key fields are preprocessed and the necessary fields to define the acceptance criteria of records. In the task reduces the Job 2, the required fields for comparison and classification of records are preprocessed. Finally, only if it is necessary to issue the result, the rest of the fields involved are preprocessed.

3.2 Indexing

The standard blocking method works with a single key field. If we need to search for duplicates by comparing only the records that match on a single field then issues the register in the Map of Job 1. Additionally if we want to find records that match in several key fields, we must generate a common key between them and repeat the same process.

```
function map(key, value)
  fileName = get file name from job context
  fields = split value
  keyField = get key file from fields
  idRegister = get id field from fields
  write(keyField, fileName + idRegister + fields)
```

The traditional standard blocking method does not contemplate comparing records that match any of the key fields but not necessarily all.

3.3 Indexing by More than One Key Field

We have adapted the task performed by the Map function of Job 1 such that it is able to use more than one key field for comparison of records that may match in any of them:

```
function map(key, value)
  fileName = get file name from job context
  fields = split value
  idRegister = get id field from fields
  foreach keyField in fields
    write(keyField, fileName + idRegister + fields)
```

In this new scenario, N pairs (key, value) for each record will be generated, where N is the number of key fields chosen. This modification may generate redundant comparisons of records for cases where two records have similarities in more than one key field. This effect is counterproductive since the performance of the algorithm would begin to depend on the nature of the data and not only the volume.

One way to partially avoid this problem is to use the method Combiner MapReduce. The combiner method runs on the output of the map phase and is used as a filtering or an aggregating step to lessen the number of intermediate keys that are being passed to the reducer. Using this option prevents that Map of Job 2 write duplicate pairs (key, value) that are then being processed by Reducer of Job 2 causing losses in the performance of the process.

```
function combiner(key, values, context)
  value = select first of values
  write(key, value)
```

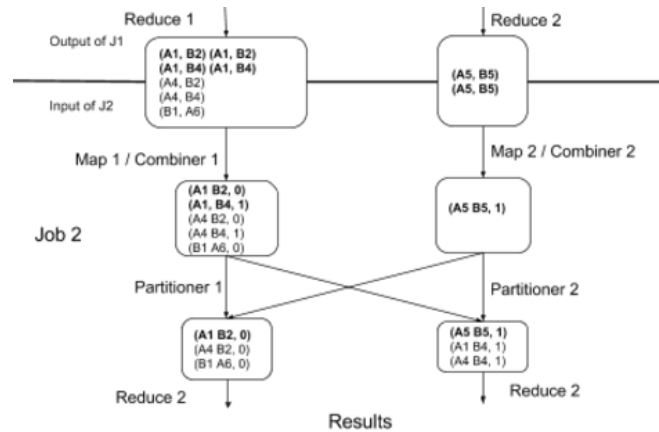


Fig. 2. Pairs of records A1, B2, A1, B4 and A5, B5 have been issued by the Map of Job 1 twice, because they match in two key fields. The Combiner method of Job 2, before being transferred to the reducer of Job 2, groups them into a single entry, preventing comparisons of same records more than once.

As we can see the Map process along with the combiner in Job 2 did not send duplicates to Reduce process values. However this may not always happen, being the combiner just a way to reduce the impact of the records that match in more than one key field.

3.4 Including Matching and Classification

The task of comparing and classifying records, run in the Reducer function of Job 2. As shown in Figure 3, where two records to compare are received, they must obtain the fields in each record, preprocess non-key fields as seen in section 4.2, compare the records, determine the level of similarity between the two records, and if that level is relevant then proceeds to issue the registration.

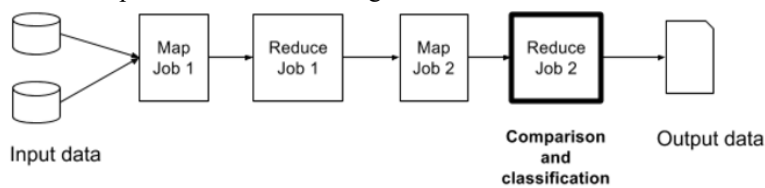


Fig. 3. Comparison and classification are performed in reducer of Job 2.

3.5 Final Method

In this section we summarize the full method when it has 2 data set. Then we explain how to modify this approach for 1 dataset and finally how to apply it when we have more than 1 data set.

1. Data Entry: The allocation of input data set.

2. Preprocessing + exchange key and id fields (Map of Job 1): The preprocessing is performed only in key fields (section 3.1) and for each key field we write pairs key field and id register (section 3.3).
3. Standard partitioner (Partitioner of Job 1): partitioner standard process that distributes tasks in the Reducers available is used.
4. Generation of comparable pairs (Reduce of Job 1): For each key, we write pairs of registers with different file source (entity resolution) or different id (deduplication), pseudocode in section 3.6.
5. Assignment of reducer number (Map of Job 2): Reducer numbers assigned according to the algorithm introduced in [1], the pair (Register 1 Register 2, number of Reducer) is write. A pseudocode is provided:


```
function map(key, value, context)
  reducer = 0
  reduceAssigned = false
  reducers = get number of reducers from context
  register1=key
  register2=value
  while (!reduceAssigned)
  do
    randomNumber = get random number(0, reducers)
    if randomNumber not in selectedReducers
      reducer = randomNumber
      add reducer to selectedReducers
  end
  write (register1 + register2, reducer)
  if quantity of selectedReducers = reducers
    selectedReducers = empty
```
6. Combiner (Combiner of Job 2): Verifies that pairs of matching records in more than one key field are not generated more than once (section 3.3).
7. Mapping reducer with reducer number (Reducer of Job 2): sends pairs of records to the reducer indicated by the number of reducer.
8. Comparison + Classification: we proceed to make the necessary preprocessing of non-key fields for the comparison of records and its subsequent classification (section 3.4).

3.6 Application for One Set of Data (deduplication)

To adapt the proposed method to a single database we only need to modify the process 4:

```
function reduce(key, values) // values = registers
  processedIds = new list
  foreach i values
    foreach j values
      id_i = get id of values[i]
      id_j = get id of values[j]
```

```

    if (id_i <> id_j and id_j not in processedIds)
        write(register_i, register_j)
    Add id_i to processedIds
End foreach
End foreach

```

3.7 Application for more than Two Databases.

In order to find entities that exist in more than 2 databases, the extended method should be performed N-1 times, where N is the number of data set to be analyzed.

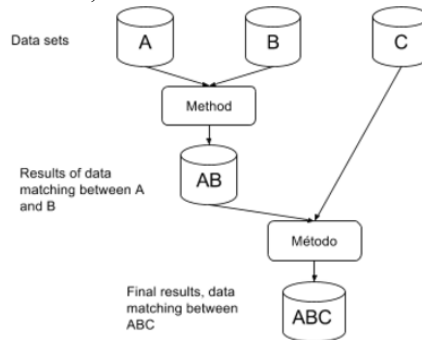


Fig. 3. A, B and C are data sets within which we want to find common entities. First, we run out method between A and B, and then between AB and C .

4 Evaluation

In this section we evaluate the method proposed in the previous section. We use a data set of 250,000 records which keep information about people in a SQL Server database. We generate synthetic records to reach a dataset of 1 million records. Then we divide it into 2 different files.

Table 1. Partial view data set used.

id	first name	last name	email_personal	phone_mobile01	country
101	juan carlos	vaccaro	jvaccaro327@gmail.com		Argentina
102	pablo	salerno	pablos@hotmail.com	+56-1-415449	Chile
103	daniel	ruiz	druiz@yahoo.com	+54-11-12487459	Argentina
105	carolina	Collado	caro_collado@gmail.com	+54-11-25415448	Argentina
110	Magdalena	Gauna	mgauna1979@outlook.com	+54-11-12416446	Argentina

Similar procedures in Transact-SQL language were performed in order to assess that the results are correct. The data set is then exported to two plain text files, to finally upload to an Azure Blob Container which will use our instance of Hadoop for

obtaining and write results. We use Apache Hadoop ¹ 2.7.0 with different amounts of nodes (1, 2, 4 and 8) where each node has 4 processors, 7 GB of RAM and 8 hard drives. The provider was Microsoft Azure² and Microsoft Blob Service Storage was used to upload the input files.

We use a data set of 500,000 records as the key fields *email_personal* and *phone_personal*. For preprocessing we performed the following task:

1. Disposal of records containing invalid data in either of the two key fields and in the field *first_name*. (map of job 1)
2. Turn *email_personal* into lowercase (map of job 1)
3. Normalize *phone_personal* to format [country] - [area] - [number] - [internal], if that is not possible, we continue with the formatting of the original field. (map of job 1)
4. Turn *first_name* into lowercase. (reduce of job 2)
5. Encode *first_name* with DobleMethaPhone algorithm. (reduce of job 2)

For record comparison, we consider as a match those records that contain the same *email_personal* or *phone_personal* and in both cases the same *first_name*.

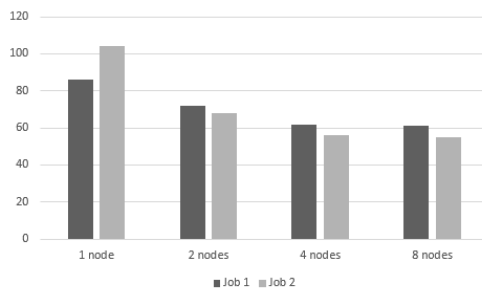
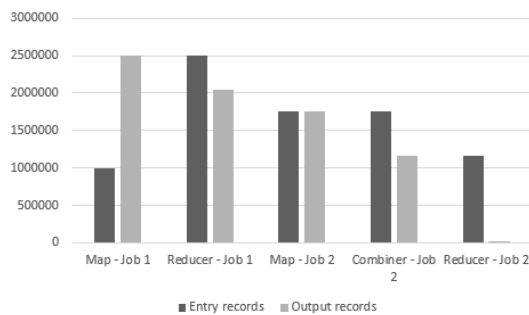


Fig. 4. Execution time according to number of nodes used.

As we increase the number of nodes, runtime decreases, but there is no noticeable difference between the configuration of 4-node and 8-node. This is because it reached its stabilization in 4 nodes.



¹ <http://hadoop.apache.org/>

² <https://azure.microsoft.com>

Fig. 5. Shows entry and output records processed for each phase. With 1 million input registers finally a little over 1.2 million compare. We can also see how the Combiner avoids comparing 0.5 million records that match the email_personal and phone_personal field.

In figure 5 we can see how the number of records that are transferred between one phase and the other increase in large quantities in relation to the total of records that were entered to compare. Since 1.5 million records (job entry map 1) are entered and finished making 116 million comparisons (entry job reducer 2).

5 Conclusions and Further Research

In this presentation a method for solving entity resolution and deduplication problem using Hadoop Framework is proposed. The proposed method includes preprocessing tasks, indexed by more than one key field, comparison and classification results. We introduce alternatives to work with one and more than 2 data set. The main strengths of our method are: its ease for implementation, its ability to include different preprocessing and classification techniques according to each particular problem, and the possibility of using the method of standard blocking, together with the Combiner of Hadoop, with the use of more than one key field avoiding large number of multiple comparisons of the same records. The proposed solution is scalable and can be used both for structured data as well as for semi-structured data.

Future research could incorporate the use of non-relational and distributed databases such as HBase³ with the aim of re use the results of comparisons in future comparisons within the same process, as the authors in [2].

References

1. D. Karapiperis and V.S. Verykios. LoadBalancing the Distance Computations in Record Linkage, ACM SIGKDD Explorations Newsletter, Volume 17 Issue 1 (2015)
2. S. Prabhakar Benny, S. Vasavi, P. Anupriya: Hadoop Framework For Entity Resolution Within High Velocity Streams in CMS 2016, Volume 85, 2016, Pages 550–557 (2016)
3. C. Yan, Y. Song, J. Wang and W. Guo: Eliminating the Redundancy in MapReduce-Based Entity Resolution Cluster, Cloud and Grid Computing (CCGrid)15th IEEE/ACM International Symposium on, Shenzhen, pp. 1233-1236. (2015)
4. Aye Chan Mon, Mie Mie, Su Thwin: Effective Blocking for Combining Multiple Entity Resolution Systems. International Journal of Computer Science Engineering, Vol. 2, No.4, pp 126-136 (2013)
5. Qing He, Qing Tan, Xudong Ma, Zhongzhi Shi: The High-Activity Parallel Implementation of Data Preprocessing Based on MapReduce in Lecture Notes in Computer Science, Volume 6401, pp 646-654 (2015)

³ <https://hbase.apache.org/>

Proceso de Descubrimiento de Reglas de Caracterización de Grupos Espacialmente Referenciados

Giovanni Daián Rottoli^{1,2,3}, Hernán Merlino³, Ramón García-Martínez³

¹ Programa de Doctorado en Ciencias Informáticas. Facultad de Informática.
Universidad Nacional de La Plata. Argentina.

² Programa de Becas “Formación de Doctores para Fortalecer áreas de I+D+i”.
Universidad Tecnológica Nacional. Argentina.

³ Grupo de Investigación en Sistemas de Información. Departamento de Desarrollo Productivo
y Tecnológico. Universidad Nacional de Lanús. Argentina.
gd.rottoli@gmail.com, hmerlino@gmail.com, rgm1960@yahoo.com

Resumen. El descubrimiento de grupos sobre información espacialmente referenciada es un tema de interés en minería de datos espaciales. Los algoritmos existentes se orientan a descubrir tres tipos de grupos: regiones, grupos propiamente dichos, y zonas calientes. Sin embargo, no dan una caracterización de dichos grupos. El presente trabajo propone un proceso de descubrimiento de reglas de caracterización de grupos espacialmente referenciados que utiliza algoritmos TDIDT para obtener las características que fueron elegidas automáticamente para la generación de los mismos. Se presentan pruebas de concepto en las que se utilizan distintos algoritmos de agrupamiento espacial.

Palabras Clave. Reglas de Caracterización de Grupos, Agrupamiento, Información Espacial, TDIDT, Explotación de Información.

1 Introducción

El descubrimiento de grupos, o *clustering* por su término en inglés, permite identificar particiones de la masa de información sobre un dominio de problema determinado [Britos, 2008]. La incorporación de información espacialmente referenciada a esta tarea agrega dificultad a la misma [Kataria & Rupal, 2012], haciendo que nuevos algoritmos y métodos nuevos hayan surgido en los últimos años. Las distintas formas de considerar esta información espacial y las restricciones tenidas en cuenta para la generación de grupos permiten diferenciar tres tipos de abordajes diferentes: regionalización, descubrimiento de zonas calientes o *hotspots* y descubrimiento de grupos espaciales propiamente dicho, concepto que utilizaremos para referirnos a los tres enfoques en forma conjunta [Mennis & Guo, 2009].

En principio, la generación de grupos espaciales es el caso más general, en el cual se buscan conjuntos disjuntos utilizando atributos espaciales, no espaciales, o ambos [Liu et al., 2012]. Por otro lado, la regionalización agrega al descubrimiento de grupos una condición de contigüidad entre los objetos espaciales, permitiendo encontrar conjuntos de objetos contiguos cuyos valores de sus atributos permitan

optimizar cierta función objetivo, tratándose usualmente de una función de homogeneidad o heterogeneidad [Guo, 2008]. Por último, la utilización de los algoritmos de agrupamiento espacial se aplica al descubrimiento de zonas calientes o hotspots, zonas del espacio en las cuales existen concentraciones de puntos espaciales inusuales, haciéndose uso de algoritmos de clustering basados en densidades [Brimicombe, 2007; Nisa et al., 2014; Santoso & Nisa, 2016].

La creación y mejora de algoritmos para estos objetivos ha sido, como se mencionó anteriormente, una tarea constante en los últimos años, pudiéndose encontrar ejemplos de estos en [Brimicombe, 2007; Guo, 2008; Yang & Cui, 2008; Mennis & Guo, 2009; Zhong et al., 2010; Deng et al., 2011; Shah et al., 2012, Popat & Emmanuel, 2014]. Sin embargo, ninguno de estas propuestas brinda en sí misma la posibilidad de caracterizar los grupos generados automáticamente, de forma tal de poder conocer cuáles fueron los criterios utilizados por los algoritmos para realizar esta actividad.

En el presente trabajo, y con base en los hechos presentados hasta el momento, se presenta en la sección 2 la problemática derivada, en la sección 3 una solución propuesta a la misma, pruebas de concepto de dicha solución en la sección 4, y por último se presentan conclusiones en la sección 5.

2 Definición del Problema

Muchos son los algoritmos y métodos desarrollados para el descubrimiento de grupos espaciales en cualquiera de sus formas: grupos, regiones o zonas calientes, tanto para puntos en el espacio como para otros tipos de datos espaciales. Sin embargo, como se mencionó en el apartado anterior, los algoritmos y métodos desarrollados no permiten caracterizar los grupos espaciales descubiertos, en función de los atributos elegidos para tal actividad.

Por similitud con [Britos, 2008], resulta interesante la creación de un proceso para la caracterización de grupos espacialmente referenciados que permita el descubrimiento de grupos y su descripción posterior mediante reglas, de manera sistemática, independientemente del abordaje que se seleccione para la generación de los mismos. Por esta razón, se plantea un proceso de explotación de información para la obtención de reglas de caracterización de grupos espacialmente referenciados utilizando algoritmos TDIDT.

3 Solución Propuesta

[Britos, 2008] propone un proceso de explotación de información para el descubrimiento de reglas de pertenencia a grupos, en el cual utiliza algoritmos TDIDT sobre el resultado de un proceso de descubrimiento de grupos para hallar las reglas que caracterizan a cada uno de los grupos generados. Bajo el mismo concepto, se propone un proceso de explotación de información denominado Proceso de Descubrimiento de Reglas de Caracterización de Grupos Espacialmente Referenciados, que permite determinar las características de los objetos agrupados en un mismo grupo espacial, independientemente de si son de regiones, grupos

espaciales o zonas calientes. Para tal fin, tal como se puede observar en la Figura 1, a partir de un conjunto de información espacialmente referenciada disponible en distintas fuentes y formatos, se realiza un proceso de integración para obtener otro conjunto de información espacial integrada que conste de atributos espaciales, tales como la ubicación espacial del objeto, y atributos no espaciales, en un único registro.

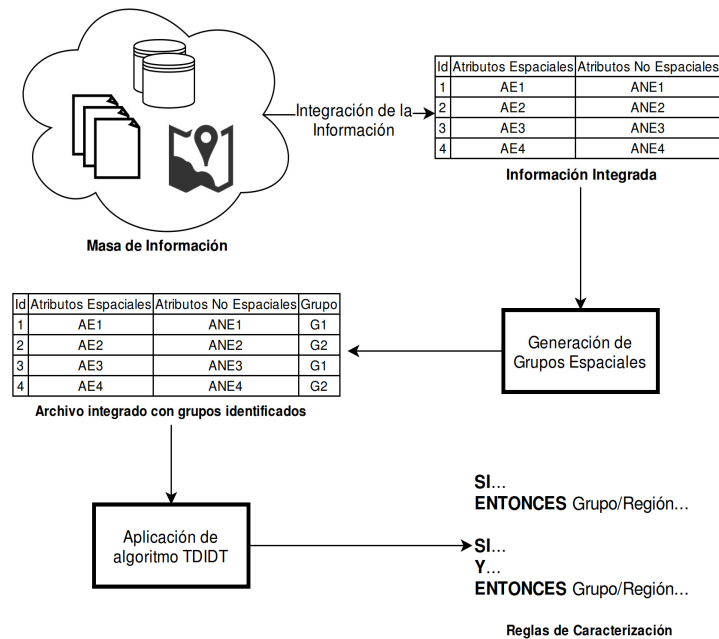


Fig. 1. Proceso de descubrimiento de reglas de caracterización de grupos espacialmente referenciados, donde la sigla AE significa Atributo Espacial y la sigla ANE Atributo No-Espacial

La información integrada es sometida posteriormente a un proceso de descubrimiento de grupos, como se puede observar en la Figura 2. En este paso es necesario optar por una de las tres clases de grupos espaciales mencionadas anteriormente en función al dominio del problema que se esté abordando y seleccionar los algoritmos adecuados en cada oportunidad: para la generación de regiones se sugiere la utilización de algoritmos de la familia REDCAP, debido a sus ventajas sobre otros algoritmos [Guo, 2008]. En este caso, es necesario, además, identificar la relación de contigüidad entre los objetos espaciales. Por otro lado, tanto en el caso del descubrimiento de grupos espaciales como de zonas calientes, se sugiere la utilización de algoritmos basados en densidad tales como aquellos de la familia DBSCAN [Ester et al., 1996; Sander et al., 1998; Nisa et al., 2014], DENCLUE [Hinneburg & Keim, 1998], ASCDT [Deng et al., 2011] o DBSC [Liu et al., 2012], por los mismos motivos mencionados anteriormente [Shah et al., 2012; Popat & Emmanuel, 2014]. En cada caso, del algoritmo seleccionado dependen los atributos de entrada que se le proporcionará. Como resultado de esta etapa se obtiene

un archivo en el cual consta la información integrada más un nuevo atributo en el cual se especifica el grupo espacial al que pertenece.

En un paso posterior, el archivo generado es utilizado como entrada de un algoritmo *Top-Down Induction of Decision Trees* – TDIDT – para la generación de las reglas de caracterización de grupos espacialmente referenciados, especificando el atributo agregado en el paso anterior, “Grupo”, como atributo objetivo o *target*, y los atributos no espaciales como atributos de entrada, tal como se puede observar en la Figura 3.

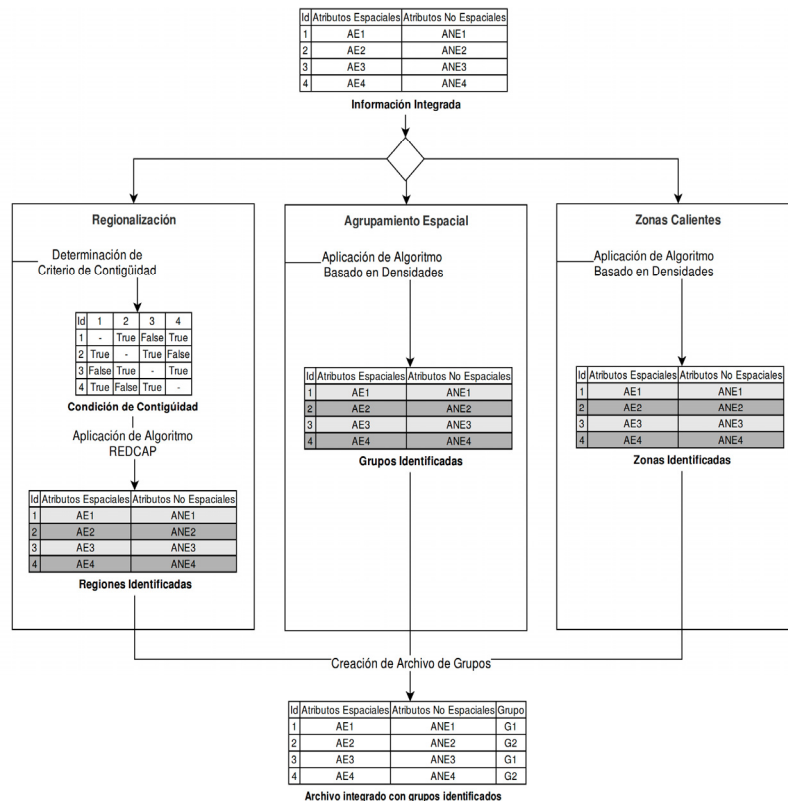


Fig. 2. Subproceso para agrupación de datos espaciales, donde la sigla AE significa Atributo Espacial y la sigla ANE significa Atributo No-Espacial

4 Pruebas de Concepto

En esta sección se presentan dos pruebas de concepto del proceso propuesto utilizando datos reales obtenidos de distintas fuentes. En la sección 4.1 se utiliza algoritmos de regionalización sobre datos referidos a las provincias de Argentina, mientras que en la sección 4.2 se hace uso de algoritmos de descubrimiento de grupos espaciales sobre datos recogidos diariamente por distintas estaciones meteorológicas de Argentina.

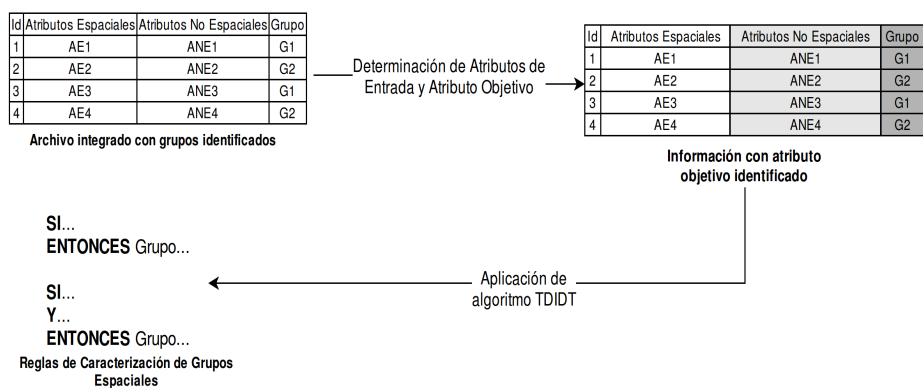


Fig. 3. Subproceso para obtención de reglas de caracterización, mediante la aplicación de Algoritmos TDIDT

4.1 Regionalización de provincias Argentinas

Se realiza la primera prueba de concepto utilizando datos obtenidos en el Censo Nacional de Población, Hogares y Viviendas realizado por la Institución Nacional de Estadísticas y Censos de Argentina en el año 2010 [INDEC, 2010a, 2010b, 2010c, 2010d, 2010e].

Los datos en cuestión han sido integrados y normalizados en un conjunto de atributos, tal como se muestra en la Tabla 1, siendo estos de tipo numérico a excepción del atributo Provincia, el cual corresponde a un valor alfanumérico.

Tabla 1. Descripción de atributos de los datos de prueba sobre las provincias argentinas

Atributo	Descripción
Provincia	Nombre de la provincia
TotViv	Total de viviendas en cada provincia
Superf	Superficie de la provincia, en Km ²
Pob	Cantidad de Habitantes
Dens	Cantidad de personas por kilómetro cuadrado
Analf	Cantidad de personas analfabetas por provincia
EdadM	Edad mediana del total de la población
Porc +65	Porcentaje de la población mayor a 65 años respecto al total de la población
PobOrig	Cantidad de personas descendientes de pueblos originarios con viviendas particulares

Los datos integrados son utilizados como entrada de un algoritmo de regionalización REDCAP. Para este experimento se ha seleccionado el algoritmo First Order SLK [Guo, 2008] debido a la simplicidad de su implementación, usando además, como criterio de contigüidad, la vecindad real de las provincias argentinas, generándose seis regiones espaciales tal como se puede observar en la Figura 5.

Posteriormente, se integran las regiones identificadas en un único archivo de datos en el que constan tanto los datos de entrada originales como la región a la cual pertenece cada uno de ellos. Dicho archivo se utiliza como entrada del algoritmo TDIDT Random Tree [Breiman, 2001] utilizando Tanagra [Rakotomalala, 2005] para su ejecución, obteniéndose un árbol de decisión que se deriva en las reglas que se pueden observar en la Figura 4, de la cual se puede extraer las características de cada grupo destacándose que la confianza del obtenida es del 100%.

SI PobOrig < 49703,5	SI PobOrig < 49703,5
Y Analf >= 14444,5	Y Analf >= 14444,5
Y Superf >= 26162,5	Y Superf < 26162,5
Y Superf < 139895,5	ENTONCES Región 3
Y Dens < 10,9227	SI PobOrig >= 49703,5
ENTONCES Región 1	Y TotViv >= 775948,5
SI PobOrig >= 49703,5	ENTONCES Región 4
Y TotViv < 775948,5	SI PobOrig < 49703,5
ENTONCES Región 1	Y Superf < 139895,5
SI PobOrig < 49703,5	Y Analf < 14444,5
Y Analf >= 14444,5	Y Porc +65 < 6,86%
Y Superf >= 26162,5	ENTONCES Región 5
Y Superf < 139895,5	SI PobOrig < 49703,5
Y Dens >=10,9227	Y Superf >= 139895,5
ENTONCES Región 2	Y EdadM >= 28,95
SI PobOrig < 49703,5	ENTONCES Región 5
Y Superf < 139895,5	SI PobOrig < 49703,5
Y Analf < 14444,5	Y Superf >= 139895,5
Y Porc +65 >= 6,86%	Y EdadM < 28,95
ENTONCES Región 3	ENTONCES Región 6

Fig. 4. Reglas de Caracterización de grupos espacialmente referenciados de la primera prueba de concepto

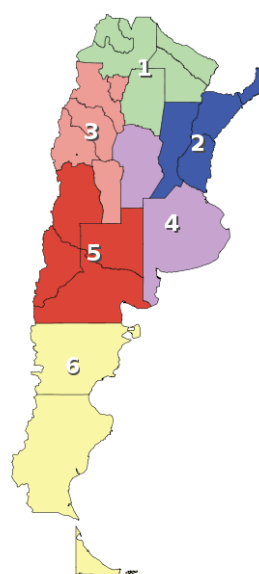


Fig. 5. Mapa de Argentina con regiones generadas

Resulta interesante analizar las características de las regiones descubiertas. En primera instancia se puede observar que el principal atributo para la decisión es la cantidad de habitantes de pueblos originarios: aquellas regiones con mayor cantidad son la región 1, correspondiente al noroeste argentino, y la región 4, correspondiente a Buenos Aires, Córdoba y la Ciudad Autónoma de Buenos Aires, teniendo esta última región mayor cantidad de viviendas que la primera. Por otro lado, aquellas regiones con menor cantidad de habitantes originarios se caracterizan por su superficie, cantidad de personas analfabetas, el porcentaje de personas mayores a 65 años, y la densidad poblacional.

La región 5 y la región 6 se caracterizan por poseer una mayor extensión territorial, y se distinguen entre sí según la edad mediana de la población, teniendo la primera mayor edad mediana que la segunda, siendo el límite aproximadamente 29 años de edad. Cabe destacar que, según las reglas encontradas, existe una región con superficie menor que las restantes pertenecientes al mismo grupo, similar en características a la región 3, la cual posee en su mayoría baja tasa de analfabetismo.

Aquellas regiones con superficie pequeña y mayor tasa de analfabetismo se diferencian a su vez según su densidad poblacional. Aquellas con una densidad menor a 11 personas por Km² corresponden a algunas provincias de grupo 1, y si por el contrario, la densidad es mayor a 11 personas por Km², a algunas provincias del grupo 2.

4.2 Agrupamiento de estaciones meteorológicas argentinas

Se utilizaron datos recolectados por estaciones meteorológicas de Argentina el día 6 de junio de 2016 [INTA, 2016]. Se utilizaron los atributos que se pueden observar en la Tabla 2, utilizando para el agrupamiento basado en densidades utilizando DBSCAN [Ester et al., 1996] mediante el software WEKA [Hall et al., 2009], indicando como parámetros *eps* y *minpts* del algoritmo los valores 0,075 y 4 respectivamente, resultando 5 grupos espaciales, con la distribución que se observa en la Figura 7. Para este paso, los valores fueron normalizados.

Posteriormente, dichos grupos espaciales son integrados al archivo de datos y utilizados como entrada del algoritmo C4.5 [Quinlan, 1993] mediante el software Tanagra, utilizando como atributo objetivo el grupo descubierto y como atributos de entrada las temperaturas mínima, máxima y media, con valores sin normalizar, resultando un árbol de decisión que se deriva en las reglas que se pueden observar en la Figura 6.

Tabla 2. Descripción de atributos de los datos de centrales meteorológicas argentinas considerados en la prueba de concepto

Atributo	Descripción
Lat	Latitud de las coordenadas de la central meteorológica.
Long	Longitud de las coordenadas de la central meteorológica.
TMin	Temperatura Mínima medida en el día.
TMax	Temperatura Máxima medida en el día.
TMed	Temperatura Media del día. (Dato calculado)

SI TMed < 9,85	SI TMed >= 9,85	SI TMed >= 9,85
Y TMin > -0,65	Y TMin >= 7,2	Y TMin < 7,2
Y TMax < 13,5	ENTONCES Grupo 2	ENTONCES Grupo 4
ENTONCES Grupo 0	SI TMed < 9,85	
SI TMed < 9,85	Y TMin > -0,65	
Y TMin < -0,65	Y TMax >= 13,5	
ENTONCES Grupo 1	ENTONCES Grupo 3	

Fig. 6. Reglas de caracterización de grupos espacialmente referenciados obtenidas como resultado de la aplicación de TDIDT sobre los datos de centrales meteorológicas argentinas

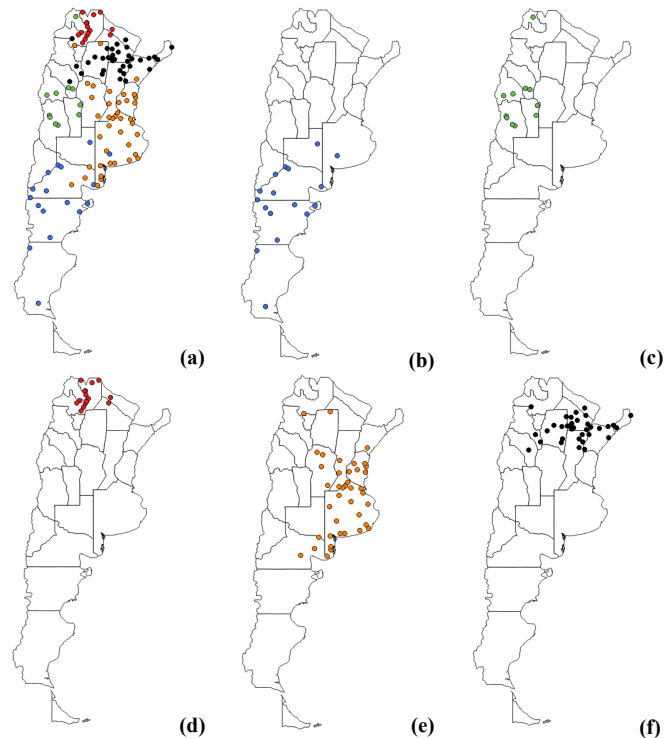


Fig. 7. (a) Distribución de todas las estaciones meteorológicas de Argentina con datos el día 6 de junio de 2016. (b) Estaciones meteorológicas en el grupo 0. (c) Estaciones meteorológicas en el grupo 1. (d) Estaciones meteorológicas en el grupo 2. (e) Estaciones meteorológicas en el grupo 3. (f) Estaciones meteorológicas en el grupo 4.

Las reglas de caracterización permiten diferenciar los distintos grupos con una certeza del 83,19%. En primer lugar, el grupo 0 posee una temperatura media menor a $9,85^{\circ}\text{C}$, temperatura mínima mayor a $-0,65^{\circ}\text{C}$ y una temperatura máxima menor a $13,5^{\circ}\text{C}$, distinguiéndose por este último valor del grupo 3, el cual posee una temperatura máxima mayor o igual a $13,5^{\circ}\text{C}$.

Por otro lado, el grupo 1 se caracteriza por sus bajas temperaturas media y mínima, siendo estas menores a $9,85^{\circ}\text{C}$ y $-0,65^{\circ}\text{C}$ respectivamente, y siendo estos mismos valores mayores a $9,85^{\circ}\text{C}$ y $7,2^{\circ}\text{C}$ en el grupo 2.

Por último, el grupo 4 se caracteriza por temperaturas medias mayores a $9,85^{\circ}\text{C}$ y temperaturas mínimas menores a $7,2^{\circ}\text{C}$.

5 Conclusiones

Se ha diseñado un proceso de explotación de información para el descubrimiento de reglas de caracterización de grupos espacialmente referenciados que permite la generación automática de grupos de objetos espaciales y la obtención de las características de cada uno de ellos utilizando algoritmos TDIDT. La caracterización

de estos grupos independientemente de que sean regiones, puntos calientes o grupos propiamente dichos.

Se han presentado dos pruebas de concepto que ilustran el funcionamiento del proceso propuesto utilizando datos reales. La primera utiliza algoritmos de generación de regiones, y la segunda utiliza algoritmos de generación de grupos. En ambos casos el algoritmo utilizado se compuso con un algoritmo de la familia TDIDT.

Queda pendiente evaluar el proceso propuesto para la caracterización de zonas calientes, que en esta etapa no se ha hecho por estar involucrada la misma familia de algoritmos que en la caracterización de grupos espaciales propiamente dicho.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por el Programa Formación de Doctores para Fortalecer áreas de I+D+i (2016-2020) de la Universidad Tecnológica Nacional (Argentina) y por los Proyectos de Investigación 33B133 y 33A205 de la Secretaria de Ciencia y Técnica de la Universidad Nacional de Lanús (Argentina).

Referencias

- Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.
- Brimicombe, A. J. (2007). A dual approach to cluster discovery in point event data sets. *Computers, environment and urban systems*, 31(1), 4-18.
- Britos, P. V. (2008). Procesos de explotación de información basados en sistemas inteligentes Tesis de Doctorado en Ciencias Informáticas. Facultad de Informática. Universidad Nacional de La Plata.
- Deng, M., Liu, Q., Cheng, T., & Shi, Y. (2011). An adaptive spatial clustering algorithm based on Delaunay triangulation. *Computers, Environment and Urban Systems*, 35(4), 320-332.
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, No. 34, pp. 226-231).
- Guo, D. (2008). Regionalization with dynamically constrained agglomerative clustering and partitioning (REDCAP). *International Journal of Geographical Information Science*, 22(7), 801-823.
- Hinneburg, A., & Keim, D. A. (1998, August). An efficient approach to clustering in large multimedia databases with noise. In *KDD* (Vol. 98, pp. 58-65).
- INDEC. (2010a). Cuadro V1. Total del país. Total de viviendas por provincia. Año 2010. Disponible en: www.indec.gov.ar/definitivos_bajarArchivoNacionales.asp?idc=1&arch=x&c=2010. Accedido el 1 de junio de 2016.
- INDEC. (2010b). Cuadro P 3. Total del país. Población total, superficie y densidad por provincia. Años 2001~2010. Disponible en: www.indec.gov.ar/definitivos_bajarArchivoNacionales.asp?idc=10&arch=x&c=2010. Accedido el 1 de junio de 2016.
- INDEC. (2010c). Cuadro P17. Total del país. Edad mediana de la población por sexo, según provincia. Año 2010. Disponible en: www.indec.gov.ar/definitivos_bajarArchivoNacionales.asp?idc=24&arch=x&c=2010. Accedido el 1 de junio de 2016.
- INDEC. (2010d). Cuadro P18. Total del país. Envejecimiento de la población por provincia, según censos nacionales 1970 a 2010. Disponible en: www.indec.gov.ar/definitivos_bajarArchivoNacionales.asp?idc=25&arch=x&c=2010. Accedido el 1 de junio de 2016.

- INDEC. (2010e). Cuadro P46. Total del país. Población indígena o descendiente de pueblos indígenas u originarios en viviendas particulares por tipo de cobertura de salud, según provincia. Año 2010. Disponible en: www.indec.gov.ar/definitivos_bajarArchivoNacionales.asp?idc=62&arch=x&c=2010. Accedido el 1 de junio de 2016.
- INTA. (2016). Datos Diarios. SIGA - Sistema de Información y Gestión Agrometeorológica. Instituto Nacional de Tecnología Agropecuaria. Argentina. Disponible en: <http://siga2.inta.gov.ar/en/datosdiarios/>. Accedido el 6 de junio de 2016
- Kataria, P., & Rupal, N. (2012). Mining Spatial Data & Enhancing Classification Using Bio-Inspired Approaches. *International Journal of Science and Research (IJSR)*. 3(2), 1473 – 1479.
- Liu, Q., Deng, M., Shi, Y., & Wang, J. (2012). A density-based spatial clustering algorithm considering both spatial proximity and attribute similarity. *Computers & Geosciences*, 46, 296-309.
- Mennis, J., & Guo, D. (2009). Spatial data mining and geographic knowledge discovery—An introduction. *Computers, Environment and Urban Systems*, 33(6), 403-408.
- Nisa, K. K., Andrianto, H. A., & Mardhiyyah, R. (2014, October). Hotspot clustering using DBSCAN algorithm and shiny web framework. In *Advanced Computer Science and Information Systems (ICACSIS), 2014 International Conference on* (pp. 129-132). IEEE.
- Popat, S. K., & Emmanuel, M. (2014). Review and Comparative Study of Clustering Techniques. *International Journal of Computer Science and Information Technologies*, 5(1), 805-812.
- Quinlan, J. R. (1993). C4. 5: programs for machine learning.
- Rakotomalala, R. (2005). TANAGRA: a free software for research and academic purposes. In *Proceedings of EGC (Vol. 2, pp. 697-702)*.
- Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (1998). Density-based clustering in spatial databases: The algorithm gbscan and its applications. *Data mining and knowledge discovery*, 2(2), 169-194.
- Santoso, A., & Nisa, K. K. (2016, January). Cloud Computing Application for Hotspot Clustering Using Recursive Density Based Clustering (RDBC). In *IOP Conference Series: Earth and Environmental Science (Vol. 31, No. 1, p. 012004)*. IOP Publishing.
- Shah, G. H., Bhensdadia, C. K., & Ganatra, A. P. (2012). An empirical evaluation of density-based clustering techniques. *International Journal of Soft Computing and Engineering (IJSCE) ISSN, 2231-2307*.
- Yang, X., & Cui, W. (2008, December). A novel spatial clustering algorithm based on Delaunay triangulation. In *International Conference on Earth Observation Data Processing and Analysis* (pp. 728530-728530). International Society for Optics and Photonics.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), 10-18.
- Zhong, C., Miao, D., & Wang, R. (2010). A graph-theoretical clustering method based on two rounds of minimum spanning trees. *Pattern Recognition*, 43(3), 752-766.

Proceso de Descubrimiento de Patrones de Co-Localización alrededor de Tipos de Eventos de Referencia

Giovanni Daián Rottoli^{1,2,3}, Hernán Merlino³, Ramón García-Martínez³

¹ Programa de Doctorado en Ciencias Informáticas. Universidad Nacional de La Plata. Argentina.

² Programa de Becas “Formación de Doctores para Fortalecer áreas de I+D+i”. Universidad Tecnológica Nacional. Argentina

³ Grupo de Investigación en Sistemas de Información. Universidad Nacional de Lanús. Argentina.

gd.rottoli@gmail.com, hmerlino@gmail.com, rgm1960@yahoo.com

Resumen. El descubrimiento de patrones de co-localización revela subconjuntos de tipos de eventos espaciales cuyas instancias ocurren frecuentemente vecinas entre sí. Muchos algoritmos y métodos han sido desarrollados a través de los años, sin embargo, cuando se requiere encontrar estos patrones alrededor de tipos de eventos espaciales determinados, la alternativa existente resulta incompleta e incorrecta. En el presente trabajo, en consecuencia, se desarrolla un proceso de explotación de información para el descubrimiento de patrones de co-localización alrededor de tipos de eventos espaciales de referencia que utiliza cliques máximos y algoritmos TDIDT para brindar una solución a este problema. Se presenta una prueba de concepto del proceso propuesto.

Keywords: Patrones de Co-Localización, TDIDT, Cliques Máximos, Explotación de Información.

1 Introducción

Dado un conjunto de tipos de eventos espaciales booleanos y una relación de vecindad, el descubrimiento de patrones de co-localización permite encontrar subconjuntos de dichos tipos de eventos cuyas instancias se encuentran ubicadas frecuentemente vecinas entre sí [Shekhar & Huang, 2001]. En este contexto, se entiende evento espacial como un suceso que ocurre en un lugar del espacio determinado. En consecuencia, un tipo de evento espacial hace referencia a la clase de suceso que ocurre.

Para conseguir este objetivo, se han propuesto distintos algoritmos y métodos basados en análisis de asociaciones, los cuales se pueden dividir en dos grupos. El primero comprende a los Algoritmos No Transaccionales (*Transaction-Free Algorithms*), los cuales utilizan internamente algoritmos de minería de datos para el descubrimiento de reglas de asociación sobre la información correspondiente a instancias de eventos espaciales. Por otro lado, el segundo grupo abarca los algoritmos Transaccionales (*Transaction-Based Algorithms*), que generan información transaccional a partir de las instancias de eventos espaciales, para ser

utilizada como entrada de algoritmos de descubrimiento de reglas de asociación de manera explícita, siendo un enfoque más eficiente que la alternativa no transaccional. [Shekhar & Huang, 2001; Shekhar et al., 2011, Kim et al., 2014].

Por otro lado, existen tres diferentes modelos de generación de transacciones para resolver el problema de descubrimiento de patrones de co-localización [Shekhar & Huang, 2001; Xiong et al., 2004]. La primera forma, denominada Modelo Centrado en Eventos (*Event Centric Model*), se utiliza cuando existen muchos tipos de eventos espaciales y se desea encontrar subconjuntos de los mismos que sucedan frecuentemente juntos y ha sido ampliamente utilizada en trabajos como [Agrawal & Srikant, 1994; Shekhar & Hung, 2001; Huang et al., 2003, 2006; Yoo et al., 2004; Xiong et al., 2004; Yoo & Shekhar, 2006; Celik et al., 2007; Eick, 2008; Adilmagambetov et al., 2013; Kim et al., 2011, 2014].

El segundo modelo de generación de transacciones, denominado Modelo Centrado en Ventanas (*Window Centric Model*), permite descubrir patrones dentro de subdivisiones del espacio de datos, llamadas ventanas. Este modelo es utilizado en áreas como la Minería, donde cada ventana se correspondería con parcelas de terreno.

Por último, el tercer modelo es llamado Centrado en Tipos de Eventos de Referencia (*Reference Feature Centric Model*) o Basado en Referencias (*Reference Based*), consiste en encontrar patrones de co-localización generando transacciones alrededor de determinados tipos de eventos espaciales, utilizándose por ejemplo para la determinación de factores ambientales condicionantes de casos de Cáncer.

De estos modelos existentes es notable mencionar que el modelo centrado en tipos de eventos de referencia no ha sido implementado en demasiados algoritmos y métodos para el descubrimiento de patrones de co-localización, resultando incorrectos e incompletos aquellos que sí lo implementan [Adilmagambetov et al., 2013; Kim et al., 2014]. Por estas razones, en la sección 2 del presente trabajo se introduce a la problemática derivada del análisis del estado del arte, se presenta en la sección 3 una solución al problema planteado, se realizan en la sección 4 pruebas de concepto mediante herramientas estadísticas, y por último se presentan conclusiones en la sección 5.

2 Definición del problema

Ante una gran cantidad de tipos de eventos espaciales las búsquedas de patrones de co-localización con generación de transacciones basada en eventos puede resultar extenuante, demandando entonces una gran cantidad de recursos.

Ante la presencia de cierto tipo de evento de interés para un dominio del problema dado, un enfoque de generación de transacciones alrededor de tipos de eventos de referencia, resulta una alternativa más adecuada, sin embargo, la solución que utiliza este enfoque utiliza verificaciones de la relación de vecindad de cada evento espacial, con las instancias del tipo de evento espacial seleccionado como referencia, junto con consideraciones especiales en los cálculos de prevalencia para la creación de patrones [Shekhar & Huang, 2001; Xiong et al., 2004].

Este acercamiento propuesto, sin embargo, no permite generar transacciones correctas, debido a que no se asegura que todos los elementos de la misma sean vecinos entre sí, o completas, pudiéndose perder ciertos vecindarios en el proceso

[Adilmagambetov et al., 2013; Kim et al., 2014]. Este motivo hace necesaria el desarrollo de una solución que permita el descubrimiento de patrones de co-localización alrededor de tipos de eventos espaciales determinados de una forma correcta y completa.

En el presente trabajo se desarrolla un proceso de explotación de información [Britos, 2008] para dar solución a este problema.

3 Solución propuesta

Como se mencionó anteriormente, se propone en el presente trabajo un proceso de explotación de información que permita la obtención de patrones de co-localización correctos y completos alrededor de tipos de eventos espaciales determinados.

A tal fin, se toma como base el trabajo de Kim et al., (2014), en el cual se desarrolla un *framework* transaccional para el descubrimiento de patrones de co-localización, y se evalúa la conveniencia de utilizar cliques máximos sobre las relaciones de vecindad como forma de generar transacciones que aseguren completitud y correctitud.

Dado un grafo de vecindad un clique es un subgrafo completo, lo cual significa que en sí mismo todos los nodos son vecinos entre sí. Un clique máximo, entonces, es un clique de dicho grafo que no está incluido en ningún otro clique. Cada clique máximo correspondería entonces a una transacción donde todos los elementos son vecinos entre sí asegurando la correctitud del método [Kim et al., 2014, Lemma 1]. A su vez, la utilización de cliques máximos como transacciones asegura la completitud del método, al estar todas las relaciones de vecindad consideradas por lo menos en un cliqué máximo [Kim et al., 2014, Lemma 2].

A fin de resolver el problema presentado, se propone un proceso de explotación de información para el descubrimiento de patrones de co-localización alrededor de tipos de eventos espaciales de interés que utiliza cliques máximos bajo un modelo centrado en eventos, sometiendo a las transacciones creadas a un proceso de explotación de información para el descubrimiento de reglas de comportamiento utilizando algoritmos de la familia *Top-Down Induction of Decision Trees* – TDIDT – [Britos, 2008].

Como puede observarse en la Figura 1, el proceso parte de un conjunto de información espacial representada en distintos formatos (texto plano, bases de datos, mapas referenciados geográficamente, entre otras), la cual es integrada para formar un único repositorio donde conste, por cada evento espacial, un identificador de dicha instancia, el tipo de evento espacial al cual pertenece, y su ubicación en el espacio.

Posteriormente, la información integrada es utilizada para generar información transaccional. Este subproceso, como se observa en la Figura 2, calcula primero todas las relaciones de vecindad entre las instancias de eventos espaciales, evaluando la distancia que existe entre ellos, para posteriormente encontrar los cliques máximos en el grafo de vecindad construido, generando una transacción por cada uno de ellos, en la cual conste los tipos de eventos espaciales de cada evento que forma parte de dicho clique.

Una vez obtenida la información transaccional, es necesario especificar el tipo de evento espacial alrededor de las cuales se desea hallar los patrones de co-localización

para ser utilizado como atributo objetivo o *target* de un algoritmo TDIDT, utilizando los demás tipos de evento espacial como atributos de entrada del mismo.

Como salida del paso anterior obtendremos un conjunto de reglas de comportamiento, en función al árbol de decisión generado. Debido a que las transacciones poseen valores booleanos que evidencian la presencia o ausencia de los tipos de evento espacial presentes en los vecindarios, es necesario filtrar solo aquellas que muestren como consecuente la presencia del tipo de evento seleccionado como objetivo.

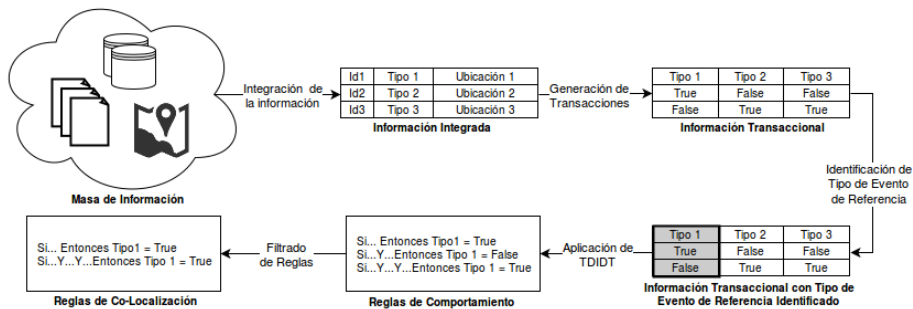


Fig. 1. Proceso para el descubrimiento de patrones de co-localización alrededor de tipos de eventos de referencia

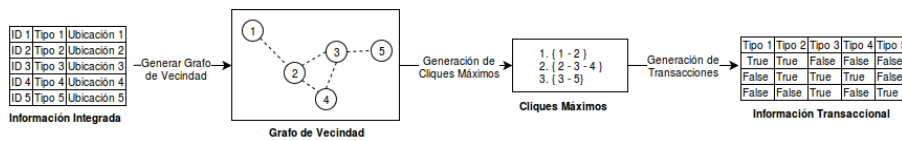


Fig. 2. Generación de transacciones

Esta secuencia de pasos hace posible la reutilización de la información transaccional para descubrir patrones de co-localización alrededor de distintos tipos de evento espacial de interés, sin necesidad de realizar el cálculo de vecindarios en cada oportunidad.

Las reglas obtenidas, por otro lado, no solo describen los tipos de eventos espaciales vecinos, sino también las condiciones que deben reunir los vecindarios, esto es, si es necesaria en algunos casos la ausencia de ciertos tipos de eventos, agregándose de esta forma información a los resultados.

4 Prueba de Concepto

Se realiza una prueba de concepto comparando el proceso propuesto con un algoritmo basado en tipos de eventos de referencia para determinar si el primero puede detectar mayor cantidad de patrones correctos que los métodos existentes hasta el momento.

Para ello, se cuenta con 10 conjuntos sintéticos de 500 puntos generados y clasificados en 7 tipos de manera aleatoria, con coordenadas en el plano en el intervalo [0; 40] tanto del eje de las ordenadas como del de las abscisas. A modo de ejemplo, la Figura 3 ilustra la distribución de los puntos del primer conjunto de datos, donde cada símbolo representa un tipo de evento diferente.

Los conjuntos en cuestión son usados como entrada tanto del proceso propuesto, cómo del algoritmo seleccionado para comparar: Co-Location Miner bajo un modelo centrado en tipos de evento de referencia [Shekhar & Huang, 2001].

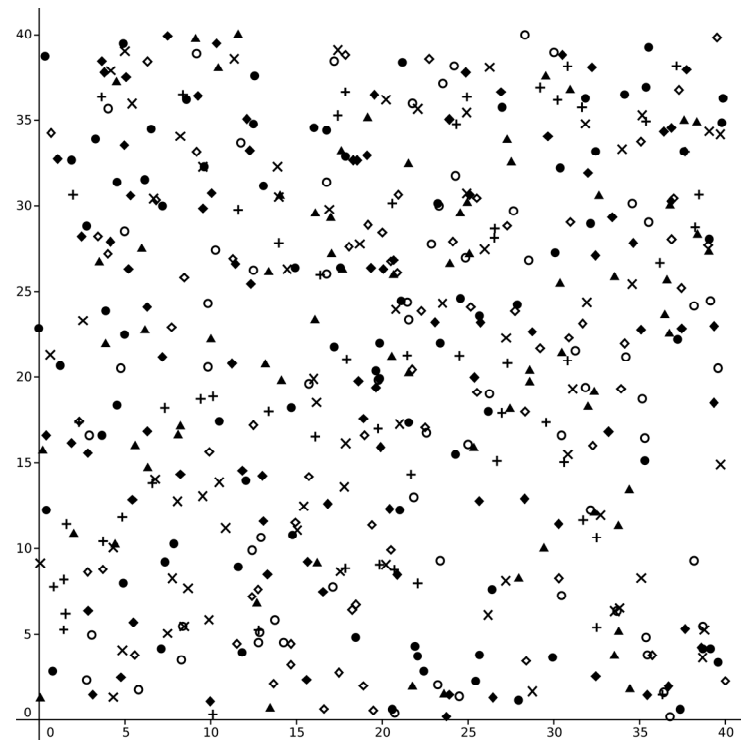


Fig. 3. Distribución de puntos del primer conjunto sintético de datos, donde cada símbolo corresponde a un tipo de evento diferente

Por otro lado, ejecución del proceso de explotación de información se ha realizado calculando en una primera instancia las relaciones de vecindad entre los puntos, verificando si la distancia entre cada par de puntos es menor al cierto umbral de vecindad especificado. En esta etapa nos valemos de la propiedad de simetría de la función de distancia a fin de reducir las comparaciones y aumentar la velocidad de ejecución del proceso. Se utilizó además el algoritmo CLIQUES para la generación de cliques máximos sobre las relaciones de vecindad por demostrar eficiencia superior a otros métodos [Uno, 2005; Tomita et al., 2006], y el software Tanagra [Rakotomalala, 2005] para la ejecución del algoritmo TDIDT seleccionado, C4.5 [Quinlan, 1993].

Luego de la ejecución de tanto el proceso presentado como del algoritmo Co-Location Miner, los patrones de co-localización obtenidos fueron evaluados para corroborar la correctitud de los mismos, a fin de determinar la cantidad de patrones correctos detectados en cada oportunidad. Para demostrar que el proceso propuesto posee un mejor comportamiento, se utilizó el test estadístico no paramétrico de rangos con signo de Wilcoxon [Wilcoxon, 1945], buscando rechazar la hipótesis nula H_0 y aceptar la hipótesis alternativa H_A , siendo estas las que se muestran en la tabla 1.

La ejecución del test de rangos con signos de Wilcoxon puede observarse en la Tabla 2, junto con los valores obtenidos durante las pruebas, utilizándose posteriormente el método del P-Valor sobre la suma de los rangos positivos con un nivel de significación del 1%, obteniéndose un valor igual a 0,0038, por lo cual se rechaza la hipótesis nula, aceptándose la alternativa en consecuencia, confirmando de esta forma que el proceso de explotación de información permite encontrar mayor cantidad de reglas correctas que el método existente hasta el momento con un 99% de confianza.

Tabla 1. Hipótesis nula e hipótesis alternativa consideradas en el test de rangos con signos de Wilcoxon

H_0 :	La cantidad de Patrones Correctos detectados por el algoritmo Co-Location Miner es Mayor o Igual a la cantidad detectada por el proceso propuesto.
H_A :	La cantidad de Patrones Correctos detectados por el proceso propuesto es Mayor a la cantidad detectada por el algoritmo Co-Location Miner.

Tabla 2. Ejecución del test de rangos con signo de Wilcoxon sobre los conjuntos de datos sintéticos, ordenados según las diferencias positivas

Conjunto	Patrones correctos con el proceso propuesto	Patrones correctos con Co-Location Miner	Diferencias positivas	Rangos
Conjunto 7	3	3	0	-
Conjunto 8	2	1	1	2
Conjunto 9	2	1	1	2
Conjunto 10	5	4	1	2
Conjunto 1	3	1	2	5
Conjunto 2	3	1	2	5
Conjunto 5	4	2	2	5
Conjunto 4	4	1	3	7
Conjunto 3	6	2	4	8.5
Conjunto 6	7	3	4	8.5
Suma de rangos:				45

5 Conclusiones

Se ha presentado un proceso de explotación de información para el descubrimiento de patrones de co-localización de manera correcta y completa. El proceso propuesto utiliza cliques máximos para la generación de información transaccional, y la

aplicación a esta de algoritmos TDIDT para la obtención de reglas de comportamiento sobre los tipos de eventos espaciales asociados, resultando un método original a tal fin.

Se ha presentado una prueba de concepto que muestra con un método estadístico no paramétrico un mejor desempeño del proceso propuesto frente al que actualmente está en uso.

El proceso propuesto permite además (i) la búsqueda de patrones alrededor de distintos tipos de eventos espaciales sin necesidad de realizar el cálculo del grafo de vecindad en cada oportunidad, y (ii) agregar información a los resultados determinando bajo qué condiciones ocurren los patrones hallados.

Como próximo paso se prevé realizar casos de validación en los dominios de accidentología, incidentes de defensa civil, y factores eco-ambientales condicionantes de enfermedades.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por el Programa para la Formación de Doctores para Fortalecer áreas de I+D+i (2016-2020) de la Universidad Tecnológica Nacional (Argentina) y por los Proyectos de Investigación 33B133 y 33A205 de la Secretaria de Ciencia y Técnica de la Universidad Nacional de Lanús (Argentina).

Referencias

- Adilmagambetov, A., Zaiane, O. R., & Osornio-Vargas, A. (2013). Discovering co-location patterns in datasets with extended spatial objects. In *Data Warehousing and Knowledge Discovery* (pp. 84-96). Springer Berlin Heidelberg.
- Agrawal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB* (Vol. 1215, pp. 487-499).
- Britos, P. V. (2008). *Procesos de explotación de información basados en sistemas inteligentes*. Tesis de Doctorado en Ciencias Informáticas. Facultad de Informática. Universidad Nacional de La Plata.
- Celik, M., Kang, J. M., & Shekhar, S. (2007, October). Zonal co-location pattern discovery with dynamic parameters. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on* (pp. 433-438). IEEE.
- Chicago Police Department (2016). *Reported Incidents occurred in the City of Chicago from 2001 to present* [On-Line]. Chicago, USA. [Consultado el 3 de Febrero de 2016]. Disponible en: <https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2>
- Eick, C. F., Parmar, R., Ding, W., Stepinski, T. F., & Nicot, J. P. (2008, November). Finding regional co-location patterns for sets of continuous variables in spatial datasets. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems* (p. 30). ACM.
- Huang, Y., Xiong, H., Shekhar, S., & Pei, J. (2003, March). Mining confident co-location rules without a support threshold. In *Proceedings of the 2003 ACM symposium on Applied computing* (pp. 497-501). ACM.

- Huang, Y., Pei, J., & Xiong, H. (2006). Mining co-location patterns with rare events from spatial data sets. *Geoinformatica*, 10(3), 239-260.
- Kim, S. K., Kim, Y., & Kim, U. (2011). Maximal cliques generating algorithm for spatial co-location pattern mining. In *Secure and Trust Computing, Data Management and Applications* (pp. 241-250). Springer Berlin Heidelberg.
- Kim, S. K., Lee, J. H., Ryu, K. H., & Kim, U. (2014). A framework of spatial co-location pattern mining for ubiquitous GIS. *Multimedia tools and applications*, 71(1), 199-218.
- Quinlan, J. R. (1993). C4. 5: programs for machine learning.
- Rakotomalala, R. (2005). TANAGRA: a free software for research and academic purposes. In *Proceedings of EGC (Vol. 2, pp. 697-702)*.
- Shekhar, S., & Huang, Y. (2001). Discovering spatial co-location patterns: A summary of results. In *Advances in Spatial and Temporal Databases* (pp. 236-256). Springer Berlin Heidelberg.
- Shekhar, S., Evans, M. R., Kang, J. M., & Mohan, P. (2011). Identifying patterns in spatial information: A survey of methods. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(3), 193-214.
- Tomita, E., Tanaka, A., & Takahashi, H. (2006). The worst-case time complexity for generating all maximal cliques and computational experiments. *Theoretical Computer Science*, 363(1), 28-42.
- Uno, T. (2005). MACE_GO: MAXimal Clique Enumerator (CLIQUES Implementation) [C Code]. Versión 2.0.. Disponible desde: <http://research.nii.ac.jp/~uno/code/macego10.zip>
- Venkatesan, M., Thangavelu, A., & Prabhavathy, P. (2011). Event Centric Modeling Approach in Colocation Pattern Snalysis from Spatial Data. arXiv preprint arXiv:1109.1144.
- Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6), 80-83.
- Xiong, H., Shekhar, S., Huang, Y., Kumar, V., Ma, X., & Yoo, J. S. (2004, April). A Framework for Discovering Co-Location Patterns in Data Sets with Extended Spatial Objects. In *SDM* (pp. 78-89).
- Yoo, J. S., Shekhar, S., Smith, J., & Kumquat, J. P. (2004, November). A partial join approach for mining co-location patterns. In *Proceedings of the 12th annual ACM international workshop on Geographic information systems* (pp. 241-249). ACM.
- Yoo, J. S., Shekhar, S., & Celik, M. (2005, November). A join-less approach for co-location pattern mining: A summary of results. In *Data Mining, Fifth IEEE International Conference on* (pp. 4-pp). IEEE.
- Yoo, J. S., & Shekhar, S. (2006). A joinless approach for mining spatial colocation patterns. *Knowledge and Data Engineering, IEEE Transactions on*, 18(10), 1323-1337.

Clasificación de Distintos Conjuntos de Datos Utilizados en Evaluación de Métodos de Extracción de Conocimiento Creados para la Web

Juan M. Rodríguez^{1,2}, Hernán D. Merlino², Patricia Pesado¹, Ramón García-Martínez²

¹ Programa de Doctorado en Ciencias Informáticas. Facultad de Informática.
Universidad Nacional de La Plata. Argentina.

² Grupo de Investigación en Sistemas de Información. Departamento de Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús. Argentina.
jmrodriguez1982@gmail.com, hmerlino@gmail.com,
pjesado@lidi.info.unlp.edu.ar, rgm1960@yahoo.com

Resumen. En varios artículos se han utilizado distintos textos de prueba, como datos de entrada para medir el desempeño de los métodos de extracción de relaciones semánticas para la Web (OIE): ReVerb y ClausIE. Sin embargo estos textos nunca han sido analizados para entender si ellos guardan o no similitudes o para saber si existe entre ellos un lenguaje común o pertenecen a un mismo dominio. Es la intención de este trabajo analizar dichos textos utilizando distintos algoritmos de clasificación. Y comprender si se pueden agrupar de una forma coherente, de tal suerte que a priori uno pueda identificar que textos son los que trabajan mejor con ClausIE y cuales con ReVerb.

Palabras Clave. Extracción de conocimiento, extracción de relaciones semánticas, métodos de extracción auto-supervisados, *open information extraction*, procesamiento de lenguaje natural, clasificación de textos, Bayes Naive, SMO, J48.

1 Introducción

En [Rodríguez et al., 2015] fue realizada una investigación documental sobre distintos métodos de extracción de relaciones semánticas para la Web (*Open Information Extraction*, en inglés según la definición dada en [Banko et al., 2007]). En dicha investigación se relevaron nueve métodos de extracción de relaciones semánticas utilizando comparaciones y experimentos publicados por distintos autores. El resultado del trabajo de [Rodríguez et al., 2015] es el trazado de una línea evolutiva de estos distintos métodos en función del tiempo con la conclusión de que el método de extracción de relaciones semánticas para la Web más preciso (ver fórmula 1) es ClausIE (propuesto en [Del Corro & Gemulla, 2013]), seguido por OLLIE (propuesto en [Schmitz, 2012]), y en tercer lugar por ReVerb (propuesto en [Fader et al., 2011]). Esta conclusión estuvo basada principalmente en los resultados de los experimentos publicados en los artículos: [Del Corro & Gemulla, 2013; Schmitz et al., 2012].

En [Rodríguez et al., 2016] se realizó un experimento para verificar el supuesto anterior sobre la precisión de los distintos métodos. Para ello se utilizó un conjunto de

datos de entrada creado con 55 cables de noticias elegidos de forma aleatoria de la base de datos Reuters-21578 [Lewis, 1997]. El resultado de dicho experimento se resume en la Tabla 1.

Tabla 1. Precisión, Exhaustividad (*recall*) y Medida-F1

Método	Precisión	Exhaustividad	Medida-F1
ClausIE	0.513	0.503	0.508
ReVerb	0.671	0.355	0.464
OLLIE	0.488	0.420	0.451

Contrariamente a lo esperado el método más preciso resultó ser ReVerb, para este conjunto de datos de entrada. El experimento incluyó además la medición de la Exhaustividad (*recall* en inglés) y una medida que combina ambas (Medida-F1). Las fórmulas de estas tres medidas de rendimiento se dan a continuación:

$$\text{Precisión} = \frac{\text{cantidad de piezas de conocimiento extraídas correctamente}}{\text{cantidad_de_piezas_de_ conocimiento_extraídas}} \quad (1)$$

$$\text{Exhaustividad} = \frac{\text{cantidad de piezas de conocimiento extraídas correctamente}}{\text{cantidad_de_piezas_de_conocimiento_totales_en_el_documento}} \quad (2)$$

$$F1 = \frac{2 \cdot \text{precisión} \cdot \text{exhaustividad}}{\text{precisión} + \text{exhaustividad}} \quad (3)$$

En donde una “pieza de conocimiento” refiere de forma genérica a una relación semántica, ya que en términos más generales una relación semántica es una representación estructurada (manipulable en procesos de razonamiento automático) del conocimiento embebido en una fuente de datos, en principio no estructurada, como lo es el lenguaje natural [García-Martínez & Britos, 2004; Gómez et al., 1997].

En la fórmula 2, para calcular la cantidad total de piezas de conocimiento (extraídas y no extraídas), se realizó una extracción manual de las relaciones semánticas halladas en cada cable de noticias y se sumaron además las extracciones correctas realizadas por el algoritmo que no habían sido descubiertas durante la extracción manual.

1.1 Hipótesis

Los 55 cables de noticias evaluados permiten afirmar que Reverb es el método más preciso para extraer relaciones semánticas de la base Reuters-21578, con un nivel de confianza del 86%, considerando un margen de error del 10% [Rodríguez et al., 2016]. Sin embargo en los experimentos realizados en [Del Corro & Gemulla, 2013], en donde se utilizaron tres conjuntos de datos diferentes: 200 oraciones extraídas aleatoriamente del *New York Times collection* [Sandhaus E., 2008], 200 oraciones extraídas aleatoriamente de páginas de Wikipedia y 500 oraciones tomadas del servicio *Yahoo's random link* (este es el conjunto de datos utilizados originalmente para probar ReVerb en [Fader et al., 2011]) se obtuvo como resultado que ClausIE es un método más preciso que ReVerb.

A partir de estos resultados, incompatibles entre sí, proponemos como hipótesis de investigación que el conjunto de datos de entrada, es decir el dominio al cual pertenece un texto escrito en lenguaje natural, es relevante a la hora de extraer las relaciones semánticas embebidas en él. Para probar esta hipótesis será necesario encontrar una forma de catalogar o clasificar los textos de entrada a partir de una o varias de sus características.

2 Experimentación

La demostración de la hipótesis propuesta, fue abordada como un problema de clasificación de textos. Para ello se utilizaron los siguientes algoritmos de clasificación: Bayes Naïv, Bayes Naïv Multinomial, SMO y J48. Los cuatro algoritmos son parte de Weka [Hall et al., 2009], la herramienta de investigación y desarrollo utilizada en el presente trabajo.

2.1 Bayes Naïve y Bayes Naïve Multinomial

Bayes Naïve es uno de los modelos más simples y parte del supuesto de que todos los atributos de un documento de entrada (por ejemplo unigramas) son independientes entre sí en el contexto de una clase, esto es llamado "el supuesto de *Naïve Bayes*" [Mccallum et al., 1998]. A pesar de que este supuesto es en verdad falso, en la mayoría de las tareas del mundo real *Naïve Bayes* realiza buenas clasificaciones [Mccallum et al., 1998].

Este algoritmo de clasificación busca identificar a la clase que maximice el resultado de la multiplicación entre la probabilidad de una clase dada y las probabilidades individuales de las palabras dada dicha clase, matemáticamente:

$$C_{\text{map}} = \operatorname{argmax}_{c_i} P(c_i) \prod_{x \in X} P(w | c_i) \quad (4)$$

Las probabilidades de las distintas clases (c) junto con las probabilidades de cada una de las palabras (w) de pertenecer a una clase son estimadas en el conjunto de entrenamiento.

Una de las razones por las cuales fue seleccionado *Naïve Bayes* como algoritmo de clasificación, es por el trabajo de [Banko & Brill, 2001], en donde se utilizaron tres clasificadores distintos: (a) *Winnnow* [Golding & Roth, 1999], (b) *Perceptron* [NG et al., 1997] y (c) *Naïve Bayes*, se observó que estos tienden a converger para grandes volúmenes de datos. Otro de los motivos es que *Naïve Bayes* es un algoritmo rápido, incluso con grandes cantidades de datos [Lowd & Domingos, 2005].

Por su parte la implementación de *Bayes Naïve Multinomial* que viene incluida en la herramienta WEKA es una mejora respecto al algoritmo tradicional de *Bayes Naïve* que consiste en lo siguiente:

- No considera de igual forma la ocurrencia de una palabra como su ausencia. Usa la probabilidad de la misma.
- Tiene en cuenta las múltiples repeticiones de una palabra en un documento
- No trata a todas las palabras de la misma forma, discrimina entre palabra más frecuentes y menos frecuentes, etc.

Ambos algoritmos fueron testeados contra un mismo conjunto de documentos de prueba extraídos de Reuters-21578 en [Hall et al., 2009] y el resultado fue que *Bayes Naïve Multinomial* obtuvo una precisión de 0.91 contra una precisión de 0.8 que obtuvo el algoritmo original *Bayes Naïve*.

2.2 SMO

El segundo método escogido para realizar la clasificación de textos y poder comparar resultados fue una implementación de *support vector machines* (SVMs) [Joachims, 1998] llamada *Sequential Minimal Optimization* (SMO): la cual consiste en una mejora en el algoritmo de entrenamiento de SVMs, de forma tal que este llega a ser 1200 veces más rápido para SVMs lineales y 15 veces más rápido para SVMs no lineales [Platt, 1998].

Las SVMs como método de clasificación son muy populares y ampliamente utilizadas [Kolesov et al., 2014] debido a su éxito, no solo para clasificar textos sino también para diversos problemas de clasificación. Sang-Bum Kim en [Kim et al., 2006] menciona que los clasificadores basados en complejos métodos de aprendizaje como los SVMs pertenecen al *state-of-the-art*.

Sequential Minimal Optimization (SMO) es un algoritmo simple que puede resolver rápidamente los grandes problemas de programación cuadrática (QP) de SVM sin utilizar una matriz de almacenamiento y sin utilizar pasos QP de optimización numérica. SMO descompone al problema QP general en sub-problemas QP, utilizando el teorema de Osuna para asegurar la convergencia [Platt, 1998].

2.3 J48

J48 es una implementación de código abierto del algoritmo C4.5 incluida en la herramienta Weka. C4.5 a su vez es un programa que crea arboles de decisión basados en un conjunto de datos de entrada previamente etiquetados. El algoritmo fue desarrollado por Ross Quinlan. Los arboles de decisión generados por el algoritmo C4.5 pueden ser utilizados para clasificación y suelen ser llamados clasificadores estadísticos [Gholap, 2012]. Este es un algoritmo es utilizado ampliamente para resolver problemas de aprendizaje automático [Witten et al., 1999].

El algoritmo trabaja partiendo la lista inicial de ejemplos etiquetados según aquellos atributos que más eficazmente dividen el conjunto inicial, utilizando para ello el concepto de entropía de información [Quinlan, 2014]. Aplicando esta técnica de forma recursiva por cada sub lista el algoritmo va construyendo un árbol.

2.4 Conjunto de datos de entrada

Para constituir el conjunto de datos de entrada se tomaron la totalidad de los datos utilizados para evaluar a los clasificadores y se construyó un primer gran conjunto de datos:

- Subconjunto de 55 cables de noticias la base Reuters-21578
- 200 oraciones extraídas aleatoriamente del *New York Times collection*
- 200 oraciones extraídas aleatoriamente de páginas de Wikipedia

- 500 oraciones tomadas del servicio *Yahoo's random link*

El conjunto anterior se utilizó para crear dos conjuntos: el conjunto “ReVerb” y el conjunto “ClausIE”. El conjunto “ReVerb” se conformó con todos aquellos textos del conjunto anterior, tal que, al ser utilizados como entrada de los métodos ReVerb y ClausIE, ReVerb extrajo relaciones semánticas con una precisión al menos 50% superior a ClausIE. A su vez el conjunto “ClausIE” quedó conformado con todos aquellos textos en los cuales ClausIE logró una precisión al menos 50% superior a ReVerb en la extracción de relaciones semánticas.

El conjunto “ClausIE” quedó constituido por 124 casos y el conjunto “ReVerb” por 58 casos. Cómo es posible observar los casos en los cuales ClausIE es más preciso que ReVerb son mayoría. Hay que tener en cuenta que la mayoría de los casos son los mismos que fueron utilizados en el trabajo de [Del Corro & Gemulla, 2013].

Además el texto original en idioma inglés fue convertido palabra por palabra a sus categorías gramaticales (*PosTags*) y también se lo convirtió en etiquetas de IOB [Ramshaw, 1995], según la técnica de *text-chunkin*. La técnica de *text-chunking*, consiste en dividir frases en segmentos de texto que no se superponen, en base a un análisis superficial. En [Abney, 1991] se propuso este método como un precursor útil y simple de implementar para detectar principalmente frases nominales y verbales [Ramshaw, 1995]. Para convertir una porción de texto en sus categorías gramaticales o bien para obtener las etiquetas de IOB correspondientes al *text-chunking* se utilizó el mismo programa ReVerb, solo que se lo modificó para que convierta el texto de la forma mencionada utilizando las mismas librerías que utiliza para extraer relaciones semánticas.

Para ilustrar lo anterior supongamos el siguiente texto en inglés como ejemplo:

She has done so with depth and confidence.

Convertido a categorías gramaticales (*pos-tags*) quedó como:

- PRP VBZ VBN RB IN NN CC NN .

Convertido a etiquetas IOB de al *text-chunking* quedó como:

- B-NP B-VP I-VP B-ADVP B-PP B-NP I-NP I-NP O

Además, tanto el texto en lenguaje natural, como las oraciones convertidas a categorías gramaticales y a etiquetas IOB de *text-chunking* fue por un lado clasificado tal cual, es decir como unigramas, pero fue convertido también a bigramas y trigramas. Siguiendo con el ejemplo anterior, la oración:

She has done so with depth and confidence.

Fue clasificada, según las siguientes maneras de agrupar palabras:

1. **Unigramas:** (*She, has, done, so, with, depth, and, confidence*)
2. **Bigramas:** (<start>-*She, She-has, has-done, done-so, so-with, with-depth, depth-and, and-confidence, confidence-<end>*)
3. **Trigramas:** (<start>-*She-has, She-has-done, has-done-so, done-so-with, so-with-depth, with-depth-and, depth-and-confidence, and-confidence-<end>*)

Este mismo tratamiento se le dio a la oración como categorías gramaticales y etiquetas IOB de *text-chunking*. Cada uno de los 4 métodos de clasificación fue entrenado 9 veces, una vez por cada conjunto de características: unigramas, bigramas y trigramas para el texto en lenguaje natural, luego unigramas, bigramas y trigramas para su versión en categorías gramaticales y nuevamente unigramas, bigramas y trigramas sobre las etiquetas IOB de *text-chunking*. De esta forma se intentó encontrar que características en un texto son más relevantes para que un método de extracción de relaciones semánticas actúe mejor que otro.

3 Resultados

A continuación se muestran los resultados en tres tablas distintas, en la primera tabla se muestran los resultados para el texto en lenguaje natural, en la segunda tabla los resultados para el texto convertido a sus respectivas categorías gramaticales y por último el texto convertido a etiquetas IOB de *text-chunking*. A su vez cada tabla muestra, para cada uno de los algoritmos de clasificación de texto seleccionados, un porcentaje de las instancias clasificadas correctamente en cada uno de los conjuntos de características de un texto. En todos los casos se utilizó el 66% de los datos para entrenar el algoritmo y el resto para validar el mismo.

Tabla 2. Texto en ingles

Clasificador	Unigramas	Bigramas	Trigramas
Bayes Naïve	77.4194	67.7419	70.9677
Bayes Naïve Multinomial	70.9677	51.6129	32.2581
SMO	64.5161	70.9677	69.3548
J48	53.2258	70.9677	69.3548

Tabla 3. Categorías gramaticales

Clasificador	Unigramas	Bigramas	Trigramas
Bayes Naïve	56.45	62.9	53.23
Bayes Naïve Multinomial	64.51	67.74	54.84
SMO	67.74	59.68	59.68
J48	54.84	72.58	58.06

Tabla 4. Etiquetas IOB de *text-chunking*

Clasificador	Unigramas	Bigramas	Trigramas
Bayes Naïve	62.90	61.29	59.68
Bayes Naïve Multinomial	69.35	61.29	66.13
SMO	67.74	58.06	59.68
J48	69.35	67.74	59.68

4 Conclusiones

De la tabla 4 se deduce que convertir un texto a etiquetas de *text-chunking*, técnica que utiliza ReVerb para realizar extracciones semánticas, no da buenos indicios acerca de si el texto será bien interpretado por ReVerb o no, tampoco por ClausIE. El porcentaje de 69.35 es muy bajo si tenemos en cuenta que solo se trabajó con dos conjuntos: ReVerb y ClausIE (ver punto 2.4). Menos de 20 puntos por encima de una clasificación completamente aleatoria. El valor más alto se registró en el conjunto de unigramas. Los bigramas y trigramas, es decir las secuencias de etiquetas de *text-chunking* aportan menos información que los simples unigramas.

La tabla 3 muestra un valor un poco más alto para el porcentaje de clasificaciones positivas. Las categorías gramaticales tomadas como bigramas, logran ser clasificadas con un porcentaje de aciertos de: 72.58%. La secuencia de categorías gramaticales (*PosTags*) son un indicio, aunque muy leve, acerca de qué método (y por ello de qué técnica) de extracción de relaciones semántica será más adecuada para un texto dado.

De la tabla 2 se desprende que es posible identificar, con una precisión mayor al 77%, si un texto será mejor analizado por ClausIE o bien ReVerb. Es decir que las relaciones semánticas presentes en un texto dado, podrán ser extraídas de forma más precisa con uno u otro método y eso solo depende de los unigramas presentes. Dado que los unigramas coinciden con las palabras, (la *tokenizacion* utilizada para crear los unigramas es la misma que utilizan ambos métodos para separar las palabras en una oración dada), es posible afirmar en el 77% de los casos, que si determinadas palabras están presentes en un texto entonces dicho texto será mejor analizado ReVerb (o bien por ClausIE según sea el caso).

A partir de lo anterior se puede concluir que:

- El dominio al cual pertenece un texto será crucial para saber cómo reaccionará cualquiera de los dos métodos de extracción de relaciones semánticas analizados.
- La secuencia de categorías gramaticales tiene una ligera influencia en el proceder de ambos métodos.
- Las secuencias de etiquetas IOB de *text-chunking* no dan mucha información acerca de cómo el texto será interpretado por cualquiera de los métodos.

5 Futuras líneas de investigación

A partir de las conclusiones extraídas en el punto 4, queda pendiente el trabajo de aplicar un algoritmos de clasificación de textos como un paso previo a la extracción de relaciones semánticas. Según el resultado de este algoritmo debería ser utilizado ClausIE o ReVerb. Si los supuestos en el punto 4 son correctos se espera una mejora notable en la calidad de las piezas de conocimiento extraídas en comparación con la que obtendrían los métodos trabajando por separado.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por los Proyectos de Investigación 33B177 y 33A205 de la Secretaría de Ciencia y Técnica de la Universidad Nacional de Lanús (Argentina).

Referencias

- Abney, S. P. (1991). Parsing by chunks (pp. 257-278). Springer Netherlands.
- Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007, January). Open information extraction for the web. In IJCAI (Vol. 7, pp. 2670-2676).
- Banko, Michele; Brill, Eric. Scaling to very very large corpora for natural language disambiguation (2001). En Proceedings of the 39th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2001. p. 26-33.
- Del Corro, L., & Gemulla, R. (2013, May). ClausIE: clause-based open information extraction. In Proceedings of the 22nd international conference on World Wide Web (pp. 355-366). International World Wide Web Conferences Steering Committee.
- Evan Sandhaus. The New York Times Annotated Corpus, 2008.
- Fader, A., Soderland, S., & Etzioni, O. (2011, July). Identifying relations for open information extraction. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 1535-1545). Association for Computational Linguistics.
- García-Martínez, R. & Britos, P. V. (2004). Ingeniería de sistemas expertos. Nueva Librería. ISBN 987-1104-15
- Gholap, J. (2012). Performance tuning of J48 Algorithm for prediction of soil fertility. arXiv preprint arXiv:1208.3943.
- Golding, Andrew R.; Roth, Dan. A winnow-based approach to context-sensitive spelling correction. Machine learning, 1999, vol. 34, no 1-3, p. 107-130.
- Gómez, A., Juristo, N., Montes, C., & Pazos, J. (1997). Ingeniería del conocimiento. Editorial Centro de Estudios Ramón Areces. ISBN 84-8004-269-9.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann P., Witten, I.H. (2009). The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.
- Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features (pp. 137-142). Springer Berlin Heidelberg.
- Kim, Sang-Bum, et al. Some effective techniques for naive bayes text classification. Knowledge and Data Engineering IEEE Transactions on, 2006, vol. 18, no 11, p. 1457-1466.
- Kolesov, Anton, et al. On Multilabel Classification Methods of Incompletely Labeled Biomedical Text Data. Computational and Mathematical Methods in Medicine, 2014, vol. 2014.
- Lewis, D. (1997). Reuters-21578 text categorization test collection, <http://goo.gl/NrOfu>, página veinte al 10/09/2016
- Lowd, Daniel; Domingos, Pedro. Naïve Bayes models for probability estimation. En Proceedings of the 22nd international conference on Machine learning. ACM, 2005. p. 529-536.

- Mccallumzy, Andrew; Nigamy, Kamal. A comparison of event models for Naïve bayes text classification. En AAAI-98 workshop on learning for text categorization. 1998. p. 41-48.
- NG, Hwee Tou; GOH, Wei Boon; LOW, Kok Leong. Feature selection, perceptron learning, and a usability case study for text categorization. En ACM SIGIR Forum. ACM, 1997. p. 67-73.
- Platt, J. (1998). Sequential minimal optimization: A fast algorithm for training support vector machines
- Quinlan, J. R. (2014). C4. 5: programs for machine learning. Elsevier.
- Ramshaw, L. A., & Marcus, M. P. (1995). Text chunking using transformation-based learning. arXiv preprint [cmp-lg/9505040](https://arxiv.org/abs/cmp-lg/9505040). Vigente al 10/09/2016.
- Rodríguez, J. M., Merlino, H., García-Martínez, R. (2015). Revisión Sistemática Comparativa de Evolución de Métodos de Extracción de Conocimiento para la Web. XXI Congreso Argentino de Ciencias de la Computación (CACIC 2015). Buenos Aires, Argentina.
- Rodríguez, J. M., Merlino, H. D., Pesado, P., & García-Martínez, R. (2016, August). Performance Evaluation of Knowledge Extraction Methods. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (pp. 16-22). Springer International Publishing.
- Schmitz, M., Bart, R., Soderland, S., & Etzioni, O. (2012, July). Open language learning for information extraction. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 523-534). Association for Computational Linguistics.
- Witten, I. H., Frank, E., Trigg, L., Hall, M., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

Estudio Comparativo de Algoritmos de Minería de Subgrafos Frecuentes

Santiago Bianco, Sebastian Martins, Ramón García-Martínez

Laboratorio de Investigación y Desarrollo en Ingeniería de Explotación de Información
Grupo de Investigación en Sistemas de Información. Universidad Nacional de Lanús
Remedios de Escalada, Buenos Aires, Argentina.
santiago.bianco.sb@gmail.com, smartins@gmail.com, rgm1960@yahoo.com

Resumen. Dentro las técnicas de minería de grafos se encuentran las correspondientes a búsqueda de subgrafos frecuentes. Existen varios algoritmos orientados a reconocer subestructuras comunes entre un conjunto de grafos entre los que destacan: FSG, FFMS, gSpan y GASTON. El objetivo de esta investigación es analizar el comportamiento de estos algoritmos a través de distintos experimentos diseñados para identificar si existe un algoritmo superior al resto y, en caso de que no lo haya, poder definir en qué escenarios es más recomendable la elección de cada uno.

Palabras clave. Minería de grafos, minería de subgrafos frecuentes, ingeniería de explotación de información, análisis de algoritmos.

1. Introducción

La búsqueda de subgrafos frecuentes se basa en encontrar estructuras recurrentes en un conjunto de grafos, es decir, buscar subgrafos que se repitan en una base de datos compuesta por grafos. El descubrimiento de estos patrones puede ser el propósito final del proceso o los subgrafos descubiertos pueden ser parte de otro proceso de clasificación [1]. Para determinar que un subgrafo es frecuente, tiene que superar determinado umbral, denotado como support. La definición formal es la siguiente: dado un conjunto de grafos GD y un umbral σ (o threshold en inglés), de manera que $0 < \sigma \leq 1$, el soporte de un grafo G , denotado como sup_G , es igual a la cantidad de grafos en GD en los cuales G es un sub-isomorfismo. Escrito como fórmula sería:

$$\text{Sup}_G = \frac{|\{G' \in GD \mid G \subseteq G'\}|}{|GD|}$$

Teniendo en cuenta esto, un grafo G es frecuente en una base de datos de grafos si $\text{sup}_G \geq \sigma$, siendo σ el soporte mínimo o minimum support. Por lo tanto, el problema de FSM se resume en, dado un umbral σ y un conjunto de grafos GD , encontrar todos los subgrafos frecuentes G en GD que cumplan con $\text{sup}_G \geq \sigma$.

Existen una variedad de algoritmos para realizar esta tarea, entre los que destacan: algoritmo FSG [6], algoritmo gSpan (graph-based Substructure pattern mining) [2],

lgoritmo FFSM (Fast Frequent Subgraph Mining) [3], y algoritmo GASTON (GrAph/SequencE/Tree extractiON) [7].

Algoritmo FSG (Frequent Sub-Graph) [6]: es el algoritmo más antiguo de los que se evalúan en este trabajo, por lo que introdujo algunas características que luego serían usados por el resto, que tienen que ver con la forma de representación de los grafos, la generación de subestructuras candidatas y detección de isomorfismos. En primer lugar, utiliza un tipo de representación para grafos dispersos que minimiza costos de procesamiento y almacenamiento. Este tipo de representación es usada para almacenar candidatos intermedios y los subgrafos que se van encontrando. Consiste en transformar las representaciones canónicas de los grafos, inicialmente representadas con matrices de adyacencia, para implementarlos como listas de adyacencia, las cuales disminuyen el uso de memoria y del procesador para estructuras dispersas. En segundo lugar, incrementa el tamaño de los subgrafos a buscar de a una arista por vez, permitiendo que la generación de candidatos sea más eficiente. Finalmente, usa algoritmos simples para implementar las representaciones canónicas de los grafos y las detecciones de isomorfismos que funcionan de manera eficiente para grafos chicos, e incorpora varias optimizaciones para el proceso de generación de candidatos y conteo (determinación de la frecuencia de un subgrafo) que permiten que el algoritmo sea escalable para grandes conjuntos de grafos.

Algoritmo gSpan (graph-based Substructure pattern mining) [2]: busca superar los inconvenientes que tienen los algoritmos que utilizan una estrategia apriorística como el FSG: el costo de la generación de candidatos y la detección de falsos positivos a la hora de la evaluación de isomorfismos. Como el problema de búsqueda de sub-isomorfismos está catalogado como NP-completo, es muy costoso computacionalmente volver a evaluar los resultados. Lo más destacado de este algoritmo es la implementación del recorrido en profundidad para reducir el espacio de búsqueda y la introducción de dos nuevas técnicas: los códigos DFS y DFSM para generar la representación canónica de los grafos. Otros aspectos destacables incluyen la eliminación de los procesos de generación de candidatos para el descubrimiento de subgrafos, así como también el recorte de falsos positivos. Además, combina los procedimientos de crecimiento y evaluación de subestructuras en uno solo, acelerando el proceso de búsqueda.

Algoritmo FFSM (Fast Frequent Subgraph Mining) [3]: utiliza el mismo enfoque de búsqueda en profundidad del gSpan, incorporando nuevas técnicas para mejorar su eficiencia como la utilización de la representación canónica CAM y nuevos métodos para agilizar el proceso de generación de candidatos mediante la manipulación de las matrices de adyacencia. Se introduce un procedimiento para garantizar que todos las subestructuras frecuentes sean enumerada unívocamente y sin ambigüedades (suboptimal CAM tree) y se evita el testeo de sub-isomorfismos, manteniendo una lista de cada subgrafo frecuente. Esta última es quizá la más relevante de todas las características, debido al gran potencial que puede llegar a tener el algoritmo al evitar ese procedimiento tan computacionalmente complejo.

Algoritmo GASTON (GrAph/SequencE/Tree extractiON) [7]: aprovecha un principio que llamaron ‘quickstart principle’, que considera siguiente hecho: los grafos, árboles y caminos están incluidos unos en otros, por lo que se puede dividir el proceso en distintos pasos de creciente complejidad, lo que simplifica el procedimiento general.

Primero, se buscan los caminos frecuentes, luego los árboles y finalmente los subgrafos frecuentes. Cada etapa tiene un proceso distinto para representar las estructuras en su forma canónica, debido a las distintas características que presentan dichas estructuras. Sin embargo, debido al quickstart principle antes mencionado, los códigos generados en una etapa pueden ser usados para la etapa siguiente, concatenando las nuevas ramificaciones encontradas en el caso de los árboles o los ciclos en el caso de los grafos. De esta manera, se reduce la complejidad del proceso. Para el proceso de conteo de grafos se utiliza un procedimiento similar al del algoritmo FFSM, con listas que almacenan las subestructuras ya analizadas. Debido a problemas de escalabilidad con este método, también proponen utilizar un proceso alternativo para grafos de gran tamaño, similar al utilizado en el algoritmo FSG. En este contexto, emerge el interés de analizar el estudio comparado del comportamiento de estos algoritmos. En la sección 2 se describe el problema, se describe en la sección 3 la formulación del diseño experimental con detalle de los experimentos con grafos sintéticos (sección 3.1) y experimentos con grafos reales (sección 3.2), en la sección 4 se presenta los resultados discriminando el análisis de las pruebas con grafos sintéticos (sección 4.1) de las realizadas con grafos reales (sección 4.2), en la sección 5 se enuncian las conclusiones preliminares y las futuras líneas de trabajo.

2. Descripción del Problema

Se han desarrollado varios algoritmos de minería de grafos en los últimos años [3; 12]. Los mismos, varían en la estrategia que utilizan para recorrer los grafos, el tipo de entrada que utilizan y la información de salida que proveen. Para dar soluciones más eficientes a los problemas, es de interés compararlos y determinar su comportamiento en distintos escenarios de manera que se pueda elegir la mejor opción en base a los datos con los que se cuente. En esta investigación se evaluarán los algoritmos FFSM, FSG, gSpan y GASTON, los cuales buscan estructuras considerando una frecuencia de ocurrencia mínima determinada y como resultado generan una lista con las subestructuras que cumplan con esa condición. Los mismos fueron elegidos ya que cada uno presenta novedosos enfoques y son los más reconocidos en el estado del arte. Si bien se han hecho comparaciones entre algoritmos [6; 7], las mismas sólo comparan el tiempo de ejecución de los algoritmos en base a distintas estructuras y definiendo cuál es el más rápido. Se considera de mayor interés ampliar las investigaciones determinando bajo qué circunstancias es más eficiente uno que otro, no solo en base al tiempo que demore sino también en la cantidad de patrones que pueda encontrar.

De lo anteriormente expuesto surgen la siguiente pregunta de investigación: ¿es posible definir distintos escenarios experimentales de manera que se pueda determinar bajo qué circunstancias es más eficiente la utilización de un algoritmo por sobre otro, de manera que se pueda facilitar la elección de los mismos dependiendo de las necesidades que se tengan y los datos disponibles?

3. Formulación del Diseño Experimental

En las siguientes secciones, se provee una descripción más detallada de los experimentos que se llevan a cabo y sus diferentes variantes: con grafos sintéticos y con grafos reales. Posteriormente, se establecen las variables dependientes e independientes involucradas en cada una de las pruebas a realizar tanto para los grafos sintéticos como para los grafos reales.

3.1. Experimentos con grafos sintéticos

Se prueban los algoritmos con datos generados aleatoriamente, variando la cantidad de nodos y arcos. Se utilizan conjuntos de cien grafos en todas las pruebas. Las pruebas se repiten cien veces y los resultados finales son un promedio de los resultados parciales. Este tipo de escenario a su vez contiene otros dos, que varían de acuerdo a cómo se incrementa la cantidad de arcos experimento a experimento. Estas pruebas se dividen en: (i) experimentos con incremento fijo, y (ii) experimentos con incremento variable.

Dentro de estos dos últimos hay otras variaciones que consisten en probar los algoritmos en conjuntos de grafos en los que todos sus nodos son diferentes, así como también en estructuras con repeticiones en las etiquetas de los elementos que los componen. Teniendo en cuenta esto, cada experimento a su vez incluye otros dos:

- Experimentos sin repetición de nodos: todos los vértices de los grafos contienen etiquetas distintas, de manera que cada elemento sea distinto del resto. Un ejemplo de un grafo con diez nodos y diez arcos que podría formar parte de este tipo de pruebas puede observarse en la figura 1. En todos los experimentos con grafos sintéticos las pruebas se realizan con grafos no dirigidos y cuyas aristas pueden tener tres tipos de etiquetas: “1”, “2” o “3”.
- Experimentos con repetición de nodos: en estos experimentos se ejecutaron los algoritmos con bases de datos en las cuales los vértices de los grafos contenían etiquetas repetidas, de manera que se puedan modelar situaciones en las cuales existan dos o varios objetos iguales dentro de la red. En estas pruebas, se utiliza una variable llamada nodos únicos que es igual a $|V|/2$, la cual determina la cantidad máxima de vértices con distintas etiquetas que puede tener un grafo. Por ejemplo, si la variable nodos únicos tiene un valor de 5, quiere decir que los nodos tendrán etiquetas desde el 0 al 4. Con esto se garantiza que haya elementos repetidos. Un ejemplo de un grafo con diez nodos, diez arcos y cinco nodos únicos que podría formar parte de este tipo de pruebas puede observarse en la Figura 2.

Se consideran las siguientes variables independientes para cada uno de estos procesos:

Cantidad de Nodos: Indica el total de nodos que tiene cada grafo del conjunto de prueba en cada experimento. Su valor es de 10 en la primera prueba y se incrementa a 100 en la última, con incrementos de a diez por cada cinco pruebas.

Cantidad de Arcos: Indica el total de arcos que tiene cada grafo del conjunto de prueba en cada experimento. Difiere dependiendo del tipo de experimento. Para aquellos con incremento fijo, su valor es de 10 en la primera prueba y se incrementa hasta 120 en la última, con incrementos de a cinco y reajustando su

valor por cada cinco pruebas. Para aquellos con incremento variable, comienza en 10 y se incrementa hasta 300. Los incrementos se hacen según la fórmula (*Cantidad de Nodos / 2*).

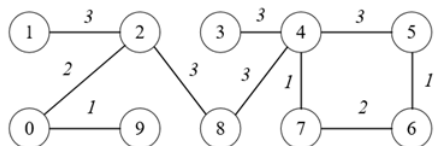


Fig. 1. Ejemplo de grafo, Pruebas con grafos sintéticos sin repetición de nodos

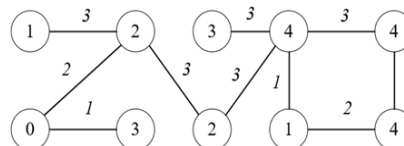


Fig. 2. Ejemplo de grafo, Pruebas con grafos sintéticos con repetición de nodos.

Nodos Únicos: Indica la cantidad máxima de nodos sin repetirse que puede haber en cada grafo. Es utilizada para las pruebas con repeticiones de nodos. Su valor se corresponde con la fórmula (*Cantidad de Nodos / 2*), garantizando que siempre haya nodos con etiquetas repetidas.

Cantidad de grafos: Indica la cantidad de grafos que tendrán los datos generados para las pruebas. Su valor para todos los experimentos de 100.

Minimum Support Threshold o Umbral de Frecuencia Mínimo: Indica la cantidad mínima de grafos que deben contener a una subestructura para considerarla de frecuente. En estos experimentos el valor se fija en 5% para todos los algoritmos.

A su vez, se busca comparar las siguientes variables dependientes para poder evaluar el comportamiento de los algoritmos en cada experimento:

Cantidad de subestructuras: Indica la cantidad de subestructuras frecuentes que los algoritmos encuentran en cada experimento. Con esto se busca identificar en qué situaciones un algoritmo puede llegar a ser más útil que el resto.

Tiempo de ejecución: Indica cuánto demora cada algoritmo en conseguir resultados. Esta variable será medida en segundos.

Cotejando ambas variables en conjunto se busca analizar el comportamiento global de cada algoritmo según las características del conjunto de datos en los que se apliquen.

3.2. Experimentos con grafos reales

En estas pruebas se ejecutan los algoritmos sobre un dataset real Compounds_422 distribuido con la implementación de gSpan. El mismo es utilizado por varios autores para testear el rendimiento de sus algoritmos [5, 7]. Sus características pueden verse en la Tabla 1. Sobre esta base de datos se ejecutan pruebas cien veces y se promedian los resultados.

Para estas pruebas se considera únicamente a la siguiente variable independiente:

Minimum Support Threshold o Umbral de Frecuencia Mínimo: Indica la cantidad mínima de grafos que deben contener a una subestructura para considerarla de frecuente. En estos experimentos el valor se fija en 25% para la primera prueba y aumenta 5% por cada prueba hasta llegar a 95%.

Tabla 1. Características del archivo a utilizar en las pruebas con grafos reales.

Compounds 422: Archivo de prueba con estructuras moleculares	
Cantidad de grafos	422
Cantidad de etiquetas de arcos distintas	4
Cantidad de etiquetas de nodos distintas	21
Promedio de arcos por grafo	42
Promedio de nodos por grafo	40
Cantidad máxima de arcos por grafo	196
Cantidad máxima de nodos por grafo	189

Además, Se busca comparar las siguientes variables dependientes para poder evaluar el comportamiento de los algoritmos en cada experimento:

Cantidad de subestructuras: Indica la cantidad de subestructuras frecuentes que los algoritmos encuentran en cada experimento.

Tiempo de ejecución: Indica cuánto demora cada algoritmo en conseguir resultados. Esta variable será medida en segundos.

Cotejando ambas variables en conjunto se busca analizar el comportamiento global de cada algoritmo en un conjunto de datos reales, incrementando en cada paso el umbral mínimo.

4. Resultados

A continuación se realiza un análisis de los resultados obtenidos por cada tipo de prueba y se presentan las observaciones desprendidas de cada uno.

4.1. Análisis de las pruebas con grafos sintéticos

Para el análisis de los resultados obtenidos a partir de las pruebas con nodos únicos, se unificaron los experimentos con incremento fijo y variable para así realizar una comparación global, generando un total de 82 distintos escenarios al combinar los dos tipos de prueba. En estos escenarios se varía el promedio de nodos y arcos del conjunto de grafos en el siguiente orden (teniendo en cuenta <aristas promedio, arcos promedio>):

(10,10), (10,15), (10,20), (10,25), (10,30), (20,20), (20,25), (20,30), (20,35), (20,40), (20,50), (20,30), (30,30), (30,35), (30,40), (30,45), (30,50), (30,60), (30,75), (30,90), (40,40), (40,45), (40,50), (40,55), (40,60), (40,80), (40,100), (40,120), (50,50), (50,55), (50,60), (50,65), (50,70), (50,75), (50,100), (50,125), (50,150), (60,60), (60,65), (60,70), (60,75), (60,80), (60,90), (60,120), (60,150), (60,180), (70,70), (70,75), (70,80), (70,85), (70,90), (70,105), (70,140), (70,175), (70,210), (80,80), (80,85), (80,90), (80,95), (80,100), (80,120), (80,160), (80,200), (80,240), (90,90), (90,95), (90,100), (90,105), (90,110), (90,135), (90,180), (90,225), (90,270), (100,100), (100,105), (100,110), (100,115), (100,120), (100,150), (100,200), (100,250), (100,300)

En la figura 3 se muestra el gráfico comparativo. En el eje de abscisas se presentan los resultados expresados en estructuras/milisegundos. De esta manera, puede observarse el desempeño general de cada algoritmo a medida que se incrementa el tamaño de las bases de datos, y como son afectados por la variación en la densidad de los grafos que las componen. Analizando el gráfico puede notarse que el algoritmo GASTON es superior al resto para los primeros experimentos, en los que las bases son relativamente chicas. Sin embargo, su rendimiento decae abruptamente en las pruebas

11 y 12, en las cuales se incrementa la cantidad de aristas. En estos escenarios, la implementación directamente no es capaz de encontrar ninguna subestructura y cesa su ejecución, mientras que el resto de los algoritmos sí lo hace. Este comportamiento es recurrente cada vez que se aumenta la densidad de los grafos (pruebas 18, 25, 35, 42, 52, 60, 70 y 78). En el resto de las pruebas, cuando es capaz de encontrar estructuras, sus resultados son mejores que los que producen los otros algoritmos. Los otros dos algoritmos, el gSpan y el FSG son más lentos pero son capaces de encontrar resultados en todos los escenarios, incluso en las últimas pruebas. También se puede observar que sus comportamientos son más lineales que los del resto.

Comparando al gSpan y FSG, éste último obtiene mejores resultados en gran parte de las pruebas, pero esta tendencia se revierte en los experimentos finales, a partir del número 48. Desde allí, el gSpan es ligeramente superior, y luego de la prueba 78 el gSpan supera no sólo al FSG sino a los demás algoritmos también.

En el caso de las pruebas con repetición de nodos, para el análisis de los algoritmos también se combinarán los resultados obtenidos de las pruebas con incremento fijo e incremento variable al igual que en los experimentos anteriores, siempre recordando que la cantidad máxima de etiquetas distintas va a ser igual a la mitad de la cantidad de vértices ($|V|/2$). El gráfico resumen de los resultados obtenidos se puede observar en la figura 4.

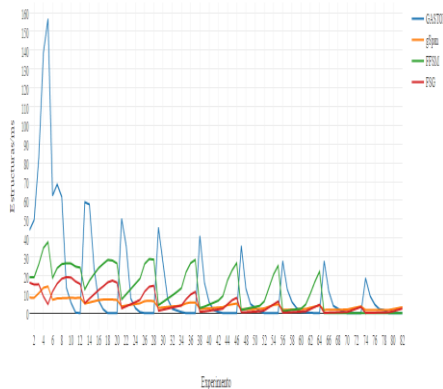


Fig. 3. Gráfico resumen de los resultados de los experimentos con grafos sintéticos con nodos únicos

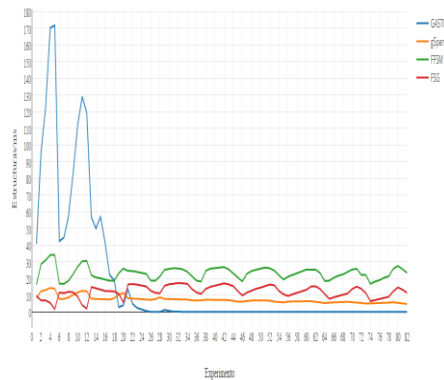


Fig. 4. Gráfico resumen de los resultados de los experimentos con grafos sintéticos con nodos repetidos.

Analizando el gráfico generado a partir de los resultados, a simple vista se puede observar un cambio en el comportamiento de los algoritmos con respecto a los presentados en las pruebas anteriores. El GASTON sigue siendo muy superior para las primeras pruebas pero hasta la prueba 17, luego su rendimiento decae y a partir de la prueba 24 no es capaz de encontrar subestructuras, exceptuando la prueba 29. Se siguen observando dificultades a medida que aumenta la densidad de la base de datos, aunque ahora este problema se ve más acentuado. El resto de los algoritmos se comporta de manera más constante que en las pruebas anteriores y todos son capaces de encontrar resultados hasta el final, siendo el FFSM el que encuentra más resultados en menor tiempo de comienzo a fin. En las primeras pruebas el gSpan y el FSG arrojan

resultados cambiantes hasta la prueba 20, en la cual ambos se estabilizan y el FSG consigue tener un mejor rendimiento.

4.2. Análisis de las pruebas con grafos reales

Mirando la conformación de la base (Tabla 1) puede notarse que la estructura de los grafos es muy variada, teniendo en cuenta que el grafo con mayor cantidad de vértices posee 189 y el promedio es de 40. Considerando sólo los promedios y haciendo una analogía con las pruebas con grafos sintéticos, esta prueba podría clasificarse dentro de los experimentos con nodos repetidos, aproximadamente entre las pruebas 13 y 21 analizadas anteriormente. En la figura 5 se presentan los resultados obtenidos, contrastando el minimum support de cada prueba con la cantidad de estructuras encontradas por milisegundo. A simple vista puede verse como el algoritmo GASTON supera ampliamente al resto de los algoritmos en las pruebas. Esto se debe a que las estructuras que conforman la base de datos no son tan densas y tal como ocurrió en las pruebas con grafos sintéticos, es estos escenarios el GASTON es el que se comporta de manera más eficiente. Para analizar lo ocurrido con el resto de los algoritmos con más detalle se presenta la figura 6. En la misma se observa que el algoritmo gSpan es capaz de encontrar resultados en todas las pruebas y más eficientemente que el FSG. Este último deja de encontrar resultados cuando el support es de 95%. El algoritmo FFSM, a pesar de haber conseguido obtener más resultados por unidad de tiempo que el resto con excepción del GASTON en las pruebas con grafos sintéticos, no pudo encontrar resultados con esta base de datos. Tal vez se deba al tamaño de la base en relación al support utilizado ya que en las pruebas con grafos sintéticos, el umbral mínimo era del 5% y las bases estaban compuestas por 100 grafos, en lugar de los 400 que se utilizaron ahora. Hay menor probabilidad de hallar grafos frecuentes con un support elevado a medida que aumenta el tamaño de la base.

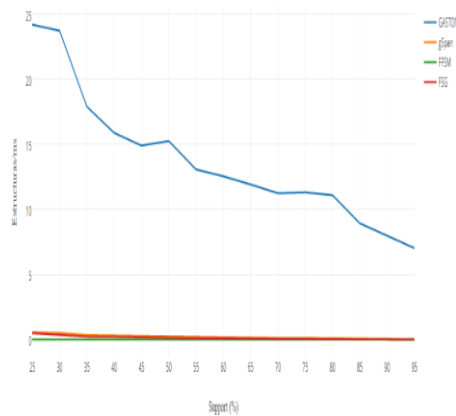


Fig. 5. Análisis de los resultados para las pruebas con grafos reales, contrastando el minimum support porcentual contra las estructuras encontradas.

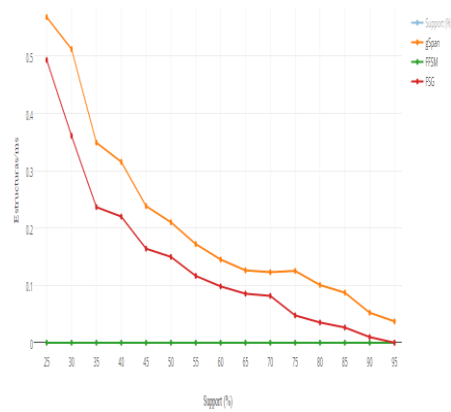


Fig. 6. Detalle de resultados para las pruebas con grafos reales de los algoritmos gSpan, FFSM y FSG.

5. Conclusiones Preliminares y Futuras Lineas de Trabajo

De lo expuesto en las secciones anteriores se pueden derivar las siguientes conclusiones con respecto a los experimentos realizados:

- La conformación de la base altera el rendimiento de los algoritmos. No es lo mismo tener estructuras con elementos únicos, como una red social, a tener estructuras con elementos repetidos, como puede ser una molécula. El tamaño de los grafos y la densidad de los mismos también afecta al comportamiento de todos los algoritmos.
- El algoritmo GASTON encuentra más resultados en menor tiempo que el resto para una base de datos compuesta por grafos chicos, de hasta aproximadamente 30 vértices y 60 aristas, pero se ve fuertemente afectado al aumentarse la densidad de la base y cuando se tienen muchos nodos repetidos.
- El algoritmo FFSM es el segundo algoritmo que más estructuras encuentra por segundo en las primeras pruebas, y el más eficiente luego de que disminuye el rendimiento del GASTON. Su rendimiento es mejor para las pruebas con grafos repetidos, ya que siempre es capaz de encontrar resultados. En los otros experimentos hay varias pruebas en la que no puede (56 y de la 66 en adelante).
- El comportamiento de los algoritmos gSpan y FSG son muy similares en las pruebas con nodos únicos, siendo el gSpan superior hacia las pruebas finales. En el caso de los experimentos con vértices repetidos, el FSG es mejor a partir de la prueba 20 hasta el final.

En resumen, se podría decir que el algoritmo GASTON es el más apropiado cuando las bases de datos no son muy densas (entre 30 vértices y 60 aristas en promedio aproximadamente) y no poseen mucha complejidad. Si las estructuras son complejas pero con un tamaño moderado (hasta 70 vértices y 90 aristas en promedio aproximadamente), el algoritmo FFSM es la mejor opción, sobre todo si los grafos cuentan con etiquetas repetidas. En el caso de grandes estructuras (más de 80 vértices y 100 aristas en promedio), las mejores opciones son el FSG y el gSpan, siendo este último el único que pudo encontrar subestructuras en todos los experimentos, por lo que se lo considera el más estable.

De las pruebas con grafos reales se pudo verificar también el hecho de que los algoritmos siempre encontraron la misma cantidad de subestructuras en cada iteración de las pruebas, por lo que no es necesario ejecutar los algoritmos reiteradas veces para tener resultados confiables una vez que se hayan definido los parámetros a utilizar.

En cuanto al trabajo futuro, la minería de grafos es un campo de constante crecimiento en la cual podrían desarrollarse las siguientes líneas de investigación:

- Ampliar los algoritmos a evaluar incluyendo otros como el SPIN [12] o CloseGraph para verificar su comportamiento.
- Ampliar las pruebas con grafos reales de manera similar a lo realizado con grafos sintéticos: usar estructuras más simples y más complejas a la base de datos utilizada en esta investigación y comparar los resultados.
- Desarrollar otras pruebas que involucren conjuntos de datos distintos en cuanto al tamaño o la conformación de las bases de datos. Podrían utilizarse grafos dirigidos o en el caso de que se usen bases de datos reales, podrían utilizarse otro tipo de estructuras que no sean moléculas.

Financiamiento

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por el Proyecto de Investigación 33A205 de la Secretaria de Ciencia y Técnica de la Universidad Nacional de Lanús (Argentina).

Referencias

- [1] García-Martínez, R., Britos, P., Pesado, P., Bertone, R., Pollo-Cattaneo, F., Rodríguez, D., Pytel, P., Vanrell, J. 2011. Towards an Information Mining Engineering. En Software Engineering, Methods, Modeling and Teaching. Sello Editorial Universidad de Medellín. ISBN 978-958-8692-32-6. Páginas 83-99.
- [2] Yan, X., & Han, J. (2002). gspan: Graph-based substructure pattern mining. In Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on (pp. 721-724). IEEE.
- [3] Huan, J., Wang, W., & Prins, J. (2003, November). Efficient mining of frequent subgraphs in the presence of isomorphism. In Data Mining, 2003. ICDM 2003. Third IEEE International Conference on (pp. 549-552). IEEE.
- [4] Holder, L. B., Cook, D. J., & Djoko, S. (1994, July). Substructure Discovery in the SUBDUE System. In KDD workshop (pp. 169-180).
- [5] Ullman, J. R. (1976). An algorithm for subgraph isomorphism. Journal of the ACM (JACM), 23(1), 31-42.
- [6] Kuramochi, M., & Karypis, G. (2001). Frequent subgraph discovery. In Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on (pp. 313-320). IEEE.
- [7] Nijssen, S., & Kok, J. N. (2004, August). A quickstart in frequent structure mining can make a difference. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 647-652). ACM.
- [8] Zaiat, J., & Romero-Zaliz, R. Minería de datos sobre grafos: un enfoque multiobjetivo aplicado a bioremediación.
- [9] Lahiri, M., & Berger-Wolf, T. Y. (2007, March). Structure prediction in temporal networks using frequent subgraphs. In Computational Intelligence and Data Mining, 2007. CIDM 2007. IEEE Symposium on (pp. 35-42). IEEE.
- [10] Takigawa, I., & Mamitsuka, H. (2013). Graph mining: procedure, application to drug discovery and recent advances. Drug discovery today, 18(1), 50-57.
- [11] Rehman, S. U., Khan, A. U., & Fong, S. (2012, August). Graph mining: A survey of graph mining techniques. In Digital Information Management (ICDIM), 2012 Seventh International Conference on (pp. 88-92). IEEE.
- [12] Huan, J., Wang, W., Prins, J., & Yang, J. (2004, August). Spin: mining maximal frequent subgraphs from graph databases. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 581-586). ACM.

Algoritmo de Clustering Basado en el Concepto de Densidad Atómica

Oscar Andrés Monroy Medina¹,

¹ Facultad de Ingeniería, Universidad Austral,
Cerrito 1250, Ciudad Autónoma de Buenos Aires, Argentina
{oscar.andres.monroy}@gmail.com

Resumen. El análisis de clúster o Clustering agrupa un conjunto de objetos de datos en clústers o grupos de manera que en cada grupo los objetos sean similares entre si y disimiles de los objetos de otros grupos. En la actualidad, existen distintas técnicas de agrupamiento que permiten cumplir con esta tarea. En búsqueda de un algoritmo más natural se hizo uso del concepto de densidad atómica de los elementos como base para generar uno nuevo. El algoritmo propuesto tiene como ventajas, poseer un método concreto de selección de centroides, además de tener mejores agrupamientos que otros algoritmos basados en centroides como k-means y k-medoids.

Palabras Clave: Data Mining – Clúster – Clustering - Densidad Atómica - Matriz de distancia

1 Introducción

En algunos problemas de data mining llega a ser necesario agrupar *dataset*¹ en diferentes grupos o clases para determinar una acción a ejecutar o para optimizar un algoritmo de clasificación o predicción. El proceso de particionar un conjunto de datos en un conjunto de sub-clases (o grupos) significativas se llama Clustering [1].

En la actualidad existen diferentes algoritmos de clustering que dependiendo de la orientación pueden dar un resultado muy diferente de otro [1, 4, 5, 8, 9], sin embargo, no se puede decir que los clústers resultantes dan una respuesta “errónea”, simplemente un algoritmo de clustering se adapta mejor que otro para un problema determinado, esto da a lugar a la posibilidad de crear nuevos algoritmos para determinados problemas.

En búsqueda de un algoritmo más natural se ha establecido una agrupación basada de un estudio que lleva más de un siglo de investigación, la tabla periódica de los elementos. Una de las propiedades que tienen los elementos en la tabla periódica es la densidad atómica que posee diversas aplicaciones como se puede ver [2, 3].

¹ Un *dataset* está definido como un conjunto de datos, termino usualmente usado en minería de datos.

Observando las distintas aplicaciones que posee la densidad atómica, se podría usar el cálculo definido por la teoría [13] aplicado a un espacio n -dimensional (donde n es la cantidad de variables del *dataset*) para agrupar los puntos en clústers donde la densidad sea la máxima.

Para definir la fórmula de la densidad sobre un *dataset* de n variables se usará la definición inicial dada por [13].

$$d = \frac{m}{v} \quad (1)$$

Donde m es la masa y v el volumen de la hiperesfera definido en [14]. Para aplicarlo a un *dataset* dado se definirá la masa y el volumen como:

$$m_c(r) = |\{x|x \in D \wedge \|c - x\| \leq r\}| \quad (2)$$

$$V_n(r) = \frac{\pi^{\frac{n}{2}}}{\Gamma(\frac{n}{2} + 1)} r^n \quad (3)$$

Donde $\| \cdot \|$ es la distancia euclidiana, r es el radio de la hiperesfera, c el centro, n el número de dimensiones y D el *dataset*. Según [13] esta fórmula de volumen es influenciada fuertemente por el número de dimensiones, es decir que en un *dataset* con una gran cantidad de variables el volumen tiende a 0. Es por eso que se utilizará el volumen del hipercubo que contiene la hiperesfera para el cálculo de la dimensión.

$$V_n(r) = (2r)^n \quad (4)$$

Luego la fórmula de la densidad para los *dataset* estaría definida como:

$$d_n(c; r) = \frac{m_c(r)}{(2r)^n} \quad (5)$$

2 El conjunto de datos y el algoritmo.

2.1 Extracción del dataset

Para el estudio de un *dataset* real se obtuvo la información de atletas de crossfit y sus resultados para la clasificación de los regionales (el open) de los años 2011, 2012, 2013 y 2014, esta información se encuentra disponible en <http://games.crossfit.com/leaderboard>. Para esta información se usó un *scraper*² llamado scrapy y se guardaron los datos en una base de datos noSQL llamada mongoDB. Una vez obtenidos los datos se unificaron en un csv. El resultado luego de un proceso de extracción, transformación y limpieza(ETL) da un *dataset* de 6153 elementos y 34 variables. Una descripción de las variables se puede ver en la Tabla 1.

Tabla 1. Descripción del *dataset*.

Variable	Tipo	Descripción
Year	Entero	Año de competencia
X	Decimal	Coordenada del lugar de entrenamiento
Y	Decimal	Coordenada del lugar de entrenamiento
How long have you been doing crossfit	Entero	Meses de entrenamiento
Weight	Decimal	Peso del atleta
Clean and jerk	Decimal	Máximo peso que el atleta puede levantar en clean&jerk
filthy 50	Entero	Tiempo en segundos del conjunto de ejercicios filthy 50
I eat whatever is convenient	Binario	1 verdadero /0 falso relativo a la dieta del atleta
I weigh and measure my food	Binario	1 verdadero /0 falso relativo a la dieta del atleta
I eat strict paleo	Binario	1 verdadero /0 falso relativo a la dieta del atleta
I eat 13 full cheat meals per week	Binario	1 verdadero /0 falso relativo a la dieta del atleta
Sprint 400m	Entero	Tiempo que tarda el atleta para correr 400 metros
Back squat	Decimal	Máximo peso que el atleta puede levantar en back squat
Fran	Entero	Tiempo en segundos del conjunto de ejercicios fran
Fight gone bad	Entero	Tiempo en segundos del conjunto de ejercicios fight gone bad
Snatch	Decimal	Máximo peso que el atleta puede levantar en snatch
Max pull ups	Entero	Máxima cantidad de repeticiones seguidas que el atleta puede hacer del ejercicio pull up
Helen	Entero	Tiempo en segundos del conjunto de ejercicios helen
Grace	Entero	Tiempo en segundos del conjunto de ejercicios grace
Gender	Binario	Genero del atleta 0 mujer/1 hombre
Age	Entero	Edad
Deadlift	Decimal	Máximo peso que el atleta puede levantar en deadlift
I have completed the crossfit level 1 certificate	Binario	1 verdadero /0 falso relativo a la formación del atleta
I have attended one or more specialty courses	Binario	1 verdadero /0 falso relativo a la formación del atleta
I have had a life changing experience due to crossfit	Binario	1 verdadero /0 falso relativo a la formación del atleta
I train other people	Binario	1 verdadero /0 falso relativo a la formación del atleta
I began crossfit with a coach eg at an affiliate	Binario	1 verdadero /0 falso relativo a la formación del atleta
I strictly Schedule my rest days	Binario	1 verdadero /0 falso relativo a la formación del atleta
I do multiple workouts in a day 2x a week	Binario	1 verdadero /0 falso relativo a la formación del atleta

² Un *scraper* es una herramienta utilizada para obtener datos de una fuente externa, como puede ser una página web.

I usually only do 1 workout a day	Binario	1 verdadero /0 falso relativo a la formación del atleta
I typically rest 4 or more days per month	Binario	1 verdadero /0 falso relativo a la formación del atleta
I typically rest fewer than 4 days per month	Binario	1 verdadero /0 falso relativo a la formación del atleta
I do multiple workouts in a day 1x a week	Binario	1 verdadero /0 falso relativo a la formación del atleta
Rank	Entero	Posición que obtuvo el atleta para esta competencia

2.2 Desarrollo del Algoritmo

Antes de escribir el algoritmo es necesario definir un conjunto más.

$$Den_n(D) = \{d_n(c; r) | c \in D \wedge r \in R^+\} \quad (6)$$

En otras palabras (6) sería el conjunto de todas las densidades posibles para el conjunto D . Una descripción del algoritmo de clustering sería el siguiente:

Procedimiento

```

P:=D;
C:={};
mientras (|P|>0)
    Encontrar c y r tal que  $d_n(c; r) = \max(Den_n(P))$ ;
    Cluster:={x ∈ P ||x - c|| ≤ r};
    C:=C union {Cluster};
    P:=P-Cluster;
fin mientras
retornar C
fin

```

2.3 La matriz de distancias

El algoritmo descrito en 2.1 parece algo sencillo pero el cálculo de $Den_n(D)$ para cada $r \in R^+$ es algo imposible de calcular. Por eso es necesario limitar los valores de r a un conjunto finito de valores. Se dirá entonces, que todos los posibles valores que puede tomar r serían todas las distancias entre los puntos de D , es decir todas las distancias que hacen parte de la matriz de distancias de D . Es decir que para un elemento $c \in D$ los posibles valores de r serían los valores del vector de distancias de c , una descripción de este conjunto se puede ver en (7).

$$M_D(c) = \{r | r = \|x - c\|, \forall x \in D\} \quad (7)$$

$$M_D = \cup_{c \in D} M_D(c) \quad (8)$$

De esta manera se puede limitar los valores de (6) para replantearlo de la siguiente manera:

$$Den_n(D) = \{d_n(c; r) | c \in D \wedge r \in M_D(c)\} \quad (9)$$

Ahora $Den_n(D)$ se convierte en un conjunto finito. Sin embargo, esta definición deja un problema alto de complejidad, dado que para hallar M_D se tiene que realizar un total de $|D|^2$ calculos. En el caso particular de este *dataset* el valor sería de 6153^2 , es decir 37'859.409 calculos de distancia. Por supuesto, aplicando algunas propiedades de distancias como (10) y (11) se pueden reducir los cálculos a menos de la mitad.

$$\forall x, y \quad \|x - y\| = \|y - x\| \quad (10)$$

$$\|x - x\| = 0 \quad (11)$$

El total de cálculos se reduciría a (12) que sería el total de cálculos, restando los cálculos que dan 0 y los cálculos que se repetirían.

$$\frac{|D|^2 - |D|}{2} = 18'926.628 \quad (12)$$

Se reducen los cálculos en gran cantidad, sin embargo, aún sigue siendo demasiado costoso. Es por eso que sería necesario realizar un algoritmo tomando muestras representativas y evitando así tener que calcular toda la matriz de distancia.

2.4 Algoritmo por muestras aleatorias

El algoritmo aplicado a muestras aleatorias sería muy similar al anterior

Procedimiento

```

P:=D;
C:={};
mientras (|P|>0)
    si n < |P| entonces:
        S:=muestra aleatoria de tamaño n;
    Si no:
        S:=P;
    Encontrar c y r tal que  $d_n(c; r) = \max(Den_n(S))$ ;
    Cluster:={x ∈ P ||x - c|| ≤ r};
    C:=C union {Cluster};
    P:=P-Cluster;
fin mientras
retornar C
fin

```

3 Los resultados

Finalmente, los dos algoritmos dieron sus resultados. Para la comparación se hará uso de la métrica SSE (Error de suma de cuadrados) la cual se puede aplicar a clusters que son basados en centroides. Además de comparar los dos algoritmos (el algoritmo que usa el *dataset* total y el algoritmo que usa muestras aleatorias) se compararan también con el mismo algoritmo pero tomando el centroide aleatoriamente, y con los algoritmos k-means y k-medoids implementados en rapid miner (<https://rapidminer.com/>).

3.1 Calculo del SSE

El SSE representa que tan próximos están los elementos de un clúster a su centroide. Clústers bien agrupados tienden a tener un SSE más bajo. El cálculo del SSE en un conjunto de clúster sería la descrita en (13).

$$SSE(C) = \sum_{C_x \in C} \sum_{y \in C_x} \|x - y\| \quad (13)$$

Donde C es el conjunto de todos los clusters y C_x representa un cluster con centro en x .

3.2 Comparación de los SSE

Finalmente se compararon los resultados de SSE entre los algoritmos de densidad atómica usando todo el *dataset*, de densidad atómica usando muestras aleatorias, densidad atómica tomando centroide aleatorio, k-means y k-medoids. Para k-means y k-medoids se realizaron estudios con k igual a el número de clusters resultantes del algoritmo de densidad atómica usando todo el *dataset* y con un máximo número de iteraciones de 100. Los resultados se ven resumidos en la tabla 2.

Tabla 2. Comparación de los valores SSE para los distintos métodos de clustering.

Método Utilizado	SSE	Diferencia %
Densidad Atómica Total	26091428958,495	0
Densidad Atómica 10%	26357521896,261	1.02
Densidad Atómica aleatoria	146454233565,81	461,3116622
k-means	$6.4829711950049 * 10^{18}$	24847129604
k-medoid	$5,7617455041804 * 10^{25}$	$2,20829 * 10^{17}$

Como se puede ver en la tabla 1, las diferencias entre los valores de los SSE son importantes, al usar conjuntos aleatorios de 10% de tamaño del *dataset* total da una diferencia del 1.02% de pérdida de SSE con respecto al algoritmo que usa el total del conjunto. Mientras que hay una diferencia significativa si se toma como centroide un elemento aleatorio, perdiendo más del 461% del SSE posible, estos algoritmos a pesar

de perder grandes porcentajes de SSE siguen siendo mucho mejores que k-means y k-medoid generados con el mismo número de clusters.

4 Conclusiones

Como conclusión se puede decir que el algoritmo de clustering basado en la densidad atómica puede obtener excelentes resultados, sin embargo, existe un costo computacional importante ya que depende del cálculo de todas las distancias del centroide. Por otro lado, al igual que k-means la selección de centroides es muy importante, como se pudo ver en los resultados, se obtuvo una baja de más del 461% en el cálculo del SSE al tomar los centroides de forma aleatoria. Finalmente, una salida más viable podría ser la del uso de muestras aleatorias representativas, como por ejemplo la del 10%, que solo tuvo una baja de un 1.02% en el valor del SSE y se reducen los cálculos necesarios para hallar los centroides y clusters ya que no es necesario calcular la matriz de distancias por completo.

Referencias

1. A.K. Jain, M.N. Murty and P.J. Flynn: Data Clustering: A review. 16 March 2000.
2. P. J. Desréa: Homogeneous crystalline nucleation via atomic density fluctuations in the liquid state: Applications. Philosophical Magazine Letters 1994 vol 69, No. 5, 261-268(1994)
3. I.M. Sokolov, D.V. Kupriyanov, R.G. Olave, M.D. Havey: Light trapping in high-density ultracold atomic gases for quantum memory applications. 10 May 2010.
4. Daniel Fasulo: An Analysis of the Recent Works on Clustering Algorithms. 26 Apr 1999.
5. S.B. Kotsiantis, P. E. Pintelas: Recents Advances in Clustering: A brief Survey. 21 Jan 2004.
6. Toni Giorgino: Computing 1-D atomic densities in macromolecular simulations: the Density Profile Tool for VMD. 27 Aug 2013.
7. Romit Chakraborty and David A. Mazziotti: Generalized Pauli Conditions on the Spectra of One-electron Reduced Density. 21 Apr 2014.
8. Chunfei Zhang and Zhiyi Fang: An Improved K-means Clustering Algorithm. 1 Jan 2013.
9. Amandeep Kaur Mann and Navneet Kaur: Survey Paper on Clustering Techniques. Apr 2013.
10. Sutapat Thiprungsri: Cluster Analysis for Anomaly Detection in Accounting Data. 31 Jul 2010.
11. Qi Liu and Miklos Vasaherlyi: Healthcare fraud detection: A survey and a clustering model incorporating Geo-location information. 21 Nov 2013.
12. Clifton Phua, Vincent Lee, Kate Smith and Ross Gayler: A Comprehensive Survey of Data Mining-based Fraud Detection Research. 7 Nov 2007.
13. International Union of Pure and Applied Chemistry: Compendium of Chemical Terminology – Gold Book. 24 Feb 2014.
14. Bruce Cohen and David Sklar: The Gamma Function, Factorials and the Volumes of n-Balls. 1 Dec 2001.
15. Greg Glassman: Understanding Crossfit. The crossfit journal, Greg Glassman. Apr 2007.
16. Mario Köppen: The Curse of Dimensionality. Sep 2000.
17. Michel Deza and Elena Deza: Encyclopedia of Distances. 23 Oct 2012.

Wild Cetacean Identification using Image Metadata

Débora Pollicelli^{1,2}, Mariano Coscarella^{1,3}, and Claudio Delrieux⁴

¹ CESIMAR-CONICET, Centro para el Estudio de Sistemas Marinos, Consejo Nacional de Investigaciones Científicas y Técnicas, CCT CENPAT, Bv. Almirante Brown 2915, 9120, Puerto Madryn, Chubut, Argentina

² LINVI, Departamento de Informática, Facultad de Ingeniería, UNPSJB, Bv. Almirante Brown 3051, 9120, Puerto Madryn, Chubut, Argentina

³ Departamento de Biología General, Facultad de Ciencias Naturales, UNPSJB, Bv. Almirante Brown 3051, 9120, Puerto Madryn, Chubut, Argentina

⁴ Laboratorio de Ciencias de las Imágenes, Departamento de Ingeniería Eléctrica y Computadoras, CONICET-UNS, 8000 Bahía Blanca, Argentina
deborapollicelli.mcoscarella@gmail.com, cad@uns.edu.ar

Abstract. Identification of individuals in marine species, especially in Cetacea, is a critical task in several biological and ecological endeavours. Most of the times this is performed through human-assisted matching within a set of pictures taken in different campaigns during several years and spread around wide geographical regions. This requires that the scientists perform laborious tasks in searching through archives of images, demanding a significant cognitive burden which may be prone to intra and inter observer operational errors. On the other hand, additional available information, in particular the metadata associated to every image, is not fully taken advantage of. The present work presents the result of applying machine learning techniques over the metadata of archives of images as an aid in the process of manual identification. The method was tested on a database containing several pictures of 230 different Commerson's dolphins (*Cephalorhynchus commersoni*) taken over a span of seven years. A supervised classifier trained with identifications made by the researchers was able to identify correctly above 90% of the individuals on the test set using only the metadata present in the image files. This reduces significantly the number of images to be manually compared, and therefore the time and errors associated with the assisted identification process.

Keywords: machine learning, photo-identification, cetaceans, Commerson's dolphins

1 Marine Mammal Individual Identification

In Biology, Ecology, and other sciences, the ability to recognize individuals allows the researchers to obtain relevant information that is crucial for several scientific purposes, including population parameters estimation such as size, fecundity, survival and mortality rates, home ranges and movements, etc. [11] [15]. These parameters are usually derived or inferred from the implementation of capture-recapture models. Capture-recapture models are based on the possibility of identifying a specific animal

from one sampling occasion to another, considering the first time the animal was photographically registered as a “capture” and the subsequent times as “recaptures” [5]. Since the 1970s, researchers relied on natural marks or other visual features to identify animals with non-invasive means. This picture-based identification technique was developed for cetaceans or other large marine fauna, mainly because handling other recognition means (f.e., attaching straps or belts to the individuals) is expensive, difficult and invasive, being impractical as an identification mean in the field. On the other hand, taking pictures (captures or recaptures) is relatively inexpensive and less difficult, providing reliable information on which were the individuals present at a given place and time, with the obvious disadvantage of depending on further recaptures of the individual and a proper identification in the picture archives.

Individual recognition of cetaceans in pictures is usually performed using different features. For example, southern right whales (*Eubalaena australis*) may be identified using the callosities patterns located in the upper part of the whales’ head. Recognition of notches and scars in the edge trail of the fluke is common for sperm whales (*Physeter macrocephalus*) and humpback whales (*Megaptera novaeangliae*), and the shape and notches on the dorsal fin is used in the identification of the killer whales (*Orcinus orca*) or the bottle-nose dolphins (*Tursiops* sp.) [2][9][17].

As mentioned above, human-assisted recognition of dolphins and whales using pictures is a difficult and time consuming task. For this reason, some software products are available to assist researchers on this task, like DARWIN [4], [14]. However, these products are neither effective in all cetacean species, nor useful among species in which the same type of feature is used to produce the individual recognition. In particular, they require a quite accurate supervised landmarking, including identifying the tip of the fin, and the position of the notches to be able to compensate for the perspective distortion in taking the picture, an unrealistic requirement most of the times [4].

A major source of false negatives in individual identification in these systems is produced due to the unsuccessful application of 3D correction before matching a given record with previously identified individuals. This is a critical issue, because pixel-based matching (for instance, using Euclid distance) is not robust under landmark positioning differences, which are almost certain to occur due to intra and intersubjective appreciation errors. For this reason, the success of landmarking over images as an identification means is tied to the operators’ ability to produce accurate landmarkings consistently. For this and other reasons, according to Stewman ([13]), landmarking is not entirely reliable, and additional information is required during the record registration to optimize further identifications.

Even more difficult is identification in a Genus of southern hemisphere dolphins that have some species with rounded dorsal fins, because it is not possible to pinpoint landmarks. The *Cephalorhynchus* species, and particularly *C. commersonii*, require for their individual recognition to rely on the traditional method in which the operator is trained to find matches manually. The notches in the trailing edge of the dorsal fin, and also color variation patterns, are used for identification. The notches are visible at different angles, and therefore are more likely to be useful in photo-identification. In contraposition, other kind of scars and abnormalities in the coloration patterns are used as ancillary features, since generally they allow identifying the animal from only one side.

So far, no reference in the literature proposes the use of the metadata associated to the imagery as a filtering means to lighten and speed up the matching task. The purpose of this presentation is to show the preliminary results of a research line aimed to automatize marine mammal photo-identification. Apart from image-based techniques as the ones mentioned above, the ancillary information present in the photographic database is not taken advantage of. In a series of studies carried out in the Patagonian coast, a database of individually recognized Commersons dolphins had been kept in the LAMAMA-CECIMAR- CONICET Institute [7]. The information accrued includes not only pictures but also a series of dolphins' descriptors [6] (see Fig. 1). In this work we show how this information can be used in the context of automated recognition of individuals, achieving an identification accuracy above 90% employing only the images' metadata. This alleviates the cognitive burden of the researchers in applying the capture-recapture model, and shows that metadata combined with image-based techniques may derive new automated identification products that go beyond the state-of-the-art in marine mammal photo-identification.

2 Materials and methods

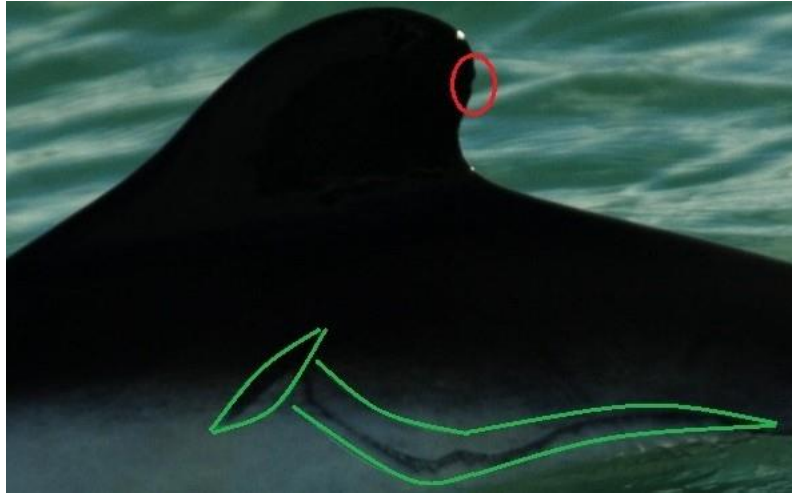
The preparation and process data were done in four stages following the methodology proposed by Ferrary [3] and Witten [16] for data mining procedures:

1. Define the goals and the information sources, and collect the data.
2. Analyze and preprocess the data.
3. Build and train models.
4. Perform validation tests.

In what follows of this Section we describe each of these steps (see also [8]).

2.1 Data collection and analysis

As stated above assisting in the identification of the dolphins can significantly reduce the operators time, by reducing the number of photographs to browse. We propose the use of a classification model that aids in the matching process using patterns present in the pictures' metadata. Also, we aim to determine how similar the marks of certain identified animals are. The information is persisted in 869 *MS Access* TM database records that hold the data and pictures of a population of Commersons dolphins, spanning along seven years, that have a total of 230 identified dolphins. These records, together with additional metadata used for photo-identification are used as instances (examples). From these instances we preselected only the specific attributes that may be relevant in the photo-identification task (see next subsection). Then the data was migrated to *MS Excel* TM, where data wrangling procedures were applied for cleansing and formatting. Finally, numerical values were assigned to nominal attributes, and to text attribute indicating ordinal values.



(a) Subtle notch and large auxiliary mark visible only from the left side



(b) Multiple visible notches and subtle auxiliary mark on the right side

Fig. 1. Individually identified Commersons dolphins in the LAMAMA-CESIMAR- CONICET data base. The red areas show notches in the trail of the dorsal fin. These are considered primary marks. The (often more subtle) auxiliary marks are shown in green. Primary marks are feasible to be recognized from both sides of the animal, while auxiliary marks generally are visible from only one side.

2.2 Attribute selection and data cleansing

A set of attributes that a priori hold significant information that could assist the photo-identification task were initially preselected to train the classifier:

- **Side.** The side of the animal where the picture was taken (“right” or “left”). The scars and amount of coloration attributes clearly depend on this attribute for a given individual.
- **Quality.** A quality index between 0 and 3 is assigned, related to image quality features including brightness and contrast, fin correctly focused, fin vertically aligned, and presence of water waves or drops obscuring the fin.
- **Distinctiveness.** A distinctiveness index between 0 and 3 is assigned given by the intrinsic features of the fin, including how visible or distinguishable are the notches and marks in the edge of the fin.
- **Scars.** A numerical quantity that represents the amount of recognizable scars observed in the picture. This attribute is related to side, quality and distinctiveness.
- **Coloration.** A numerical quantity that represents the amount of recognizable abnormal coloration spots observed in the picture. Also related to side, quality and distinctiveness.
- **Zones.** Specific areas in which the notches and marks may appear in the dorsal fin are designated with numbers 1 to 7 (see Fig. 2). This attribute takes a “true” value if the individual have notches or marks in this zone, and “false” otherwise.
- **Notches.** A numerical quantity that represents the amount of recognizable notches or marks in the edges of the fin. Not necessarily equal to the sum of all “true” values in the zone attributes since a notch may involve more than one zone, and also in a zone more than one notch may be located.
- **Catalog Number.** A unique id number for each identified animal.
- **“Big/large/extended”, “Medium”, “Small/little”.** These attributes describe the amount of marks with this size feature.
- **“Little bit/mild/imperceptible”, “Triangular”, “Rounded”, “Salience”.** These attributes describe the amount of marks with this shape feature.

The latter two attributes had to be carefully checked in order to be meaningful. Natural language attributes are prone to spelling and wording errors, and therefore disambiguation was required. Records which had incomplete information were discarded. Also, only the records that were originally used for photo-identification were considered.

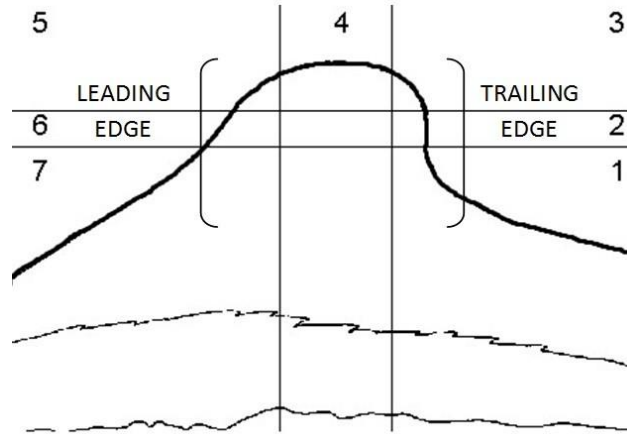


Fig. 2. The seven zones in the dorsal fin.

3 Results

3.1 Attribute selection and classification methods

Four different supervised classification algorithms, each pertaining to a different classification method, were used:

- Neural networks: Multilayer Perceptron,
- Bayesian classifier: NaïveBayes,
- Decision trees: J48,
- K-nearest neighbor algorithm: KStar

To avoid overfitting, the attributes were selected using *InfoGainAttributeEval*, *GainRatioAttributeEval* and *ChiSquaredAttributeEval* in conjunction with the *Ranker* search method, that ranked all attributes by their individual evaluations [1] [12]. In all trials, the results showed that the attributes *Rounded (R)*, *Triangular (T)*, *Zone 7 (Z7)* and *Quality (Q)* were mostly weighless and therefore were discarded. Also *CfsSubsetEval* combined with *BestFirst*, showed the same behavior for attributes *Little bit/mild/imperceptible (L)* and *Zone 4 (Z4)*. Removing some attributes we improved the accuracy of the classifier, with respect to the full set of attributes. In Table 1 the obtained accuracy of gradually subtracting these attributes can be appreciated

Table 1 Accuracy (in %) of the three different classifiers with different subsets of attributes

Dataset Name	NaiveBayes	KStar	Trees J48
full set	46.83	49.13	43.41
full set - Z7	67.73	68.35	66.02
full set - Z7 - Q	66.68	68.35	65.40
full set - Z7 - Q - R	66.62	67.43	64.57
full set - Z7 - Q - R - T	66.84	67.92	64.76

Table 2 Accuracy (in %) obtained with the four different classifiers with the complete dataset, and only with the records of individuals between 5 and 11 recaptures.

Dataset Name	NaiveBayes	KStar	Trees J48	Multilayer Perceptron
Complete dataset	46.69	49.14	43.34	41.64
5-11 dataset	67.73	68.35	66.02	63.47

Table 3 Accuracy of the four classifiers with the filtered training set (in %)

Dataset Name	NaiveBayes	KStar	Trees J48	Multilayer Perceptron
5-11 dataset	66.02	67.40	62.98	62.70

3.2 Model construction and validation

A standard cross-validation procedure was first performed. The dataset included 869 instances of 223 individuals. It is worth to note that the amount of “recaptures” of each individual is very uneven, ranging from 1 in most cases up to 24 in one case. Thus, the classes are unbalanced and therefore special consideration must be taken during the model construction to avoid biasing the classifier [10]. In our case, we splitted the dataset into three groups, according to the amount of recaptures of each individual (in ranges 1 to 5, 6 to 11, and more than 11). In the first group, the amount of instances per individual is too low to achieve a significant accuracy. On the other hand, in the third group the amount of individuals is too low (only eight), with a large amount of recaptures. For this reason, excluding these examples would avoid unbalancing the classes during learning without severely limiting the amount of individuals identified. Therefore, the most successful classification may be performed in the second group, with 373 instances of 54 individuals, each with between 5 and 11 recaptures. In Table 2 the classification results are shown for the whole dataset, and with only this 5-11 dataset. In all cases, the chosen dataset was split into training and validation subsets, and cross-validation was performed (10 folds).

Once the dataset (instances and attributes) was cleansed and filtered, it was split into two subsets, the training set with 362 instances (97%) and the validation set with 11 instances (3%). To test whether the training set is statistically meaningful, the ZeroR classifier was applied to check the accuracy of the majority class. The obtained result of 2.4862% correctly classified instances was well above the 1.8% (frac 154) expected by pure chance. Therefore the training set was considered to be adequate for classification training, which was performed with the same four learning methods with the same cross folding as mentioned above. The accuracies of the four methods are shown in Table 3.

As a further step, we combined the classifiers using AdaBoostM1, which is designed specifically for meta-classification purposes. Improvement was more notable with the trees.j48 classifier, achieving accuracy over 68% (see Table 4).

Table 4 Accuracies obtained with AdaBoostM1 together with the four different classifiers (in %) with the filtered training set

Algorithm	Accuracy
AdaBoostM1 + NaiveBayes	66.40
AdaBoostM1 + KStar	66.03
AdaBoostM1 + J48	68.23
AdaBoostM1 + MultilayerPerceptron	62.66

Table 5 Validation results (accuracies in %).

NaiveBayes	KStar	Trees J48	Multilayer Perceptron
81.8182	81.8182	90.9091	72.7273

3.3 Validation

Once the classifiers were trained with the filtered training set, we tested them with the instances in the validation set. This is the final intended use of the system, since these examples act as if they were new captures of already captured animals. In this situation, the accuracies were above 90% in TreeJ48, while other methods performed with accuracies among 72% and 81% (see Table 5).

4 Discussion and conclusion

We presented the result of applying machine learning techniques over the metadata of archives of 869 pictures taken of 230 different Commersons dolphins' images, as an aid in the process of manual identification of individuals. The metadata consisted of a set of manually taken annotations, one record per picture, which described different aspects of the animal's fin and surrounding appearance, together with ancillary information regarding the place and time where the picture was taken. The metadata was arranged as a set of attributes, and incomplete or incorrect records were filtered out. Attributes were further curated for schema conformance, mapping annotated values to numerical or ordinal categories adequate for the automated learning process. Finally, superfluous or noisy attributes were filtered out. Preliminary results showed that animals with few pictures (below 5) were almost impossible to identify with only this metadata. Therefore the learning algorithm was focused only on animals with among 5 and 11 records each. The resulting dataset comprised the metadata of 373 images taken of 54 individuals. A supervised classifier was trained with the identifications provided by the biologists. The resulting classifier was able to identify correctly above 90% of the individuals on the test set. These results show that the system may be quite helpful in the task of reducing the supervised time and effort of identification of new pictures, at least if there is a representative amount of priorly taken pictures of the same individual.

Current work around this project is focused on enhancing the accuracy on seldomly recaptured animals. Using metadata only, the semantics of the manual annotation can be further mined using text mining to deliver a more fine-grained set of nominal

attributes regarding the description of the shapes and coloration of notches and marks, using a convenient thesaurus. Also, we are currently working on image analytics, using first HaarCascade descriptors for ROI automatic detection (mainly of the fin in the pictures) and then morphometric descriptors to obtain an additional feature vector that combined with the available metadata may achieve better identification performance. Finally, we are considering other analytic features of the global population of captured animals. For instance, performing spatio-temporal analysis of capture-recapture patterns may reveal trends that may further aid in the automated identification process.

References

1. Machine Learning Group at the University of Waikato. Weka 3: Data mining software in java. <http://www.cs.waikato.ac.nz/ml/weka/>.
2. M. Bigg. An assessment of killer whale (orcinus orca) stocks of Vancouver island, British Columbia. Report, International Whaling Commission, 1982.
3. Mara José Ramirez Quintana César Ferri Ramirez Alhambra, José Hernandez Orallo. Introducción a la minería de datos. Pearson, 2004.
4. Ekerd College. Darwin. <http://darwin.ekerd.edu/>.
5. R. M. Cormack. Models for capture-recapture, pages 217–255. International Cooperative Publishing House, Maryland, 1979.
6. Mariano Alberto Coscarella. Ecología, comportamiento y evaluación del impacto de embarcaciones sobre manadas de tonina overa *Cephalorhynchus commersonii* en Bahía Engaño, Chubut. Ph.d., 2005.
7. Mariano Alberto Coscarella, Shannon Gowans, Susana Noem Pedraza, and Enrique Alberto Crespo. Influence of body size and ranging patterns on delphinid sociality: Associations among commerson's dolphins. *Journal of Mammalogy*, 92(3):544–551, 2011.
8. Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann and Ian H. Witten. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1):10–18, 2009.
9. P.S. Hammond, S.A. Mizroch, and G.P. Donovan. Individual recognition of cetaceans: Use of photo-identification and other techniques to estimate population parameters, volume 12 of International Whaling Commission Special Issue Series. International Whaling Commission, Cambridge, 1990.
10. Milton García Borroto Octavio Loyola Gonzalez, José Francisco Martínez Trinidad. Clasificadores supervisados basados en patrones emergentes para bases de datos con clases desbalanceadas. Technical Report CCC-14-004.14, Coordinación de Ciencias Computacionales. INAOE, Luis Enrique Erro 1. Sta. Ma. Tonantzintla, 72840, Puebla, Mexico., 2014.
11. Kenneth H. Pollock, James D. Nichols, Cavel Brownie, and James E. Hines. Statistical inference for capture-recapture experiments, volume 107 of Wildlife Monographs. The Wildlife Society Inc., Blacksburg, 1990.
12. Mark Hall Richard Kirkby Peter Reutemann Alex Seewald David Scuse Remco R. Bouckaert, Eibe Frank. WEKA Manual for Version 3-6-12, 2014.
13. John Stewman, Kelly Debure, Scott Hale, and Adam Russell. Iterative 3-d pose correction and content-based image retrieval for dorsal fin recognition. In International Conference Image Analysis and Recognition, pages 648–660. Springer, 2006.14. Texas A&M University. Fin scan.
15. Hal Whitehead. Mark-recapture estimates with emigration and re-immigration. *Biometrics*, 46:473–479, 1990.

16. Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.
17. Bernd Wrsig and Melany Wrsig. The photographic determination of group size, composition and stability of coastal porpoises (*tursiops truncatus*). *Science*, 198:755–756, 1977.

Búsqueda por Similitud de Posiciones de Ajedrez

Diego Gonzalez¹, Andrés Pascal¹, Anabella De Battista¹,
Norma Herrera²

¹ Dpto. de Sistemas de Información, Universidad Tecnológica Nacional,
Entre Ríos, Argentina,
{gonzalezdiego014, andrespascal22, anadebattista}@gmail.com

² Dpto. de Informática, Universidad Nacional de San Luis, Argentina,
nherrera@unsl.edu.ar

Resumen. Las búsquedas por similitud constituyen un campo de estudio de gran importancia en la actualidad. En el presente trabajo se propone una función distancia para consultar por similitud posiciones de ajedrez sobre bases de datos de partidas, ya que actualmente estas consultas están limitadas a búsquedas exactas. Se evalúa su comportamiento mediante distintos tipos de pruebas sobre las fases del juego.

Palabras clave: Búsquedas por similitud. Ajedrez. Espacios métricos. Función de distancia

1 Introducción

Las búsquedas en las bases de datos tradicionales se basan en el concepto de búsqueda exacta: la base de datos es dividida en registros, teniendo cada registro campos completamente comparables; una consulta a la base retorna todos aquellos registros cuyos campos coincidan con los aportados en la búsqueda. En la actualidad, muchas aplicaciones tienen como necesidad buscar en grandes bases de datos objetos que sean similares a uno dado. Este tipo de búsqueda se conoce con el nombre de *búsqueda por similitud* y surge en diversas áreas [1, 2]. En este artículo nos enfocamos en la búsqueda por similitud de posiciones en grandes bases de datos de partidas de ajedrez. Denominamos *posición* a un tablero de ajedrez con piezas ubicadas en celdas definidas, resultante de 0 o más movimientos de una partida.

El juego de ajedrez es foco de estudio desde hace más de cinco décadas en varias disciplinas, por ejemplo, podemos nombrar entre ellas: inteligencia artificial, elaborando algoritmos para la evaluación de las posiciones [3] y ciencias educacionales realizando estudios que exponen las ventajas de su aprendizaje para el desarrollo de habilidades personales [4].

Dentro de las ciencias de la computación, la mayor parte de los estudios se basan en el análisis de variantes sobre una posición o los llamados motores de evaluación de posiciones. Con respecto a las bases de datos, se han realizado estudios para obtener

conocimiento sobre la comparación de distintos motores como así también para la obtención de conocimiento sobre las partidas [5, 6]. Sobre las aplicaciones que consultan partidas, hoy en día se utilizan búsquedas por exactitud de las posiciones; es decir que las posiciones que se comparan deben tener cada pieza ubicada en el mismo lugar. Éste tipo de búsquedas es limitado ya que, en el ajedrez profesional, los jugadores no memorizan la exacta posición de las piezas, sino que identifican patrones dentro de una posición [7].

Aún no existe un sistema que permita búsquedas por similitud sobre posiciones de Ajedrez. La idea de las búsquedas por similitud, es obtener posiciones que puedan tener algunas variaciones respecto a la posición consultada. En éste artículo proponemos una función de distancia para las búsquedas por similitud de posiciones de ajedrez sobre una base de datos de partidas. Este tipo de consulta es útil para que aprendices del juego, ante una situación determinada, conozcan posibles jugadas o estrategias que siguieron jugadores profesionales en partidas reales. También podría ser utilizada por motores de ajedrez para podar los árboles de búsqueda o para mejorar su capacidad de juego.

Éste artículo está organizado de la siguiente manera: en la Sección 2 se describe el trabajo relacionado. En la Sección 3 se define y analiza la función distancia desarrollada, mostrando resultados experimentales de su funcionamiento en la Sección 4. Finalmente en la Sección 5 se expresan las conclusiones y el trabajo futuro.

2 Trabajo Relacionado

2.1 Búsquedas en Espacios Métricos

El problema de búsqueda de posiciones de ajedrez puede abstraerse como un universo de objetos U (las posiciones) y una función de distancia d que modela la similitud entre los objetos del universo. El par (U, d) se denomina *espacio métrico* y es uno de los modelos más utilizados como soporte para las búsquedas por similitud [1, 2].

Formalmente un espacio métrico es un par (U, d) donde U es un universo de objetos y $d : U \times U \rightarrow \mathbb{R}^+$ es una función de distancia definida entre los elementos de U que mide la similitud entre ellos; esto significa que a menor distancia, más cercanos o similares son los objetos. Esta función d cumple con las propiedades características de una función métrica:

- (a) $\forall x, y \in U, d(x, y) \geq 0$ (positividad)
- (b) $\forall x, y \in U, d(x, y) = d(y, x)$ (simetría)
- (c) $\forall x \in U, d(x, x) = 0$ (reflexividad)
- (d) $\forall x, y, z \in U, d(x, y) \leq d(x, z) + d(z, y)$ (desigualdad triangular)

La base de datos es un subconjunto finito $X \subseteq U$ de cardinalidad n . Existen dos tipos básicos de consultas en este modelo, la *búsqueda por rango* y la *búsqueda de los k vecinos más cercanos*.

La búsqueda por rango se denota $(q, r)_d$, donde $q \in U$ es el elemento de consulta que llamaremos *query*, y r es el radio de tolerancia. Una búsqueda por rango consiste en recuperar todos los objetos de la base de datos que se encuentren como máximo a distancia r de q , es decir:

$$(q, r)_d = \{x \in X / d(q, x) \leq r\}$$

La búsqueda de los k vecinos más cercanos se expresa como $NN_k(q)_d$, y consiste en recuperar los k elementos de la base de datos que posean menor distancia a q , es decir:

$$NN_k(q)_d = A, |A| = k, A = \{x \in X / \forall y \in (X-A), d(q, x) \leq d(q, y)\}$$

Para resolver estas consultas con mayor eficiencia que $O(n)$ evaluaciones de la función de distancia (que es el costo de recorrer secuencialmente la base de datos), se utilizan índices que ahorran cálculos durante el proceso de búsqueda [8, 9, 10].

2.2 Búsqueda de posiciones de Ajedrez

Particularmente dentro de la inteligencia artificial nos encontramos con varios trabajos relacionados con algoritmos para la evaluación de posiciones, en los que se aplican funciones de evaluación cuyo objetivo principal es obtener variantes¹ que proporcionen beneficios al jugador que tiene su turno en un juego [3].

Existen hoy en día diversos tipos de base de datos de ajedrez y con gran cantidad de datos, como por ejemplo los libros de Aperturas², en los cuales se aplican búsquedas exactas; es decir, se requiere que coincida la posición de cada pieza de la consulta con la misma pieza en el elemento consultado. A su vez, existen bases de datos sobre partidas jugadas, posiciones de medio juego y posiciones finales. Pero en todos los casos las consultas se realizan de la misma manera, lo cual es una limitante ya que cualquier detalle distinto sin relevancia no permite encontrar posiciones que serían de utilidad. En lugar de ello, hay propuestas como en [11] que muestran la importancia de que la búsqueda tenga en cuenta aspectos semánticos que incluyan el concepto o idea de la posición.

Por otro lado, en [7] se clasifica la idea de similitud en el ajedrez en similitud visual y similitud abstracta. Los Jugadores profesionales se centran principalmente en la similitud abstracta, identificando patrones dentro de cada posición, a diferencia de los jugadores principiantes que se basan en similitudes visuales.

Es importante considerar que la cantidad de combinaciones posibles dentro del juego está dada por la siguiente fórmula: $64! / (32! * (8!)^2 * (2!)^6)$ [12] dando un resultado aproximado de 10^{43} . En este cálculo hay posiciones que son ilegales para el juego, por lo cual se estima que la cantidad de posiciones válidas es alrededor de 10^{40} .

1 Conjunto de jugadas.

2 Jugadas iniciales.

Al consultar una posición, es difícil definir todos los patrones que puede encontrar un profesional del juego. En su lugar, una solución alternativa es contar con una función que calcule la similitud entre dos posiciones.

3 Función Distancia Propuesta

Existen decenas de funciones de distancia genéricas, que pueden aplicarse a distintos problemas [13]. Sin embargo, en casos especiales es necesario definir una función específica que represente el significado de la similitud entre dos objetos para el problema particular. En el caso de las posiciones de Ajedrez, la función que se define en este artículo se basa en la idea de medir la cantidad de movimientos necesarios para transformar una posición en otra, y en caso de que no se pueda, incluir una penalización en el cálculo que represente la ausencia de la pieza.

3.1 Descripción de la Función de Distancia

Un tablero se compone de sesenta y cuatro casillas en las cuales se ubican treinta y dos piezas divididas en dos bandos de diferente color. Dentro de la base de datos tenemos conjuntos de posiciones que corresponden a partidas jugadas. Cada posición se representa con una notación ajedrecística llamada FEN³; dicha notación es una cadena de texto que describe la ubicación de cada una de las piezas representándola mediante una letra. Las letras en minúscula representan las piezas negras y las letras en mayúsculas las blancas. Se utilizan números para representar las casillas vacías y la barra de división como separación de cada fila. En la Fig. 1 se muestra la posición inicial de una partida y el FEN asociado.

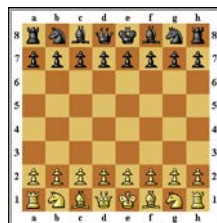


Fig. 1. FEN asociado: *rnbqkbnr/pppppppp/8/8/8/PPPPPPP/RNBQKBNR*

El objetivo de la función distancia es obtener el costo mínimo en movimientos que resulta de posicionar cada una de las piezas de un tablero origen en la misma posición de un tablero destino. El costo del movimiento de cada pieza se calcula como si el tablero estuviese vacío. En caso de que no sea posible por falta de la misma pieza en el otro tablero, se introduce un costo que depende de tipo de pieza, pero que es siempre mayor al máximo costo de movimientos de dicha pieza. Los costos utilizados

³ Forsyth-Edward Notation

asociados a las piezas fueron: Dama = 10, Torre = 8, Alfil = 7, Caballo = 6 y Peón = 5. El Rey no posee costo ya que siempre debe existir en ambos bandos.

Cada tipo de pieza tiene una función para obtener la cantidad de movimientos, ya que las distintas piezas del juego tienen movimientos diferentes. Para las piezas Rey, Dama y Alfil, se validan los movimientos de cada pieza y en base al movimiento de cada una se devuelve el valor correspondiente. Para las restantes piezas: Torre, Caballo y Peón, al no ser piezas únicas (par de Torres, par de Caballos, ocho Peones), se construye una matriz de costos de movimientos y se utiliza el método húngaro [14] para determinar la asignación óptima de correspondencia.

3.2 Algoritmo

A continuación se presenta el pseudocódigo del algoritmo de la función de distancia propuesta.

```

Función DistanciaAjedrez (pos1, pos2)
  distancia:= 0
  Foreach color in [blanco, negro]
    inicializar(matrizPeones) -- Peones
    Foreach peon1 in piezasDe(pos1, color)
      Foreach peon2 in piezasDe(pos2, color)
        matrizPeones[peon1, peon2]:=
          movs( Peon,
                casillaDe(peon1, pos1, color),
                casillaDe(peon2, pos2, color))
    distPeones:= metodoHungaro(matrizPeones)

  inicializar(matrizCaballos) -- Caballos
  Foreach caballo1 in piezasDe(pos1, color)
    Foreach caballo2 in piezasDe(pos2, color)
      matrizCaballos:= movs( Caballo,
                             casillaDe(caballo1, pos1, color),
                             casillaDe(caballo2, pos2, color))
  distCaballos:= metodoHungaro(matrizCaballos)

  inicializar(matrizTorres) - Torres
  Foreach torre1 in piezasDe(pos1, color)
    Foreach torre2 in piezasDe(pos2, color)
      matrizTorres:= movs( Torre,
                           casillaDe(torre1, pos1, color),
                           casillaDe(torre2, pos2, color))
  distTorres:= metodoHungaro(matriztorres)
  -- Piezas únicas
  distRey:= movs( Rey, casillaDe(Rey, pos1, color),

```

```

casillaDe(Rey, pos2, color)
distDama:= movs( Dama, casillaDe(Dama, pos1, color),
                casillaDe(Dama, pos2, color))
distAlfilBlanco:= movs( AlfilBlanco,
                        casillaDe(AlfilBlanco, pos1, color),
                        casillaDe(AlfilBlanco, pos2, color))
distAlfilNegro:= movs( AlfilNegro,
                       casillaDe(AlfilNegro, pos1, color),
                       casillaDe(AlfilNegro, pos2, color))

distancia:= distancia + distRey + distDama +
            distCaballos + distPeones + distTorres +
            distAlfilBlanco + distAlfilNegr
Return distancia

```

La función *casillaDe(Pieza, Posición, Color)* devuelve la casilla (compuesta por fila y columna) correspondiente a la Pieza del Color dado, en el tablero representado por Posición. La función *movs(TipoDePieza, Casilla1, Casilla2)* devuelve la cantidad mínima de movimientos de una pieza del TipoDePieza, desde la Casilla1 a la Casilla2. Por problemas de espacio no se explica en detalle el mecanismo de cálculo de esta función.

3.3 Ejemplo

A continuación se presenta un ejemplo del cálculo de la Función de Distancia para un caso en que los tableros no contienen las mismas piezas. La Fig. 2 muestra las posiciones comparadas y Tabla 1 el cálculo de la función de distancia.

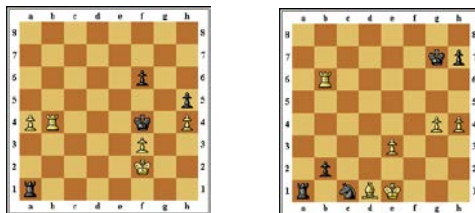


Fig. 2. Comparación de dos posiciones con distintas piezas

Tabla 1. Cálculo de la función de distancia para las posiciones de la Fig. 2.

Pieza	Color	Casilla inicial	Casilla objetivo	Distancia
Rey	Blanco	F2	E1	1
Torre	Blanco	B4	B6	1
Alfil	Blanco	-	D1	7
Peón	Blanco	A4	-	5

Peón	Blanco	F3	G4	1
Peón	Blanco	H4	H4	0
Rey	Negro	F4	G7	3
Torre	Negro	A1	A1	0
Caballo	Negro	-	C1	6
Peón	Negro	F6	B2	4
Peón	Negro	H5	H7	2
Total				30

3.4 Análisis de la Función de Distancia como métrica

Para que una función sea métrica, debe de cumplir cuatro propiedades: positividad, reflexividad, simetría y desigualdad triangular.

La función propuesta cumple con la positividad, ya que es una sumatoria de números naturales (cantidad de movimientos de una pieza para alcanzar una posición a partir de otra). Es reflexiva, ya que cuando se mide la distancia entre dos posiciones iguales al encontrarse las piezas en los mismos lugares, el resultado siempre es 0. También es simétrica, debido a que la cantidad mínima de movimientos de una pieza para ir desde una casilla de origen a una casilla destino, es siempre la misma que la cantidad mínima desde la casilla destino a la de origen, y la función se calcula como la sumatoria de estas cantidades.

Respecto a la desigualdad triangular, aunque todavía no se hizo el estudio analítico de la función, se realizaron una serie de pruebas para verificar su cumplimiento estableciendo tres conjuntos a los cuales se les aplicó la función distancia entre sus elementos. En total se evaluaron 60.000 ternas y en todos los casos se verificó el cumplimiento de esta propiedad. A pesar de que esta verificación no constituye una demostración, es un punto de partida importante, ya que en caso de no cumplirla para una cantidad reducida de casos, igualmente se podrían utilizar índices métricos para acelerar la búsqueda tal como se plantea en [15].

4 Resultados Experimentales

Para la evaluación de la función distancia, se obtuvo de Internet una base de datos con un total de 179564 posiciones correspondientes a 2273 partidas. El objetivo de los experimentos fue verificar la eficacia o precisión de la función, es decir, que los objetos resultantes de una consulta sean los esperados. Para ello se definieron tres situaciones distintas; aperturas, medio juego y finales, y se obtuvieron lotes de consultas de dos maneras: consultas externas (posiciones extraídas de otras partidas de ajedrez que no están en la base de datos) y consultas de posiciones modificadas (en este caso se extraen posiciones de la base de datos a las cuales se les realizan modificaciones y el resultado se utiliza como consulta).

4.1 Consultas externas

4.1.1 Pruebas de Apertura

Para las pruebas de apertura se obtuvieron 40 posiciones desde Internet que se corresponden a las primeras 10 jugadas de una partida y con distinto código de Apertura perteneciente a la Enciclopedia de Aperturas de Ajedrez (ECO). La enciclopedia está dividida en cinco categorías que se representan con las letras desde A hasta E, y cada categoría incluye hasta cien subcategorías.

El objetivo de esta prueba fue determinar si el vecino más cercano tiene la misma categoría y subcategoría de apertura que la consulta.

Tabla 2. Pruebas externas de Apertura.

	Aciertos	%
Aperturas	40	100 %

En la Tabla 2 se muestran los resultados de las consultas externas de Aperturas. En el 100% de los casos la función clasificó las aperturas de manera correcta.

4.1.2 Pruebas de Medio Juego y Finales

Para las pruebas de Medio Juego y Finales se obtuvieron desde Internet 100 posiciones de Medio Juegos correspondientes a jugadas entre 15 y 20 movimientos de una partida, y 50 posiciones Finales correspondientes a jugadas mayores a 40 de una partida. Se analizaron los 10 vecinos más cercanos, verificando su similitud visualmente.

Tabla 3. Pruebas externas de Medio Juegos y Finales.

	Aciertos	%
Medio Juegos	848	84,80 %
Finales	393	78,60 %

En la Tabla 3, se muestran los resultados de las pruebas de Medio Juego y Finales. Podemos observar que para el Medio Juego la función devuelve una mayor cantidad de aciertos que en las posiciones de Finales. Esto es principalmente porque en los Finales de juego hay una menor cantidad de piezas que en los Medio Juegos, por lo que la existencia o falta de una pieza cobra mucha más relevancia y la función de distancia propuesta no tiene en cuenta este aspecto. Esta situación muestra que es necesario ajustar los valores de los pesos máximos de las piezas, ya que, por ejemplo,

con los valores actuales es lo mismo que falten dos peones (valor 5 cada uno) que una dama (valor 10).

4.2 Consultas de posiciones modificadas

Para las pruebas de modificación, se obtuvieron tres posiciones de la base de datos por cada fase del juego y se le realizaron modificaciones. Las modificaciones realizadas a los tableros de Apertura fueron de movimiento y eliminación de piezas, mientras que para las pruebas de Medio Juego y Finales, se realizaron modificaciones de movimiento, eliminación y agregado de piezas.

Se realizaron 5 modificaciones a cada posición de Apertura, 8 modificaciones a cada posición de Medio Juego y 6 modificaciones a cada posición Final. En este caso se obtuvo el vecino más cercano de cada consulta y se lo clasificó como Acierto si el elemento resultante pertenecía a la misma partida.

Tabla 4. Pruebas de posiciones modificadas de Apertura, Medio Juegos y Finales.

	Aciertos	%
Apertura	13	86,67%
Medio Juegos	23	95,83%
Finales	17	94,44%

Como podemos observar en la Tabla 4, para las pruebas de modificación de posiciones de la base de datos, los porcentajes de acierto son elevados, lo que indica que la función propuesta es representativa de la similitud real de las posiciones. El valor menor de aciertos es el de las Aperturas, ya que al inicio, las partidas de un mismo tipo son muy similares, por lo cual la función puede retornar un elemento de otra partida parecida.

5 Conclusiones y Trabajo Futuro

En este artículo proponemos una función distancia para búsquedas por similitud de posiciones de ajedrez en una base de datos de partidas jugadas y mostramos su buen comportamiento en una base de datos real. Esta función se puede utilizar para clasificar Aperturas, como soporte al aprendizaje del ajedrez y para mejorar el funcionamiento de motores de juego, entre otros usos.

La función se puede mejorar aún, para que en la etapa de final de un juego tenga en cuenta que la calidad y cantidad de piezas es mucho más importante que sus posiciones.

A continuación, se enumeran los trabajos que quedan por realizar:

- Analizar las posiciones finales y modificar la función distancia incluyendo la cantidad y valor relativo de las piezas, ya que a medida que el juego avanza, el valor de cada pieza aumenta.

- Ajustar el valor de la distancia de los peones cuando no correspondan en un movimiento válido.
- Demostrar que la función distancia sea métrica para posteriormente plantear soluciones en la eficiencia de las consultas.
- Realizar ajustes a la función distancia donde se tengan en cuenta los patrones de una posición para realizar búsquedas de forma abstracta.

Referencias

1. E. Chavez, G. Navarro, R. Baeza-Yates, and J.L. Marroquin. *Searching in metric spaces*. ACM Computing Surveys, 33(3): 273–321, (2001).
2. R. Baeza-Yates. Searching: an algorithmic tour. In A. Kent and J. Williams, editors, *Encyclopedia of Computer Science and Technology*, volume 37, pages 331-359. Marcel Dekker Inc. (1997).
3. Vazquez-Fernandez, E., Coello, C.A., Sagols, F.D.: An Evolutionary Algorithm for Tuning a Chess Evaluation Function: Applied Soft Computing, vol. 13, pp. 3234-3247.(2013)
4. Sala, G., Gobet, F.: Do the benefits of chess instruction transfer to academic and cognitive skills? A meta-analysis. In: Educational Reserch Review, vol. 18, pp 46-57. (2016)
5. Fürnkranz, J.: Knowledge Discovery in Chess Databases: A Research Proficial Intelligence. (1997)
6. Mathieu Acher, François Esnault.: Large-scale Analysis of Chess Games with Chess Engines: A Preliminary Report. [Technical Report] RT-0479, Inria Rennes Bretagne Atlantique. (2016)
7. León-Villagrà, P., Frank, J.: Categorization and Abstract Similarity in Chess. COGSCI . (2013)
8. E. Chavez and K. Figueroa. Faster proximity searching in metric data. Proceedings of MICAI 2004. LNCS 2972, Springer, (2004).
9. R. Baeza-Yates, W. Cunto, U. Manber, and S. Wu. Proximity matching using fixed-queries trees. In Proc. 5th Combinatorial Pattern Matching (CPM'94), LNCS 807, pages 198-212 (1994).
10. D. Novak, M. Batko, P. Zezula: Metric Index: An efficient and scalable solution for precise and approximate similarity search. Information Systems, vol. 36, issue 4, pp 721–733 (2011).
11. Santos, G.L., Rachele, H.: ICONCHESS: An Interactive CONSultant for CHESS middlegames. ICLS '96 Proceedings of the 1996 international conference on Learning sciences, pp. 456-461. (1996)
12. Hacohen-kerner, Y.: Computer Chess and Indexing. In: Encyclopedia of Computer Sciece and Technology, vol. 44, pp.59-80. (2001)
13. S. H. Cha. Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions. International Journal of Mathematical Models and Methods in Applied Sciences (2007).
14. H. W. Kuhn,: The Hungarian Method for the assignment problem. Naval Research Logistics Quarterly, 2: 83–97 (1955).
15. T. Skopal: On Fast Non-metric Similarity Search by Metric Access Methods. Advances in Database Technology - EDBT 2006. 10th International Conference on Extending Database Technology, Munich, Germany, pp 718-736 (2006)

Vector-based word representations for sentiment analysis: a comparative study

M. Paula Villegas¹, M. José Garcarena Ucelay¹, Juan Pablo Fernández¹, Miguel A. Álvarez-Carmona², Marcelo L. Errecalde¹, Leticia C. Cagnina^{1,3}

¹LIDIC, Universidad Nacional de San Luis, San Luis, Argentina

²Language Technologies Laboratory, INAOE, Puebla, México

³CONICET, Argentina

{villegasmariapaula74, mjgarcarenaucelay, miguelangel.alvarezcarmona, merrecalde, lcagnina}@gmail.com

Abstract. New applications of text categorization methods like opinion mining and sentiment analysis, author profiling and plagiarism detection requires more elaborated and effective document representation models than classical Information Retrieval approaches like the Bag of Words representation. In this context, word representation models in general and vector-based word representations in particular have gained increasing interest to overcome or alleviate some of the limitations that Bag of Words-based representations exhibit. In this article, we analyze the use of several vector-based word representations in a sentiment analysis task with movie reviews. Experimental results show the effectiveness of some vector-based word representations in comparison to standard Bag of Words representations. In particular, the Second Order Attributes representation seems to be very robust and effective because independently the classifier used with, the results are good.

Keywords: text mining, word-based representations, text categorization, movie reviews, sentiment analysis

1 Introduction

Selecting a good document representation model is a key aspect in text categorization tasks. The usual approach, named *Bag of Words* (BoW) model, considers that words are simple indexes in a term vocabulary. In this model, originated in the information retrieval field, documents are represented as vectors indexed by those words. Each component of a vector (document) represents the weight that the corresponding word has associated in that document. Well known limitations of the BoW representations are the sparseness of the resulting vectors and the loss of any information about the locations of words within documents.

Several proposals from the computational linguistic area have attempted mitigating the above mentioned limitations by considering more elaborated *word representations* [1]. In those approaches, words are considered as first-class objects that allow “more

semantic” comparisons among words and, in consequence, better categorization results.

Two word representations that have gained increasingly interest in the last years are the ones usually referred as *distributional* and *distributed* word representations [1]. Both approaches represent words as vectors that capture contextual information of the corresponding word within the documents. Nevertheless, they differ in the way those vectors are obtained, with an emphasis of distributed representations in *learning* word representations, typically using neural networks models. The latter approach is usually referred as *word embedding*.

On the other hand, sentiment analysis and opinion mining is an area receiving increasing attention from both, industry and academia. In this kind of problems, several of the above mentioned word representation methods have been applied with varying performance [2, 3]. However, to the best of our knowledge, most of those studies have focused on a particular word representation method, with limited comparison to other more elaborated word representations.

In the present work, this research gap is addressed by considering and comparing six effective word representations in a sentiment analysis task related to movie reviews. We focus on vector-based word representations by considering four methods of the distributional area (SOA, LSA, LDA, DOR) and one representative of the distributed representation approach (Word2Vec). Our study takes BoW as baseline and the result analysis is carried out on a subset of the IMDB Review Dataset.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the representations used in our comparative study. In Section 3, we describe the original sentiment analysis dataset and the proposed corpus used in our experiments. Section 4 shows the settings and the results corresponding to the experimental study. Finally, in Section 5 some conclusions are drawn and future works are proposed.

2 Vector-based word representations

In vector-based word representations (VWRs), each word has associated a *vector*. Each dimension’s value corresponds to a *feature*, named *word feature*, which might have a semantic or grammatical interpretation [1]. VWRs are supposed to overcome limitations of the BoW model by allowing to capture richer relational structure of the lexicon [2]. This is achieved by encoding continuous (non-binary) similarities between words as distance or angle between word vectors in a high-dimensional space.

This section presents some basic explanations of the VWRs used in our work, four from the distributional representation area (subsections 2.2 to 2.6) and one representative of the distributed approach (subsection 2.7). Space limitations prevent us from giving detailed explanations of the methods and involved formulas but the interested reader can obtain them from the cited references. This section also includes in subsection 2.1 a short description of the standard BoW approach used as baseline in our study.

2.1 Bag of Words (BoW)

The traditional Bag of Words (BoW) representation is one of the most used in text categorization tasks. This popular representation is simple to implement, fast to obtain and can be used under different weighting schemes. However, the ordering of the words in the document is ignored, and the semantic and conceptual information are lost. As its name indicates, the document is represented as a bag of words and only the number of occurrences of each word is maintained. Formally, a document d is represented by the vector of weights $d_{bow} = \langle w_1, w_2, \dots, w_n \rangle$ where w_i depends on the weighting scheme selected (*tf*, *tf-idf*, *Boolean*, etc.) and n is the size of the vocabulary of the dataset [4]. Often, the vectors of this representation could be very sparse if the documents are enough different.

2.2 Second Order Attributes (SOA)

SOA is a low dimensional representation with a high level of representativeness [5] that constructs a space of profiles from which vectors of terms are built. Then, using those vectors, the representation of the documents is obtained with respect to the same profiles space. Firstly, SOA identifies the profiles according to the categories to be used for the classification of the data. Then, for each term in the vocabulary, a value indicating the relationship (term frequencies in the documents belonging to the profile) between that term with each profile, is calculated and saved into a vector. Finally, the document representation is obtained adding all the vectors corresponding to all the terms included in the document, weighted by the relative frequency of that term in the document.

2.3 Latent Semantic Analysis (LSA)

LSA is a method for representing the contextual-usage meaning of words. LSA can associate words and its contribution to automatically generated concepts (topics) [6]. LSA assumes that words that are close in meaning will occur in similar pieces of text [7]. This is usually named the *latent space*, where documents and terms are projected to produce a reduced topic based representation. LSA is built from a matrix M where m_{ij} is typically represented by the *tf-idf* weight of the word i in document j . LSA uses the Singular Value Decomposition (SVD) and makes a reconstruction with K (topics) dimensions. It makes the best possible reconstruction of the M matrix with the less possible information and noise [8].

2.4 Latent Dirichlet Allocation (LDA)

LDA is a generative probabilistic model for collections of discrete data. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of K topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities [9]. Basically, LDA observes each document, and randomly assigns each word in the document to one of the K topics. To improve the topics distribution LDA for each word in each document

assumes that this word is in a wrong topic and tries to fit the word in others topics maximizing the probability that the word is with others words with the same context. This process is repeated several times until the distribution of topical not change substantially.

2.5 Document Occurrence Representation (DOR)

DOR is an effective distributional representation based on the idea of how the semantic of a document can be described with a function of the bag of terms occurring on it, likewise the semantic of a term can be described with a function of the bag of the documents in which the term is [10]. Therefore, the more frequent the term is in a document, the more important that document is to characterize the semantic of that term. For each term in the vocabulary, DOR constructs vectors of weights considering the contribution of each document in the collection to the semantic of the term. Then, the representation of a document is obtained adding all the vectors of those terms occurring in the document [11] considering a weighting scheme.

2.6 Word2Vec

Word2Vec [12] is a representation learning method used to obtain distributed representations of words (also named *words embeddings*). The method learns a model using few labeled documents and many unlabeled documents with a neural network algorithm. The goal is to obtain word vectors with meaningful characteristics, that is, related words are in the same group if these appear in similar contexts, have similar meaning and/or have semantic relationships. The trained model is based on feature vectors (one for each word in the vocabulary) with a maximum number of dimensions. Then, for representing a document, the method searches for the words included in the text and uses the word vectors to construct the averaged vector that will be the Word2Vec representation.

3 Sentiment analysis

The sentiment analysis is a complex and challenging task in machine learning. When people write about their emotions, they can use sarcasm, some ideas can be ambiguous, and they use non common words. Then, the analysis of the sentiment of a text is not a trivial task. The sentiment label of a text can be categorical (positive or negative), continuous (a number indicating the level of positivity polarity, for example) or multidimensional (a combination of several types of labels). The first one was adopted in this work, so a document will have a positive or negative polarity.

Next, we describe the IMDB Review Dataset, a corpus containing movie reviews for sentiment analysis. Then, we present the subset of the original corpus used in our experiments.

3.1 Original corpus

IMDB Review Dataset [2] is a collection of 50000 movie reviews extracted from the popular movies, TV and celebrities content site www.imdb.com. The reviews are labeled as positive or negative regarding the sentiment polarity of the content. Both categories have the same number of reviews, that is, 25000 are positive and 25000 are negative reviews. In order to perform classification tasks, the dataset has been divided into train and test sets. Then, the training set has 25000 reviews labeled with the corresponding polarity and the testing set has also 25000 but unlabeled reviews [13].

3.2 Proposed corpus

We used only the labeled set of the IMDB Review Dataset because we need the category of each review for testing the classifiers with the different representations. We proposed a new corpus to use in this work which was constructed from the original IMDB Review Dataset training set. Our corpus is also divided in training and testing sets. Our train (named here as $IMDB_{train}$) and our test (named here as $IMDB_{test}$) sets were made selecting randomly the 90% of the reviews for the first one (constructing the model) and the 10% for the second one (testing), respectively. We maintain the same distribution of documents in each category (positive and negative) such as the original dataset, that is, the same number of reviews. Thus, $IMDB_{train}$ has 22500 reviews and $IMDB_{test}$ has 2500 reviews.

4 Experiments

4.1 Settings

We used the proposed corpus (Section 3.2) with 25000 movie reviews for all the experiments. We performed a pre-processing which included converting all the reviews to lowercase and removing stop words, numbers, punctuation marks and any special character. Then, we obtained the different representations with the following settings:

- BoW: the word vectors were weighted by the relative frequency of the word in each document. Only the 5000 most frequent terms were considered.
- Word2Vec: we kept the default parameters such as were suggested in [13], that is, the minimum word count is 40, word vector's dimensionality is 300, context window size is 10 and down sample setting for frequent words is $1e-3$.
- LSA and LDA: we used $k = 300$ for the number of topics. After performing preliminary experiments, we obtained the best accuracy with this value.

- SOA: this representation has no parameters to set.
- DOR: this representation maintains in memory huge dimensional matrices and its computation (number of terms by number of documents) can be impossible to calculate even using modern computers. For this reason, we previously performed a study of information gain over the attributes (words) of $\text{IMDB}_{\text{train}}$. In that study, we found that adjectives were mainly the type of words which ranked first in the information gain analysis. This fact is illustrated in Table 1, which shows an example of the first 10 words with the highest information gain values. As we can observe from Table 1, almost all the words are adjectives. Hence, we only considered the adjectives as terms of each document for this representation. The weighting scheme used in these experiments was the Boolean as our first approach.

Table 1. The first 10 words with the highest information gain value obtained from $\text{IMDB}_{\text{train}}$.

	Term	Information Gain		Term	Information Gain
1	bad	0.063703	6	excellent	0.021465
2	worst	0.051894	7	terrible	0.020994
3	waste	0.033709	8	worse	0.019704
4	great	0.032522	9	wonderful	0.01893
5	awful	0.032153	10	stupid	0.017746

The experiments were performed using Naïve Bayes (NB) and LibLINEAR (LL) classifiers. We constructed the models with $\text{IMDB}_{\text{train}}$ and validated it using $\text{IMDB}_{\text{test}}$. We utilized the accuracy as measure to evaluate the performance of the classifications.

4.2 Results

The results of the experiments are shown in Table 2. The highest accuracy value is highlighted in bold. We decided to use BoW representation as a baseline to compare among the different representations.

Table 2. Accuracy obtained with Naïve Bayes (NB) and LibLinear (LL) classifiers for all the proposed document representations.

<i>Document Representations</i>						
<i>Classifiers</i>	BoW	SOA	Word2Vec	LSA	LDA	DOR
NB	73.00	87.08	74.68	68.17	54.53	58.76
LL	83.76	86.96	87.12	88.52	70.73	78.60
Average	78.38	87.02	80.90	78.35	62.63	68.68

As it can be seen in Table 2, the best result was achieved by LSA representation with LibLINEAR classifier such as has previously demonstrated to perform well for personality recognition [14]. However, LSA with Naïve Bayes was over the baseline. In fact, for almost all representations, the results obtained using Naïve Bayes are below those obtained using LibLINEAR with exception of SOA.

The experiments with SOA representation shows high accuracy values for both classifiers and these ranks first if we consider the average of performance with both classifiers. Thus, we can say SOA seems to be the most robust representation with respect to the classification algorithms. Additionally, SOA has been shown to be a very robust and effective representation to classify other types of texts as well [15, 16].

On the other hand, Word2Vec, obtained good results with the LibLINEAR classifier using default parameters. This performance could be better if we change the parameters settings; due to the fact that the different model parameters can affect the quality of the Word2Vec representation.

Otherwise, LDA and DOR (last two columns in Table 2) did not perform well, both obtained accuracy values under the baseline with both classification algorithms. LDA obtained the lowest results indicating the unsuitability of this representation for this task. DOR combined with LibLINEAR is only a 5% below the baseline although this value is just above the average of the baseline. These poor results can be due to the version of DOR used in this work, because it is the most basic one, as we described in Section 4.1.

We can conclude with this comparative study that LDA, Word2Vec and SOA are the most adequate representations (over the six tested) for the sentiment analysis in movie reviews. In particular, LDA and Word2Vec representation seem to be quite dependent to the classification algorithms used with. Conversely, SOA seems to be very robust in addition to the efficiency in the performance. This can be observed in the results of Table 2 in which the values obtained with Naïve Bayes and LibLINEAR are quite similar.

5 Conclusions and future work

This article presents a comparative study about different word representations for the sentiment analysis on movie reviews. In particular, we were interested in those representations based on vectors of features (words in our case). We analyzed distributional representations such as BoW, SOA, LDA, LSA and DOR, and a distributed representation such as Word2Vec. We can conclude with this preliminary study that with some effective representations (LSA and Word2Vec) the algorithms obtain good results although these depend on the classifier used. On the other hand, with SOA representation we can obtain enough good results independently the classifier used.

For future work we plan to make a similar experimental study but using the complete IMDB Review Dataset, that is the 50000 movie reviews, in order to see if the conclusions about the good performance of the LSA, Word2Vec and SOA representations are still the same. We also propose to execute DOR using other types of parts of speech like nouns, verbs and articles, and other weighting schemes in the construction of the representation. For Word2Vec representation we plan to test with different parameters, not using just the default ones proposed by the authors. We are interested to study the stability and the execution time for all the representations.

6 References

1. Turian, J., Ratinov, L., Bengio, Y.: Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10). Association for Computational Linguistics, pp. 384--394. Stroudsburg, PA, USA (2010)
2. Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, N. Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT '11), Vol. 1. Association for Computational Linguistics, pp. 142--150. Stroudsburg, PA, USA (2011)
3. Mesnil, G., Mikolov, T., Ranzato, M., Bengio, Y.: Ensemble of Generative and Discriminative Techniques for Sentiment Analysis of Movie Reviews. arXiv preprint arXiv:1412.5335. (2014)
4. Feldman, R., Sanger, J.: The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, (2007)
5. López-Monroy, A. P., Montes-y-Gómez, M., Escalante, H. J., Villaseñor-Pineda, L., Villatoro-Tello, E.: INAOE's Participation at PAN'13: Author Profiling Task Notebook for PAN at CLEF 2013. In: CLEF (Working Notes), CEUR-WS.org, (2013)
6. Landauer, T. K., Dumais, S. T.: A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, vol. 104(2), pp. 211--240. (1997)

7. Landauer, T., Foltz, P., Laham, D.: An introduction to latent semantic analysis. *Discourse processes*, vol. 25(2-3), pp. 259--284. (1998)
8. Landauer, T. K., McNamara, D. S., Dennis, S., Kintsch, W.: *Handbook of latent semantic analysis*. Psychology Press. (2013)
9. Blei, D. M., Ng, A. Y., Jordan, M. I.: Latent dirichlet allocation. In: *Advances in neural information processing systems*, pp. 601--608. (2001)
10. Lavelli, A., Sebastiani, F., Zanolini, R.: Distributional term representations: an experimental comparison. In: *Proceedings of the thirteenth ACM international conference on Information and knowledge management*. pp. 615--624. ACM. (2004)
11. Cabrera, J. M., Escalante, H. J., Montes-y-Gómez, M.: Distributional term representations for short-text categorization. In: *International Conference on Intelligent Text Processing and Computational Linguistics*. pp. 335--346. Springer Berlin Heidelberg. (2013)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space, CoRR. abs/1301.3781. (2013)
13. Kaggle competition, <https://www.kaggle.com/c/word2vec-nlp-tutorial/>
14. Álvarez-Carmona, M. A., López-Monroy, A. P., Montes-y-Gómez, M., Villaseñor-Pineda, L., Escalante, H. J.: INAOE's Participation at PAN'15: Author Profiling task. In: *CLEF (Working Notes)*, CEUR-WS.org. (2015)
15. Rangel, F., Rosso, P., Moshe Koppel, M., Stamatatos, E., Inches, G.: Overview of the author profiling task at pan 2013. In: *CLEF Conference on Multilingual and Multimodal Information Access Evaluation*. CELCT. 352-365. (2013)
16. Rangel, F., Rosso, P., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daeleman, W.: Overview of the 2nd author profiling task at pan 2014. In: *CEUR Workshop Proceedings*, Vol. 1180. CEUR Workshop Proceedings. pp. 898--927. (2014)

LSA64: An Argentinian Sign Language Dataset

Franco Ronchetti*¹, Facundo Quiroga*¹, César Estrebou¹, Laura Lanzarini¹,
and Alejandro Rosete²

¹ Instituto de Investigación en Informática LIDI, Facultad de informática,
Universidad Nacional de La Plata

{fronchetti,fquiroga,cesarest,laural}@lidi.unlp.edu.ar

² Instituto Superior Politécnico Jose Antonio Echeverría

{rosete}@ceis.cujae.edu.cu

Abstract. Automatic sign language recognition is a research area that encompasses human-computer interaction, computer vision and machine learning. Robust automatic recognition of sign language could assist in the translation process and the integration of hearing-impaired people, as well as the teaching of sign language to the hearing population.

Sign languages differ significantly in different countries and even regions, and their syntax and semantics are different as well from those of written languages. While the techniques for automatic sign language recognition are mostly the same for different languages, training a recognition system for a new language requires having an entire dataset for that language.

This paper presents a dataset of 64 signs from the Argentinian Sign Language (LSA). The dataset, called LSA64, contains 3200 videos of 64 different LSA signs recorded by 10 subjects, and is a first step towards building a comprehensive research-level dataset of Argentinian signs, specifically tailored to sign language recognition or other machine learning tasks. The subjects that performed the signs wore colored gloves to ease the hand tracking and segmentation steps, allowing experiments on the dataset to focus specifically on the recognition of signs.

We also present a pre-processed version of the dataset, from which we computed statistics of movement, position and handshape of the signs.

Keywords: sign language recognition, handshape recognition, lexicon, corpus, automatic recognition.

1 Introduction

Sign language (SL) recognition is a complex multidisciplinary problem. It bears many similarities to speech recognition, but presents some additional difficulties [8]:

1. There is little formal standardization in most sign languages, even within a region.

* Contributed equally

2. Sign language specification languages themselves are not well standardized, and there is no consensus on which type of specification is more appropriate.
3. Signs are intrinsically multimodal: they are formed by a combination of hand shapes, positions, movements, body pose, face expression, and lip movements. In contrast, speech recognition usually requires only sound input.
4. Creating datasets for training sign language recognition systems requires being able to capture and model all of these signals.
5. There are relatively few sign language users with respect to the general population, and therefore finding expert subjects to record signs is more difficult.

For these reasons, it is generally considered that we are still far from robust sign language recognition systems.

There are numerous publications dealing with the automatic recognition of sign languages, and [1,8] present reviews of the state of the art in sign language recognition. The full task of recognizing a sign language involves a multi-step process. In the context of video-based recognition, and considering only manual information, this process can be simplified as:

1. Tracking and segmenting the hands of the interpreter in every frame of the video.
2. Recognizing the shapes of the hands, the movements they made and their positions.
3. Recognizing the sign as a syntactic entity (a visual word).
4. Assigning semantics to a sequence of signs (a visual sentence).
5. Translating the semantics of the signs to the written language.

These tasks are mostly independent from each other, and involve different techniques. Tracking, segmentation and modeling of the hand are mostly signal processing and 3D modeling tasks, while assigning semantics to a sequence of signs and translating from sign to written languages are more related to natural language processing.

1.1 Datasets

There are many datasets for sign language recognition. We distinguish three types, depending on the problem they target: handshape recognition, sign recognition or sentence recognition. Each type of dataset presents a greater challenge than the previous one, and allows experiments with more steps of the recognition pipeline.

Table 1 presents the most prominent video-based, research-level datasets for recognition. Since the dataset described in this paper focuses on sign-level recognition, we only list sign and sentence-level datasets ³.

³ A more detailed reference about sign language datasets can be found at http://facundoq.github.io/unlp/sign_language_datasets/index.html

Table 1. Recognition-oriented, video based, sign language datasets used in recent papers.

Name	Classes	Subjects	Samples	Language level	Availability
DGS Kinect 40 [1]	40	15	3000	Word	Contact Author
DGS RWTH-Weather [8]	1200	9	45760	Sentence	Public Website
DGS SIGNUM [8]	450	25	33210	Sentence	Contact Author
GSL 20 [1]	20	6	840	Word	Contact Author
Boston ASL LVD [3]	3300+	6	9800	Word	Public Website
PSL Kinect 30 [2]	30	1	300	Word	Public Website
PSL ToF 84 [2]	84	1	1680	Word	Public Website

In general, the datasets that are video-based must rely on skin color tracking and segmentation, and are therefore not robust for background variations or interpreter clothes, as well as hand-hand or hand-face occlusions [6]. Usually, to perform features extraction these datasets require the addition of morphological information as a subsequent step to the color filtering to identify the position and shape of the hand, which can be extracted using depth cameras or other sensors, but these limit the application of the methods with respect to using normal video cameras, given the availability of each type of devices.

The largest sign language dataset available (in terms of number of classes), the *American Sign Language Lexicon Video Dataset* (ASLLVD) [3], contains more than 3300 signs from the American Sign Language, but near-perfect hand tracking and segmentation on this dataset is difficult [3], making it hard to use it to evaluate a sign recognizer that focuses on the syntactic and semantic recognition. The situation is similar for the SIGNUM and RWTH-Weather datasets. Moreover, the dataset ASLLVD has only 6 subjects, and an average of 3 samples per class.

1.2 Argentinian Sign Language Dataset (LSA64)

Sign languages are different in each region of the world, and each has their own lexicon and group of signs. Thus, sign language recognition is a problem that needs to be tackled differently in each region, since new movements or handshapes or combinations thereof require new training data, and possibly involve new challenges that were not considered before [8,1].

To the best of our knowledge, there are no available datasets for the Argentinian Sign Language (LSA). There are only a few dictionaries that focus on teaching the language. Since they were recorded with this aim in mind, they have only one sample for each sign, low image quality, and poor annotations, thus making them unsuitable for training automatic recognition systems. There

is need for a research-level dataset that represents the group of signs used in LSA.

This paper presents a sign dataset called LSA64. The dataset consists of 64 signs from the LSA, and was recorded with normal RGB cameras. It is publicly available⁴, and we also provide a preprocessed version of the dataset to facilitate experiments and reproducibility.

The subjects wore colored gloves for the recording (single colored gloves, with different colors for each hand). This methodology allows researchers to easily bypass the tracking and segmentation steps, and focus on the subsequent steps of the recognition [9].

While the dataset has less classes than ASLLVD, RWTH-PHOENIX-Weather or SIGNUM, it has more samples and subjects than many other datasets (Table 1), and it is publicly available with a preprocessed version.

The document is organized as follows: Section 2 describes the LSA64 dataset and the recording conditions. Section 3 presents statistics and information of the signs recorded, to aid in the understanding of the dataset. Section 4 details an experiment carried out to establish a baseline on this dataset, and Section 5 presents the general conclusions.

2 Dataset

The sign database for the Argentinian Sign Language, created with the goal of producing a dictionary for LSA and training an automatic sign recognizer, includes 3200 videos where 10 non-expert subjects executed 5 repetitions of 64 different types of signs. Signs were selected among the most common in the LSA lexicon, and include both verbs and nouns. Some examples can be seen in Figure 1.

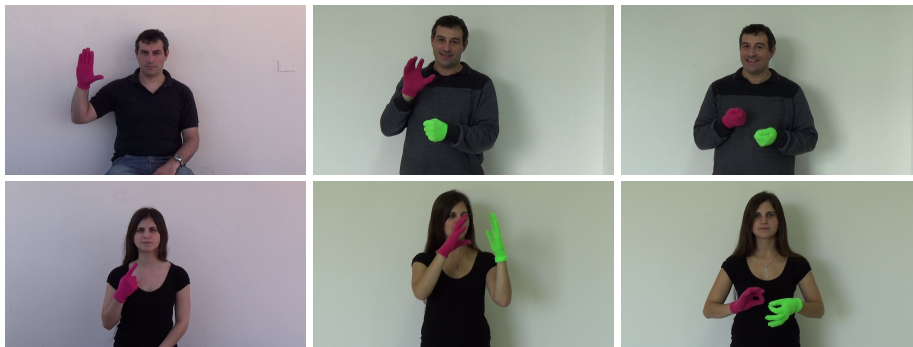


Fig. 1. Snapshots of six different signs of the LSA64 database. There are overlaps in positions and handshapes. The images on the left are from the first set of recordings.

⁴ The dataset and relevant information can be found at <http://facundoq.github.io/unlp/lisa64/>.

2.1 Recording

The database was recorded in two sets. In the first one, 23 one-handed signs were recorded. The second added 41 signs, 22 two-handed and 19 one-handed. The final dataset then contains 42 one-handed signs and 22 two-handed ones.

The first recording was done in an outdoors environment, with natural lighting, while the second took place indoors, with artificial lighting (Figure 1). Subject 10 from the first recordings was unavailable for the second set of recordings, and was replaced by another subject. This change in no way diminishes the utility of the dataset, since the set of classes recorded in the first session is disjoint from the ones recorded in the second session.

In both sets of recordings, subjects wore black clothes and executed the signs standing or sitting, with a white wall as a background. To simplify the problem of hand segmentation within an image, subjects wore fluorescent-colored gloves. These substantially simplify the problem of recognizing the position of the hand and performing its segmentation, and remove all issues associated to skin color variations, while fully retaining the difficulty of recognizing the handshape. Additionally, each sign was executed imposing few constraints on the subjects to increase diversity and realism in the database. The camera employed was a Sony HDR-CX240. The tripod was placed 2m away from the wall at a height of 1.5m.

In the following subsections we show statistics and information of the signs to better understand the nature and challenges of the dataset. These statistics show that signs in this dataset possess significant overlap in terms of types of movements, initial and final positions and handshapes, producing non-trivial experiment settings to test new sign language recognition models. All the information has been computed from the pre-processed version of the dataset described in Section 3.

2.2 Handshapes

In Figures 2 and 3 we can observe the different handshapes of the right and left hand respectively for each class of sign. There is plenty of repetition between handshapes, although their 2D projection may be different depending on the rotation of the hand.

2.3 Positions

Figure 4 presents the mean initial and final positions for each hand, along with the covariance. While a few signs can be identified by their positions, they overlap significantly in most cases.

2.4 Trajectories

Figure 5 shows sample trajectories of each sign, as performed by subject 2. There is much overlap in movements for both one-handed (for example, signs 1, 5, 7, 13 and 19) and two-handed signs (for example, signs 31, 32 and 61).

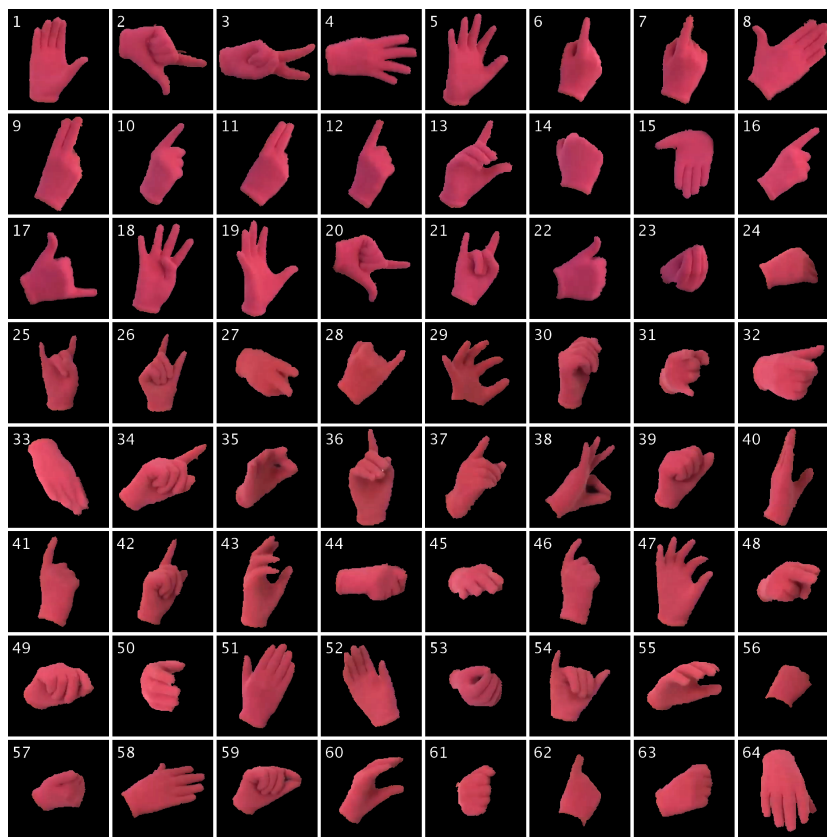


Fig. 2. Images of segmented hands as captured in the LSA64 database. Each image shows the initial handshape of the right hand for each sign in the dataset.

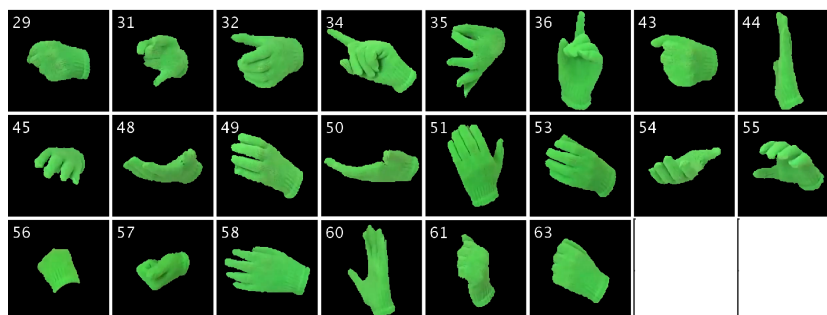


Fig. 3. Images of segmented hands as captured in the LSA64 database. Each image shows the initial handshape of the left hand for the two-handed signs of the dataset.

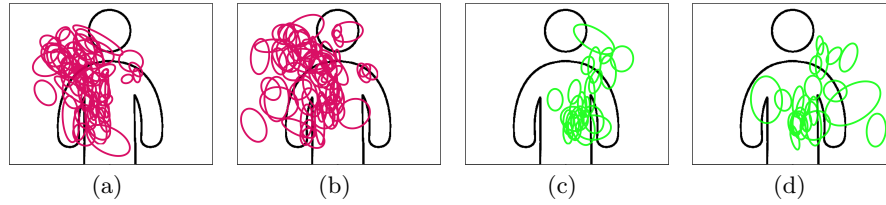


Fig. 4. Means for the initial and final positions of the right hand for each sign (a and b), and also for the left hand (c and d). Circles around means represent the covariance of the samples.

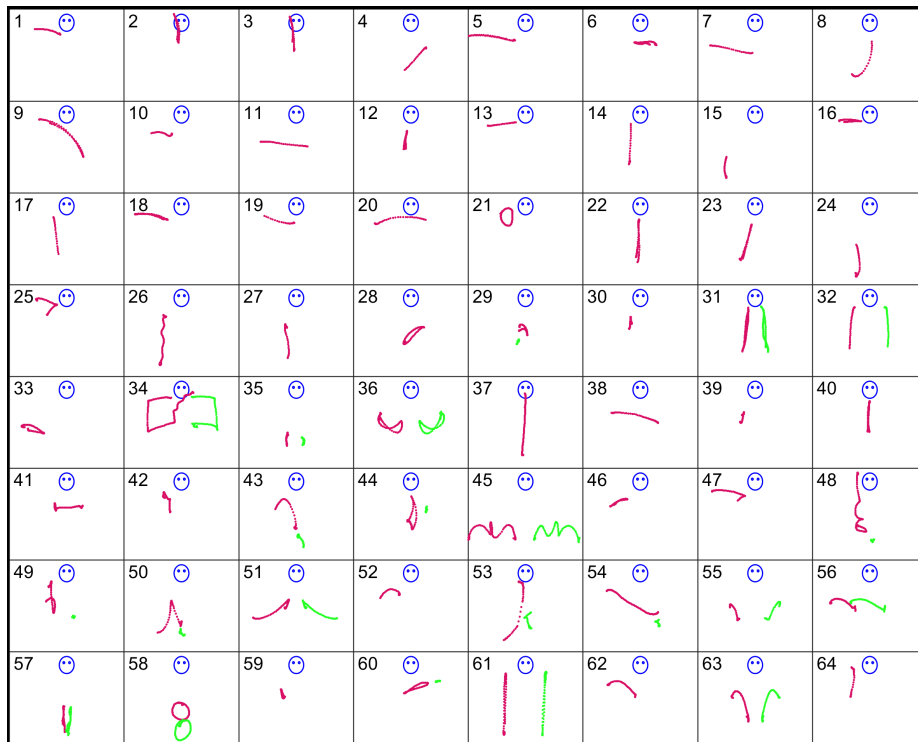


Fig. 5. Sample trajectories for each sign in LSA64. The left-hand trajectory is shown in light green, the right-hand one in red, and the head position as a blue circle.

2.5 Amount of movement

Figure 6 shows the amount of movement for each hand, measured as the maximum distance between two points in the trajectory of the hand. The movement in the left hand is significantly smaller than that of the right hand in many signs, consistent with the fact that the right hand is the dominant one for all the signers.

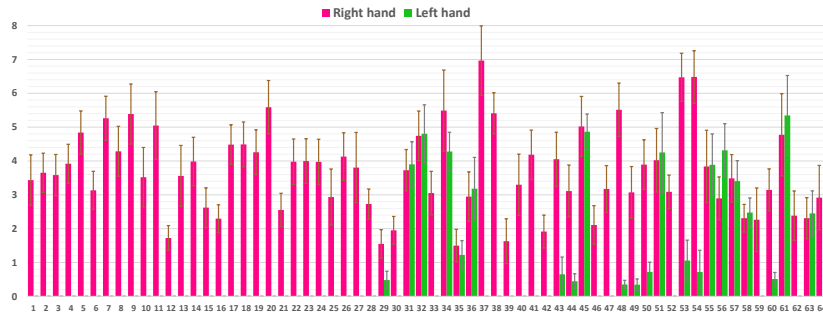


Fig. 6. Amount of movement for each class of sign. Red bars show the amount of movement of the right hand, and green bars the movement of the left hand (in two-handed signs).

3 Preprocessed version

We provide a pre-processed version of the dataset to alleviate the overhead of performing experiments with the data.

From the dataset we extracted the hand and head positions for each frame, along with images of each hand, segmented and with a black background, as show in Figure 7.

The tracking and segmentation of the hands uses the techniques described in [5]. Additionally, the head of the subject is tracked via the Viola-Jones's face detector [7]. The 2D position of each hand is translated so that the head is at the origin. The positions are then normalized by dividing by the arm's length of the subject, measured in centimeters/pixels. In this way, the transformed positions represent distances from the head, in units of centimeters.

The result of this process is a sequence of frame information, where for each frame we calculate the position of both hands, and we extract an image of each hand with the background segmented.

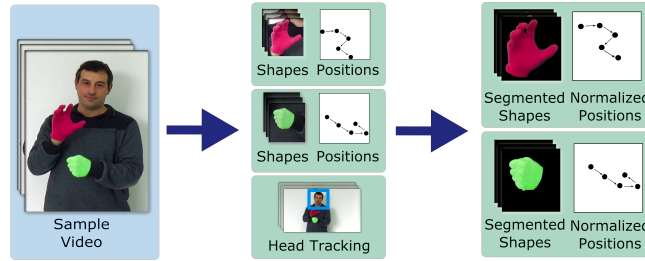


Fig. 7. Feature extraction steps for the preprocessed version of LSA64. From the sample video, we track the position of both hands and head in all frames. The shapes of each hand are segmented and the positions of them are re-centered with respect to those of the head.

4 Baseline Experiments

In this section, we briefly describe the model and results obtained in signer-dependent and independent experiments with the dataset, to establish a baseline performance. The model we used is described fully in [4].

The model we used to get a baseline performance on the dataset classifies the information for each hand separately and then multiplies the probabilities output by the subclassifiers, per class. The model for each hand includes three subclassifiers, each processing position, movement or handshape information.

The movement subclassifier contains one left-to-right Hidden Markov Model (HMM) per class, with skip transitions. All the models have 4 states. The output probabilities are modeled with a Gaussian Mixture Model (GMM) in each state. The models are trained with EM with the trajectories computed in the pre-processed version of the dataset.

The handshape subclassifier also employs HMM-GMMs, but uses as input the output of the static handshape classifier described in [5]. The static classifier is run on the segmented hand for each frame of the video, and the sequence of probabilities is fed into the handshape HMM-GMM to obtain the probability of each class for the whole sequence of frames.

The position subclassifier models the initial and final positions of the signs of each class with a set of gaussian distributions. There are two gaussian per class, one for the initial and another one for the final position.

For each class, the model outputs the product of the probabilities given by the position, movement and handshape information of both hands. For one-handed gestures, the information of the left hand is ignored, and so the probabilities output by the left hand model are not multiplied.

We performed a subject-dependent classification experiment with the model, using stratified repeated random sub-sampling validation as the cross-validation scheme, with 30 runs and an 80-20 training-test split. For each run, we measured the classification accuracy of the model on the test dataset. The mean accuracy obtained was 95.95% (standard deviation $\sigma = 0.954$).

5 Conclusion

We have presented a dataset of signs from the Argentinian Sign Language. To the best of our knowledge, there are currently no research-oriented datasets of this language created or available.

The subjects used colored gloves in the recording of the signs to significantly ease the tracking and segmentation steps. Nonetheless, we also provide a pre-processed version of the dataset to facilitate experimentation and reproducibility.

We have also presented a set of statistics and extra information to characterize the dataset and allow researchers to easily understand its nature. The signs in this dataset possess significant overlap in terms of types of movements, initial and final positions and handshapes, producing non-trivial experiment settings to test new sign language recognition models.

We intend to expand the dataset with both new signs and a set of annotated LSA sentences to provide a complete basic working vocabulary for Argentinian Sign Language.

References

1. Cooper, H., Holt, B., Bowden, R.: Sign language recognition. In: Moeslund, T.B., Hilton, A., Krüger, V., Sigal, L. (eds.) *Visual Analysis of Humans: Looking at People*, chap. 27, pp. 539 – 562. Springer (Oct 2011)
2. Kapuscinski, T., Oszust, M., Wysocki, M., Warchol, D.: Recognition of hand gestures observed by depth cameras. *International Journal of Advanced Robotic Systems* 12 (2015)
3. Neidle, C., Thangali, A., Sclaroff, S.: Challenges in development of the american sign language lexicon video dataset (asllvd) corpus. In: *Proc. 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon*. Citeseer (2012)
4. Ronchetti, F., Quiroga, F., Estrebou, C., Lanzarini, L., Rosete-Suárez, A.: Sign language recognition without frame-sequencing constraints: A proof of concept on the argentinian sign language. *IBERAMIA: Iberoamerican Society of Artificial Intelligence* (2016)
5. Ronchetti, F., Quiroga, F., Lanzarini, L., Estrebou, C.: Handshape recognition for argentinian sign language using probsom. *Journal of Computer Science and Technology* 16(1), 1–5 (2016)
6. Roussos, A., Theodorakis, S., Pitsikalis, V., Maragos, P.: Hand tracking and affine shape-appearance handshape sub-units in continuous sign language recognition. In: *Trends and Topics in Computer Vision - ECCV 2010 Workshops*, Heraklion, Crete, Greece, Revised Selected Papers, PartI. pp. 258–272 (2010)
7. Viola, P., Jones, M.J.: Robust real-time face detection. *International journal of computer vision* 57(2), 137–154 (2004)
8. Von Agris, U., Zieren, J., Canzler, U., Bauer, B., Kraiss, K.F.: Recent developments in visual sign language recognition. *Universal Access in the Information Society* 6(4), 323–362 (2008)
9. Wang, R.Y., Popović, J.: Real-time hand-tracking with a color glove. *ACM transactions on graphics (TOG)* 28(3), 63 (2009)

Distal Dynamic Spatial Approximation Forest

Edgar Chávez¹, María E. Di Genaro², Nora Reyes², and Patricia Roggero²

¹ Centro de Investigación Científica y de Educación Superior de Ensenada, México
elchavez@cicese.mx

² Departamento de Informática, Universidad Nacional de San Luis, Argentina
{mdigena,nreyes,proggero}@unsl.edu.ar

Abstract. Querying large datasets by proximity, using a distance under the metric space model, has a large number of applications in multimedia, pattern recognition, statistics, etc. There is an ever growing number of indexes and algorithms for proximity querying, however there is only a handful of indexes able to perform well without user intervention to select parameters. One of such indexes is the *Distal Spatial Approximation Tree* (DiSAT) which is parameter-less and has demonstrated to be very efficient outperforming other approaches. The main drawback of the DiSAT is its static nature, that is, once built, it is difficult to add or to remove new elements. This drawback prevents the use of the DiSAT for many interesting applications.

In this paper we overcome this weakness. We use a standard technique, the Bentley and Saxe algorithm, to produce a new index which is dynamic while retaining the simplicity and appeal for practitioners of the DiSAT. In order to improve the DiSAF performance, we do not attempt to directly apply the Bentley and Saxe technique, but we enhance its application by taking advantage of our deep knowledge of the DiSAT behavior.

Keywords: similarity search, dynamism, metric spaces, non-conventional databases

1 Introduction

The metric space approach has become popular in recent years to handle the various emerging databases of complex objects, which can only be meaningfully searched for by similarity [3, 11, 12, 5]. This approximation has applications in a vast number of fields. Some examples are non-traditional databases, text searching, information retrieval, machine learning and classification, image quantization and compression, computational biology, and function prediction. These problems can be mapped into a *metric space model* [3] as a metric database. That is, there is a universe \mathbb{X} of objects, and a non negative real valued distance function $d : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}^+ \cup \{0\}$ defined among them. This distance satisfies the three axioms that make the set a *metric space*: *strict positiveness* ($d(x, y) \geq 0$ and $d(x, y) = 0 \Leftrightarrow x = y$), *symmetry* ($d(x, y) = d(y, x)$), and *triangle inequality* ($d(x, z) \leq d(x, y) + d(y, z)$). We have a finite *database* $\mathbb{U} \subseteq \mathbb{X}$, $|\mathbb{U}| = n$, which is a subset of the universe.

Thereby, “proximity” or “similarity” searching is the problem of looking for objects in a dataset, that are “close” or “similar enough” to a given query object, under a certain (expensive to compute in time and/or resources) distance. The smaller the distance between two objects, the more “similar” they are. The database can be preprocessed to

build a *metric index*, that is, a data structure to speed up similarity searches. There are two typical similarity queries: *range queries* and *k-nearest neighbors queries*.

A large number of metric indices have flourished [3, 12, 11]. The *Distal Spatial Approximation Tree* (DiSAT) is an index based on dividing the search space and then approaching the query spatially. DiSAT is algorithmically interesting by itself, it has been shown that it gives an attractive trade-off between memory usage, construction time, and search performance. Besides, the great advantage of DiSAT compared to other indices is that it does not require any parameter tuning. However, DiSAT is a static index, that is, the index has to be rebuilt from scratch or it requires an expensive updating when the set of indexed objects undergoes insertions or deletions.

Although for some applications a static scheme may be acceptable, many relevant ones do require dynamic capabilities. Actually, in many cases it is sufficient to support insertions, such as in digital libraries and archival systems, versioned and historical databases, and several other scenarios where objects are never updated or deleted. In this paper we introduce a new dynamic version of DiSAT, by using the *Bentley-Saxe method* (BS)[1]. This method allows to transform a static index into a dynamic one, if on this index the search problem is *decomposable*. In [8] some static indexes are analyzed in combination with the BS method, obtaining certain acceptable results, but DiSAT in a static scenario has shown to outperform all these index. Now, we are focused only on supporting insertion and range searches, and we left deletions, *k*-NN searches and other improvements as future works.

The rest of this paper is organized as follows. In Section 2 we describe some basic concepts, and the BS method. Next, in Section 3 we detail the *Distal Spatial Approximation Trees* (DiSAT), and some notions of its close relatives: *Spatial Approximation Trees* (SAT) and the *Dynamic Spatial Approximation Trees* (DSAT). Section 4 introduces our dynamic variant of DiSAT. In Section 5 we show the experimental evaluation of our proposal. Finally, we draw some conclusions and future work directions in Section 6.

2 Previous Concepts

The metric space model can be formalized as follows. Let \mathbb{X} be a universe of *objects*, with a nonnegative *distance function* $d : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}^+$ defined among them. This distance satisfies the three axioms that make (\mathbb{U}, d) a *metric space*: strict positiveness ($d(x, y) = 0 \Leftrightarrow x = y$), symmetry ($d(x, y) = d(y, x)$) and triangle inequality ($d(x, z) \leq d(x, y) + d(y, z)$). We handle a finite *dataset* $\mathbb{U} \subseteq \mathbb{X}$, which can be preprocessed (to build an index). Later, given a new object from \mathbb{X} (a *query* $q \in \mathbb{X}$), we must retrieve all similar elements found in \mathbb{U} . There are two typical queries of this kind:

Range query: Retrieve all elements in \mathbb{U} within distance r to q .

k-nearest neighbors query (k-NN): Retrieve the k closest elements to q in \mathbb{U} .

In this paper we are devoted to range queries. Nearest neighbor queries can be rewritten as range queries in an optimal way [6, 7], so we can restrict our attention to range queries. The distance is assumed to be expensive to compute. Hence, it is customary to define the complexity of the search as the number of distance evaluations performed, disregarding other components such as CPU time for side computations, and even I/O

time. Given a dataset of $|\mathbb{U}| = n$ objects, queries can be trivially answered by performing n distance evaluations.

There exist a number of methods to preprocess the database in order to reduce the number of distance evaluations. (See [11, 12, 3] for more complete surveys.) Most of those structures work on the basis of discarding elements using the triangle inequality, and most use the classical divide-and-conquer approach. Algorithms to search in general metric spaces can be divided into two large areas: *pivot-based* and *clustering* algorithms. However, there are also algorithms that combine ideas from both areas.

Bentley and Saxe Method

The Bentley-Saxe method allows to transform a static index into a dynamic one, if on this index the search problem is *decomposable*, based on the binary representation of the integers [1]. A search problem with a query operation \mathcal{Q} is *decomposable* if there exists an efficiently computable binary operator \square satisfying the condition:

$$\mathcal{Q}(q, \mathbb{X}_1 \cup \mathbb{X}_2) = \square[\mathcal{Q}(q, \mathbb{X}_1), \mathcal{Q}(q, \mathbb{X}_2)]$$

where the \square operation has to be associative and commutative [1, 8]. That is, the answer to a query on a dataset $\mathbb{X}_1 \cup \mathbb{X}_2$ has to be computed efficiently from the answer to a queries for each \mathbb{X}_1 and \mathbb{X}_2 . In the particular case of range queries on \mathbb{X} , the \square operation is the union of the sets obtained with the query operation \mathcal{Q} .

The main idea of BS method is to partition the indexed set \mathbb{X} in certain subsets $\mathbb{X}_0, \mathbb{X}_1, \mathbb{X}_2, \dots, \mathbb{X}_m$ (if $|\mathbb{X}| = n, m = \lfloor \log n \rfloor$) to reduce the size of the index of each subset that need to be rebuilt when an object is inserted or deleted [8]. This partition satisfies that $\bigcup_{0 \leq i \leq m} \mathbb{X}_i = \mathbb{X}$ and $\mathbb{X}_i \cap \mathbb{X}_j = \emptyset$ for $i \neq j$, and $|\mathbb{X}_i| = 2^i$. Then, the main data structure of BS is composed by a set of data structures T_0, T_1, \dots, T_m , where T_i is an empty data structure if $\mathbb{X}_i = \emptyset$, otherwise T_i is a static data structure that contains 2^i objects. Observe that for any value of n , there is a unique collection of subsets that must be non-empty. When a new object is inserted into the index, the algorithm proceeds with the same principle used for incrementing a binary counter. At query time, the search is solved independently by searching on each non-empty T_i and then the results of all individual searches are combined.

3 Distal Spatial Approximation Trees

The *Distal Spatial Approximation Tree* (DiSAT) [2] is a variant of the *Spatial Approximation Tree* (SAT) [9], both are data structures aiming at approaching the query spatially by starting at the root and getting iteratively closer to the query navigating the tree. In both cases the trees are built as follows. An element a is selected as the root, and it is connected to a set of *neighbors* $N(a)$, defined as a subset of elements $x \in \mathbb{U}$ such that x is closer to a than to any other element in $N(a)$. The other elements (not in $N(a) \cup \{a\}$) are assigned to their closest element in $N(a)$. Each element in $N(a)$ is recursively the root of a new subtree containing the elements assigned to it. For each node a the covering radius is stored, that is, the maximum distance $R(a)$ between a and any element in the subtree rooted at a . The starting set for neighbors of the root a , $N(a)$ is empty. Therefore we can select *any* database element as the first neighbor. Once this element

is fixed the database is split in two halves by the hyperplane defined by proximity to a and the recently selected neighbor. Any element in the a side can be selected as the second neighbor. While the zone of the root (those database elements closer to the root than the previous neighbors) is not empty, it is possible to continue with the subsequent neighbor selection. The SAT considers the elements of $\mathbb{U} - \{a\}$ in increasing order of distance to a , but DiSAT considers exactly the opposite order.

The main difference between them is that DiSAT tries to increase the separation between hyperplanes, which in turn decreases the size of the covering radius; the two parameters governing the performance of these trees. The performance improvement consists in selecting distal nodes instead of the proximal nodes selected in the original algorithm. Considering an example of a metric database illustrated in Fig. 1, the Fig. 2 shows the SAT (Fig. 2(a)) and the DiSAT (Fig. 2(b)) obtained by selecting p_6 as the tree root. In both cases we also depict the covering radii for the neighbors of the tree root. It is possible to obtain completely different trees (SATs or DiSATs) if we select different roots, and each tree probably may have different search costs.

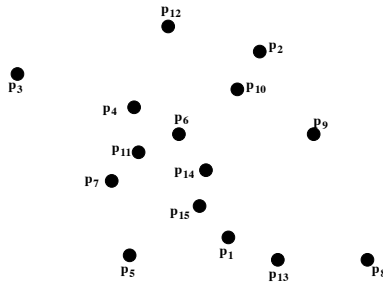


Fig. 1. Example of a metric database in \mathbb{R}^2 .

Algorithm 1 gives a formal description of the construction of DiSAT. Range searching is done with the procedure described in Algorithm 2. This process is invoked as $\text{RangeSearch}(a, q, r, d(a, q))$, where a is the tree root, r is the radius of the search, and q is the query object. One key aspect of DiSAT (SAT too) is that a greedy search will find all the objects previously inserted. For a range query of q with radius r , and being c the closest element between $\{a\} \cup N(a) \cup A(a)$ and $A(a)$ the set of the ancestors of a , the same greedy search is used entering all the nodes $b \in N(a)$ such that $d(q, b) \leq d(q, c) + 2r$ because any element $x \in (q, r)_d$, can differ from q by at most r at any distance evaluation, so it could have been inserted inside any of those b nodes [12, 9]. In the process, all the nodes x founded close enough to q are reported.

Dynamic Spatial Approximation Tree

The *Dynamic Spatial Approximation Tree* (DSAT) [10] is an online version of the SAT. It is designed to allow dynamic insertions and deletions without increasing the construction cost with respect to the SAT. A very surprising and unintended feature of the DSAT is the boosting in the searching performance. The DSAT is faster in searching even if at construction it has less information than the static version of the index. For the DSAT

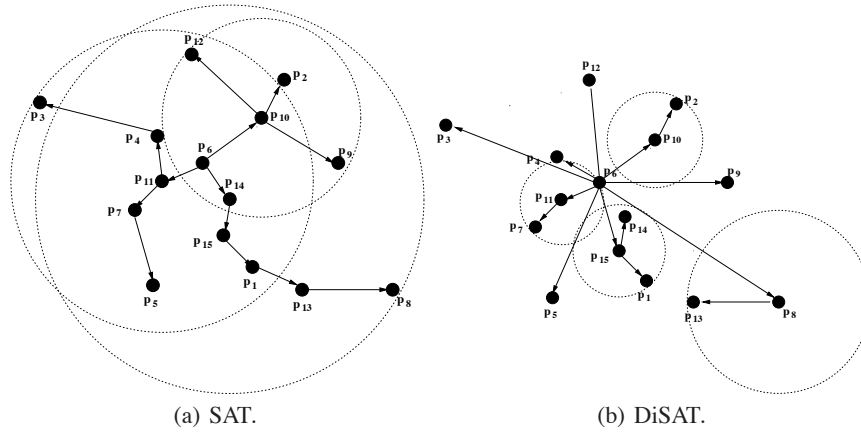


Fig. 2. Example of the SAT and DiSAT obtained if p_6 were the root.

the database is unknown beforehand and the objects arrive to the index at random as well as the queries. A dynamic data structure cannot make strong assumptions about the database and will not have statistics about all the database.

4 Our Proposal: Distal Dynamic Spatial Approximation Forest

As we mention previously, the BS method can be applied on any static data structure to transform it into a dynamic one. We select the DiSAT because it has shown that is a very competitive index and it do not need to set any parameter. Most of the more efficient indexes that need the setting of the value of at least one parameter can become inefficient at a bad choice of it.

In this particular case each T_i that considers the BS method is a tree, particularly a DiSAT, so our new dynamic data structure is named *Distal Dynamic Spatial Approximation Forest* (DiSAF), because we have a *forest* of DiSATs. The i -th DiSAT in the forest will have 2^i elements.

Considering the example illustrated in Fig. 1, the Fig. 3 and Fig. 4 illustrate the two dynamic data structures, based on spatial approximation, obtained by inserting the objects p_1, \dots, p_{15} one by one: DSAT with maximum arity of 6 (Fig. 3) and DiSAF (Fig. 4). In the DSAT the root will be p_1 , because it is the first element arrived. On the other hand, as we have 15 elements, DiSAF will build four DiSATs: T_0, T_1, T_2 , and T_3 . As it is aforementioned, each T_i will have 2^i elements. As the insertion order is from p_1 to p_{15} , the final situation will have: T_0 with the dataset $\{p_{15}\}$, T_1 with $\{p_{13}, p_{14}\}$, T_2 with $\{p_9, \dots, p_{12}\}$, and T_3 with $\{p_1, \dots, p_8\}$. We also depict the covering radii for the neighbors of the tree roots, some covering radii are equal to zero. On one hand, it is possible to obtain different DSATs if we consider different maximum arities or different

Algorithm 1 Process to build a DiSAT for $\mathbb{U} \cup \{a\}$ with root a .

BuildTree(Node a , Set of nodes U)

1. $N(a) \leftarrow \emptyset$ /* neighbors of a */
2. $R(a) \leftarrow 0$ /* covering radius */
3. **For** $v \in U$ in increasing distance to a **Do**
4. $R(a) \leftarrow \max(R(a), d(v, a))$
5. **If** $\forall b \in N(a), d(v, a) < d(v, b)$ **Then**
6. $N(a) \leftarrow N(a) \cup \{v\}$
7. **For** $b \in N(a)$ **Do** $S(b) \leftarrow \emptyset$
8. **For** $v \in U - N(a)$ **Do**
9. $c \leftarrow \operatorname{argmin}_{b \in N(a)} d(v, b)$
10. $S(c) \leftarrow S(c) \cup \{v\}$
11. **For** $b \in N(a)$ **Do** **BuildTree**($b, S(b)$)

Algorithm 2 Searching of q with radius r in a DiSAT with root a .

RangeSearch(Node a , Query q , Radius r , Distance d_{min})

1. **If** $d(a, q) \leq R(a) + r$ **Then**
2. **If** $d(a, q) \leq r$ **Then Report** a
3. $d_{min} \leftarrow \min \{d(c, q), c \in N(a)\} \cup \{d_{min}\}$
4. **For** $b \in N(a)$ **Do**
5. **If** $d(b, q) \leq d_{min} + 2r$ **Then**
6. **RangeSearch**(b, q, r, d_{min})

insertion orders, and they will likely have different search costs. On the other hand, as DiSAF has not any parameter, the only way to obtain different forests is by considering different insertion orders.

The insertion process of a new element x in a DiSAF is described in the Algorithm 3. Initially, the DiSAF has an only DiSAT $T_0 = null$. Then, the index can be built via successive insertions. **Retrieve**(Tree T) return all the elements that compose the tree T . The range search process is detailed in the Algorithm 4.

5 Experimental Results

For the empirical evaluation of the indices we consider three widely different metric spaces from the SISAP Metric Library (www.sisap.org) [4].

Dictionary: a dictionary of 69,069 English words. The distance is the *edit distance*, that is, the minimum number of character insertions, deletions and substitutions needed to make two strings equal. This distance is useful in text retrieval to cope with spelling, typing and optical character recognition (OCR) errors.

Color Histograms: a set of 112,682 8-D color histograms (112-dimensional vectors) from an image database³. Any quadratic form can be used as a distance; we chose Euclidean as the simplest meaningful distance.

NASA images: a set of 40,700 20-dimensional feature vectors, generated from images downloaded from NASA⁴. The Euclidean distance is used.

³ At <http://www.dbs.informatik.uni-muenchen.de/~seidl/DATA/histo112.112682.gz>

⁴ At <http://www.dimacs.rutgers.edu/Challenges/Sixth/software.html>

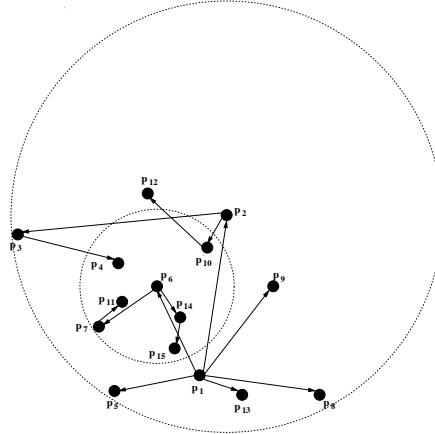


Fig.3. Example of the DSAT with maximum arity of 6, inserting from p_1 to p_{15} .

Algorithm 3 Insertion of a new element x in a DiSAF with at most m trees.

Insert(Element x)

1. $S \leftarrow \emptyset$, $k \leftarrow \min_{0 \leq i \leq m} i$, such that $T_i = null$
2. For i from 0 to $k-1$ Do
3. $S \leftarrow S \cup \mathbf{Retrieve}(T_i)$ /* retrieve all the elements of T_i */
4. $T_i \leftarrow null$ /* T_i is a new empty tree */
5. $T_k \leftarrow \mathbf{BuildTree}(x, S)$
6. If $k = m$ Then
7. $T_{k+1} \leftarrow null$, $m \leftarrow k + 1$

When we evaluate construction costs, we build the index with the complete database. If the index is dynamic, the construction is made by inserting one by one the objects, otherwise the index knows all the elements beforehand. In order to evaluate the search performance of the indexes, we build the index with the 90% of the database elements and we use the remaining 10%, randomly selected, as queries. So, the elements used as query objects are not in the index. We average the search costs of all these queries. All results are averaged over 10 index constructions with different datasets permutations.

We consider range queries retrieving on average 0.01%, 0.1% and 1% of the dataset. This corresponds to radii 0.051768, 0.082514 and 0.131163 for the Color Histograms; and 0.605740, 0.780000 and 1.009000 for the NASA images. The Dictionary have a discrete distance, so we used radii 1 to 4, which retrieved on average 0.00003%, 0.00037%, 0.00326% and 0.01757% of the dataset, respectively. The same queries were used for all the experiments on the same datasets. As we mention previously, given the existence of range-optimal algorithms for k -nearest neighbor searching [6, 7], we have not considered these search experiments separately.

We show the comparison between our dynamic DiSAF, the DSAT, and the static alternatives SAT and DiSAT. The source code of the different SAT versions (SAT and DSAT) is available at www.sisap.org. A final note in the experimental part is the arity parameter of the DSAT which is tunable and is the maximum number of neighbors

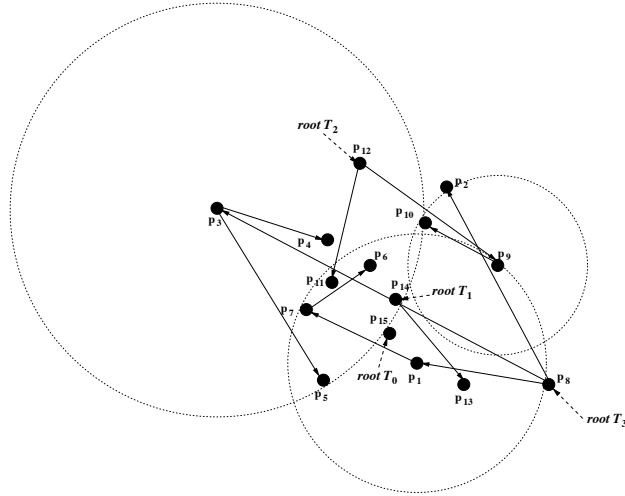


Fig.4. Example of the DiSAF, inserting from p_1 to p_{15} .

Algorithm 4 Searching of q with radius r in a DiSAF with at most m trees.

RangeSearchNew(Query q , Radius r)

1. $A \leftarrow \emptyset$
2. For i from 0 to $m - 1$
3. If $T_i \neq \text{null}$ Then
4. Let x be the root of T_i
5. $A \leftarrow A \cup \text{RangeSearch}(x, q, r, d(x, q))$
6. Report A

of each node of the tree. In our experiments we used the arity suggested by authors in [10]. The Figure 5 illustrates the construction costs of the all indices, on the three metric spaces. As it can be seen, DiSAF is surpassed for the other three indexes, because it has to rebuild the trees too many times. On the other hand, DSAT do not make any reconstruction while it builds the tree via insertions. It has to be considered that SAT and DiSAT are built with all the elements known at the same time, not dynamically.

We analyze search costs in Figure 6. As it can be noticed, DiSAF surpasses DSAT in most of spaces. The only index that is always better than DiSAF is the DiSAT, but as we already mention it is static. Therefore, we can affirm that the heuristic of construction of DiSAT allows to surpass in searches the other strategies used in SAT and DSAT. Besides, we have obtained a dynamic index that overcomes DSAT at searches and it do not have any parameter to tune.

6 Conclusions

We have presented a dynamic version of the DiSAT, which at this time is able of handling insertions without affecting significantly its search quality. Very few data structures for searching metric spaces are dynamic. Furthermore, we have shown that the

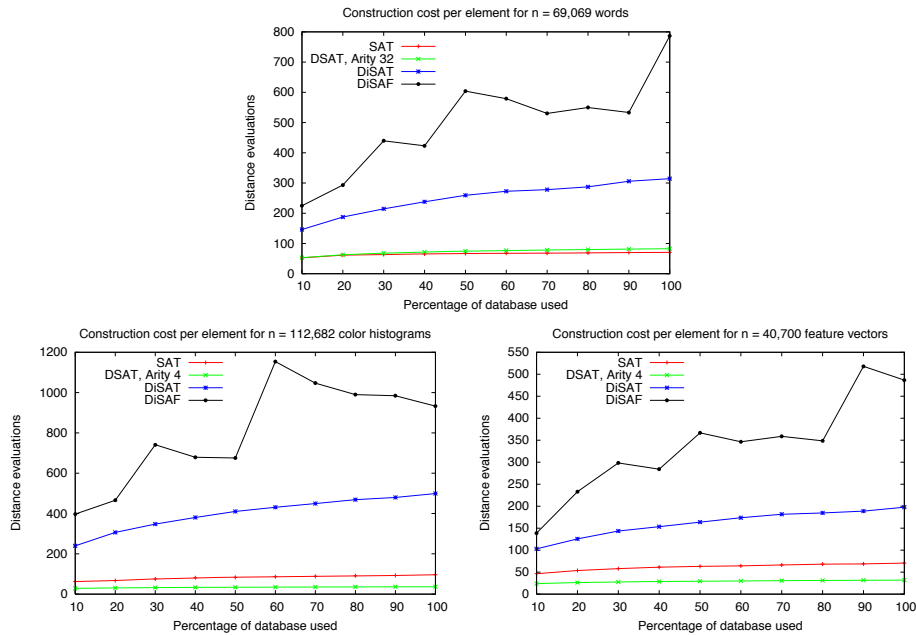


Fig. 5. Construction costs for the three metric spaces considered.

heuristic used in DiSAT and DiSAF to partition the metric space is better than that used in SAT and DSAT: distal nodes produce more compact subtrees, which in turn give more locality to the underlying partitions implicitly defined by the subtrees.

The DiSAT was a promising data structure for metric space searching, with several drawbacks that prevented it from being practical: high construction cost and inability to accommodate insertions and deletions. We have addressed one of these weaknesses. Despite of we have obtained worse construction costs, it is still possible to improve it, for example by providing a bulk-loading algorithm to create initially the DiSAF if we know beforehand a subset of elements, avoiding unnecessary rebuildings when we insert elements one by one, or with *lazy insertion* that do not always rebuild trees.

We are currently pursuing in the direction of making the DiSAF fully dynamic; that is, that it also supports deletions, and designing an efficient bulk-loading algorithm, which amortizes the insertion costs between several elements. Other topic of future work is to design a more efficient alternative of k -NN search that do not apply the basic solution of decomposable search, but that it applies a smart solution by taking advantage of all distances calculated in order to shrink, as soon as possible, the radius from q that encloses k elements.

References

1. Jon L. Bentley and James B. Saxe. Decomposable searching problems i. static-to-dynamic transformation. *Journal of Algorithms*, 1(4):301–358, 1980.
2. Edgar Chávez, Verónica Ludeña, Nora Reyes, and Patricia Roggero. Faster proximity searching with the distal sat. *Information Systems*, 59:15 – 47, 2016.

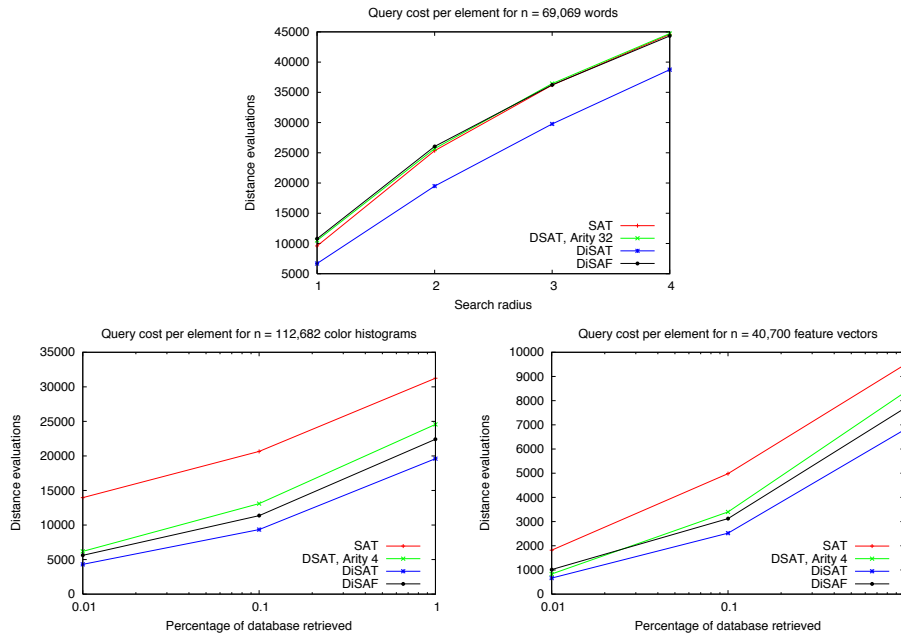


Fig. 6. Search costs for the three metric spaces considered.

3. Edgar Chávez, Gonzalo Navarro, Ricardo Baeza-Yates, and José Luis Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, September 2001.
4. Karina Figueroa, Gonzalo Navarro, and Edgar Chávez. Metric spaces library, 2007. Available at http://www.sisap.org/Metric_Space_Library.html.
5. Magnus Hetland. The basic principles of metric indexing. In Carlos Coello, Satchidananda Dehuri, and Susmita Ghosh, editors, *Swarm Intelligence for Multi-objective Problems in Data Mining*, volume 242 of *Studies in Computational Intelligence*, pages 199–232. Springer Berlin / Heidelberg, 2009.
6. Gísli R. Hjaltason and Hanan Samet. *Incremental Similarity Search in Multimedia Databases*. Number CS-TR-4199 in Computer science technical report series. Computer Vision Laboratory, Center for Automation Research, University of Maryland, 2000.
7. Gísli R. Hjaltason and Hanan Samet. Index-driven similarity search in metric spaces. *ACM Transactions on Database Systems*, 28(4):517–580, 2003.
8. Bilegsaikhan Naidan and Magnus Lie Hetland. Static-to-dynamic transformation for metric indexing structures (extended version). *Information Systems*, 45:48 – 60, 2014.
9. Gonzalo Navarro. Searching in metric spaces by spatial approximation. *The Very Large Databases Journal (VLDBJ)*, 11(1):28–46, 2002.
10. Gonzalo Navarro and Nora Reyes. Dynamic spatial approximation trees. *Journal of Experimental Algorithmics*, 12:1.5:1–1.5:68, June 2008.
11. Hanan Samet. *Foundations of Multidimensional and Metric Data Structures (The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.
12. Pavel Zezula, Giuseppe Amato, Vlatislav Dohnal, and Michal Batko. *Similarity Search: The Metric Space Approach*, volume 32 of *Advances in Database Systems*. Springer, 2006.

Propuesta de detección de datos anómalos y ruido en declaraciones juradas públicas

Rodrigo López-Pablos^{1,2} y Horacio D. Kuna³

¹Escuela de Posgrado y Formación Continua,
Universidad Nacional de La Matanza, Argentina

²Escuela de Posgrado, Facultad Regional Buenos Aires,
Universidad Tecnológica Nacional, Argentina

³Departamento de Informática, Facultad de Ciencias Exactas, Químicas y Naturales,
Universidad Nacional de Misiones, Posadas, Argentina
{rodrigo.lopezpablos,hkuna}@gmail.com

Resumen. Los procesos de detección de campos anómalos y con ruido son de suma utilidad para la evaluación de calidad de bases de datos de todo tipo. Estos procesos pueden tener una utilidad cívica y pública inédita si se encuentran dirigidos a la detección de valores anómalos en datos públicos. En este trabajo, se propone investigar la posibilidad de experimentación, validación y aplicación de procedimientos híbridos de detección de datos anómalos y ruido en sistemas de declaraciones juradas oficiales disponibles actualmente en Argentina.

Palabras clave: Datos anómalos y ruido, datos públicos, funcionarios públicos, declaraciones juradas, bases de datos, outliers.

1 Introducción

Los procesos de minería de datos y explotación de la información han sido escasamente usados para la resolución de problemas cívicos vinculados al sector público, los procesos de detección de datos anómalos y ruido no son una excepción en este sentido. Los datos públicos, como cualquier tipo de datos, pueden encontrarse sujetos a anomalías y ruido como podría ser el caso de cualquier base de datos (BBDD) considerada, solo que las implicancias del descubrimiento de comportamiento corrupto en datos públicos de funcionarios públicos (FFPP) y la calidad de su confección poseen efectos profundos y de relevancia que podrían impactar en el bienestar societario dado que la corrupción condiciona el tejido social y la calidad de vida de las poblaciones. Como herramienta, la utilidad de la minería de datos para echar luz en el descubrimiento del tejido social corrupto se dirige

especialmente al ciudadano cívico y las organizaciones civiles que luchan contra la corrupción, reivindicando la utilidad de la minería de datos como herramienta en ciencias sociales, en el monitoreo y la investigación científica [1].

En la Subsección 1.1 se plantean las preguntas hipotéticas que hacen a este trabajo, la Sección 2 hace al estado de la cuestión de las técnicas de minería de datos dirigidas al comportamiento corrupto; mientras que en la Sección 3 se estudian los sistemas de DDJJ abiertas, presentándose en la Sección 4 los procedimientos de detección de anomalías propuesto, su experimentación en la Sección 5 para finalmente concluir en la Sección 6.

1.1. Preguntas de investigación

En esta propuesta de investigación se plantean las siguientes preguntas:

- ¿Es factible la experimentación, aplicación y uso de técnicas y procesos de detección de datos anómalos sobre bases de datos públicas de DDJJ oficiales como herramienta para encontrar indicios de comportamiento corrupto en la administración pública y lucha contra la corrupción?
- ¿Es factible evaluar la calidad de tales bases de datos mediante análisis de ruido y anomalías?
- ¿Qué procedimientos de detección de datos anómalos y ruido pueden ser aplicados considerando el carácter de aquellos sistemas?

2. Estado del Conocimiento

Las técnicas y proceso de minería de datos y explotación de la información han sido escasa o nulamente usados como herramienta cívica de apoyo en la lucha contra la corrupción desde la óptica pública; no obstante, encontramos literatura relevante sobre técnicas de minería dirigida a la detección de fraude o corrupción privada, financiera y contable.

Una clasificación de su uso contra la corrupción privada, la cual se manifiesta en la forma de diversos fraudes financieros y contables, según varios autores [2,3] puede apreciarse una clasificación exhaustiva de estos últimos los cuales pueden comprender fraudes bancarios, tarjetas de crédito, blanqueo o lavado de dinero y fraude hipotecario, casos de corrupción privada compleja fraudes financieros complejos que involucran la falsificación de información corporativa, la especulación financiera, etc. A continuación una clasificación de las técnicas usadas en el campo.

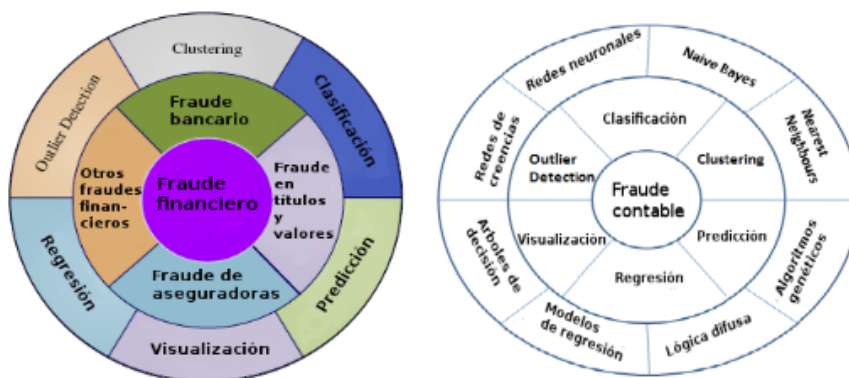


Fig. 1. A la izquierda, la clasificación de técnicas de explotación de la información para la detección de fraude financiero [2]; y a la derecha, los algoritmos y técnicas de minería de datos para el descubrimiento de fraude contable o fiscal [3].

De la figura 1 se advierte que la utilización de estas herramientas ha sido subexplotada en el ámbito público tanto para el aprovechamiento cívico como contra la corrupción dado que no encontramos antecedentes ni literatura relevante que demuestre interés por el problema de la corrupción pública.

Así mismo, sin desmedro de las restantes técnicas de minería de datos y procesos de explotación de la información, las técnicas de detección de outliers o valores anómalos no ha sido implementada hasta ahora para la solución escenarios de fraude, y por supuesto mucho menos aún, dentro del estudio de la corrupción en el ámbito público.

3. Los sistemas de DDJJ abiertas como solución optativa en la lucha contra la corrupción pública

En Argentina, los sistemas de DDJJ son sistemas de información que se encuentran regulados por sistemas o regímenes de DDJJ; los cuales, poseen tres funciones básicas para lo cual fueron implementados [4]:

- [i] Controlar la evolución patrimonial de los funcionarios de la función pública para prevenir enriquecimiento ilícito y otros delitos de corrupción.
- [ii] Detectar y prevenir conflictos de intereses e incompatibilidades de la función pública.
- [iii] Como mecanismo de transparencia y prevención de la corrupción pública.

En el sentido preventivo todo régimen de DDJJ de FFPP representa una herramienta que posibilita: el control del adecuado cumplimiento de las funciones públicas de los FFPP, prevenir el desvío de sus deberes éticos y corregir incumplimientos detectados.

3.1 Los sistemas y aplicativos abiertos de DDJJ

Del relevamiento exploratorio de datos públicos en Argentina, la experimentación se apoyó en el sistema interactivo de DDJJ Abiertas [5] del Diario La Nación en conjunto con las oenegés Directorio Legislativo, Poder Ciudadano y la Asociación Civil por la Libertad y la Justicia, iniciativa vigente desde 2013. De un total de 1550 DDJJ totales del sitio interactivo, 539 DDJJ son correspondientes a 99 funcionarios del poder ejecutivo, 843 DDJJ correspondientes a 313 funcionarios del poder legislativo, y 168 DDJJ correspondientes a 87 funcionarios del poder judicial; a partir de la versión actualización al 14 de abril de 2015. La estrategia de preparación de los datos para la experimentación se apoyo solamente en los bienes inmuebles de FFPP.

La BD de DDJJ abiertas presenta la siguiente estructura de atributos para cada declaración jurada de funcionario público.

```
dj.funcionario = (ddjj_id, ano, tipo_ddjj, poder, persona_id, nombre,
                ingreso, cargo, jurisdiccion, cant_acciones,
                descripcion_del_bien, destino, localidad, nombre_bien_s,
                origen, pais, porcentaje, provincia, tipo_bien_s,
                titular_dominio, vinculo, superficiem2, val_decl,
                valor_patrim)
```

La BD preparada para la experimentación presenta un total de 6627 tuplas con 24 atributos donde, – de 39 atributos iniciales de la BD originaria –, se descartaron 13 al encontrarse campos parcial o completamente vacíos, inconsistencias nominales en la imputación de los datos, así como atributos que pasaron a ser redundantes *a posteriori* de la homogeneización y estandarización ligados a los tres (3) atributos generados:

```
dj.patrimoniales (Gen)=(superficiem2, valor_patrim, val_decl).
```

Para el desarrollo experimental los valores monetarios patrimoniales en moneda extranjera fueron convertidos a p esos argentinos y actualizados por inflación (**valor_patrim**), la superficie inmobiliaria es homogeneizada a metros cuadrados (**superficiem2**) y se categorizó el valor del inmueble declarado por el funcionario (**val_decl**) de acuerdo a su valuación fiscal relativa, pudiendo ser fiscal, subfiscal, de mercado, o no declarándose valor alguno.

4. Procedimiento híbridos de detección de datos anómalos y ruido propuestos

Los campos anómalos se definen como un dato que por ser muy diferente a los demás pertenecientes a un mismo conjunto de datos [6], *i.e.*: una base de datos contenedora de tales campos, puede considerarse que fue creada por un mecanismo diferente; lo que, en el descubrimiento de tales mecanismos, radica el conocimiento latente en cada base analizada.

Recientemente, los métodos de detección híbridos – que combinan diferentes algoritmos de distintos enfoques de aprendizaje – se han revelado como procesos así como la combinación de distintos procedimientos permite detectar outliers con un nivel de confianza mayor al 60% [7]. Los métodos híbridos de detección de anomalías, poseen la ventaja de que combinan distintas técnicas y algoritmos para un mismo propósito, por ejemplo: LOF (Local Outlier Factor) y metadatos o LOF y K-Means para BBDD numéricas así como algoritmos de inducción como C4.5, PRISM, Teoría de la Información, RB (Redes Bayesianas); y los algoritmos de clustering LOF, DBSCAN y K-Means para BBDD alfanuméricas con tipos de procedimiento tanto supervisado como no supervisado.

A continuación, (tabla 2) se describe la utilización de métodos híbridos con distintos enfoques, dependiendo del tipo de aprendizaje de los algoritmos implicados; ya sean con aprendizaje supervisado y no supervisado en la detección de anomalías y ruido. Los métodos híbridos son la mejor alternativa para lograr la mayor ganancia de información, reducción del espacio de búsqueda y optimización de los procesos [8,9,10]. Investigaciones recientes han determinado que es posible afirmar que la combinación de algoritmos de distinta naturaleza, y también la combinación de procedimientos, permite optimizar el descubrimiento de anomalías [7]; siguiendo ese paradigma, se proponen los dos siguientes procedimientos alfanuméricos para el contralor cívico.

Tabla 2. Procedimientos híbridos propuestos según entorno, algoritmos y enfoque [].

Procedimientos híbridos	Entorno	Algoritmos y técnicas	Enfoques
I	BBDD alfanuméricas con un atributo objetivo	C4.5, Teoría de la información; LOF	No supervisado; supervisado
II	BBDD alfanuméricas que no contienen un atributo objetivo	LOF; DBSCAN; C4.5; RB; PRISM; K-Means	No supervisado; supervisado

En los procedimientos híbridos I y II, donde el procedimiento I, detecta campos de outliers en bases de datos alfanuméricas conteniendo un atributo clase [11,12,7], igualmente al procedimiento II, el cual también detecta campos en bases alfanumérica solo que sin un atributo target [13,7].

Teniendo conocimiento que las DDJJ se conforman habitualmente por datos alfanuméricos, se descubre la potencialidad de aplicar en su uso los procedimientos híbridos I y II (tabla 2) correspondientes a los procedimientos híbridos de detección de datos anómalos desarrollado recientemente por [7], los cuales se identifican como idóneos por las siguientes razones:

- [i] Son procedimientos desarrollados recientemente y representan el estado del arte en lo que hace a la detección de campos anómalos y ruido.
- [ii] Son procedimientos óptimos para la detección de ruido en BBDD alfanuméricas como generalmente se encuentran caracterizados los datos públicos.
- [iii] Son de fácil aplicabilidad y ejecución con los programas de minería de datos disponibles actualmente.

La idoneidad de estos procedimientos híbridos son propuestos como solución tentativa para un caso de contralor civil para la mejora de los datos públicos y cívicos de una población. A continuación se despliega una posible resolución y aplicación hipotética de contralor civil en base a los procedimientos alfanuméricos descriptos, sobre BBDD de DDJJ abiertas previamente tratadas.

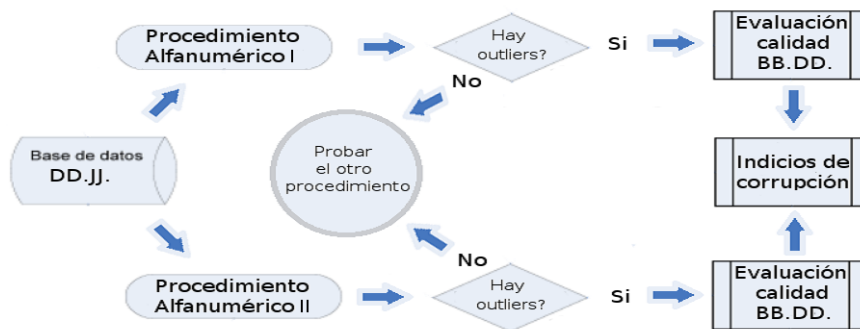


Fig. 4. Propuesta de aplicación de procedimientos híbridos de detección de campos anómalos sobre bases de datos de declaraciones juradas alfanuméricas.[Elaboración propia]

El procedimiento propuesto (figura 4) supone aplicar los procedimientos híbridos alfanuméricos de detección de anomalías en las BBDD preparadas de DDJJ para la detección de outliers, la posibilidad de detección de falsos positivos propone la retroalimentación circular tanto en uno como otro procedimiento alfanumérico propuesto, puesto que haya o no outliers, se procede igualmente con el procedimiento restante. De esta forma, se busca evaluar la calidad de los datos públicos implicados en el análisis a primera luz, y en un análisis más profundo *a posteriori* en base a los campos y atributos detectados, los indicios de comportamiento corrupto implícito, existente como output de la información pública procesada.

5. Experimentación y discusión de la metodología propuesta

De la experimentación se procede a validación del sistema propuesto en la Figura 4 en DDJJ públicas. Puesto que no se puede afirmar qué y cuales tuplas y atributos de la BBDD de DDJJ de inmuebles son realmente válidos o no, la existencia de sospecha de anomalías se asocia a determinados parámetros económicos característicos del inmueble no triviales a su composición, así como a la calidad de los datos relevados.

5.1 Validación de los procedimientos híbridos con BBDD de DDJJ

De la ejecución experimental del procedimiento I, en su fase con aprendizaje no supervisado al valor declarado como atributo clase del algoritmo de inducción C4.5, se obtienen atributos significativos al lograr la mayor ganancia informativa (tabla 3), sobre los cuales se elaboraron 6 bins de entrada-salida simulando un sistema de información; para *a posteriori*, ejecutar los flujos de minería con algoritmo LOF, donde “∞” corresponde a la cantidad de tuplas sospechosas de contener anomalías.

Tabla 3. Bins de Entrada-Salida con outliers detectados.

Bins de Entrada – (Salida)	Outliers detectados	Bins anómalos sospechosos Media o Moda(Moda)
superficiem2(val_decl)	968 (∞)	38733.15(No declara)
ano(val_decl)	122 (∞)	2001(Mercado)
nombre_bien_s(val_decl)	209 (∞)	Prop. Horizontal(Fiscal)
porcentaje(val_decl)	252 (∞)	29.18528(Sin datos)
val_patrim(val_decl)	1130 (∞)	146528.3(Subfiscal)
vinculo(val_decl)	30 (∞)	Conviviente(Subfiscal)

Donde los bins anómalos sospechosos, se asociaron a los siguientes valores medios y modas: [i] una superficie patrimonial promedio de 38733m² no declarado, [ii] inmuebles de DDJJ de 2001 declaradas a valor de mercado, [iii] propiedades horizontales a su valor fiscal, [iv] tenencia de porcentaje accionario del 29,19% promedio sin valoración monetaria, [v] valor inmobiliario promedio de \$146528 a valor subfiscal, e inmuebles del conviviente valuados subfiscalmente – [vi] –. Desde la teoría de la información [14], cuantitativamente, el valor patrimonial y la superficie parecen introducir mayor ruido y entropía al sistema de DDJJ.

Ejecutando el procedimiento II sin atributo clase, en la primera fase siguiendo reglas de determinación de outlier [7] se aplican LOF-DBSCAN y unión de algoritmos de clasificación C4.5-RB-PRISM donde se detectaron 2531 tuplas

sospechosas de contener anomalías, *i.e.*, un 38,19% de un total de 6627 tuplas, se conforma una BBDD de outliers para la siguiente fase la cual se transforma *a posteriori* para aplicar K-Means en dos agrupamientos, observándose los siguientes resultados.

Tabla 4. Distancia del centroide para cada atributo.

Atributo(id) (Cluster_0)	Valor distancia (Cluster_0)	Valor promedio (Cluster_1)
ddjj_id(1)	1.841	1.079
ano(2)	1.518	1.079
superficiem2(22)	2.033	1.079

Al ejecutar el algoritmo K-Means, el centroide más alejado contuvo los atributo superficie patrimonial, año e identificación de DJ; donde el primero – *superficiem2* – resulta el atributo más alejado así como el más sospechoso de contener campos anómalos.

5.2 Discusión de los resultados de la experimentación

El ruido anómalo producido por los atributos superficie, valuación, nombre del bien, porcentaje accionario, año y vínculo (tabla 3) vislumbra un sistema de información que expone las anomalías de los FFPP a la hora de valuar y declarar sus inmuebles frente a la ciudadanía; por otra parte, el atributo superficie presenta una fuente importante de dispersión que podría conllevar indicios de inmuebles extremadamente grandes junto a inmuebles irrisoriamente pequeños para ser considerados como tal.

Del primer análisis de inducción con atributo clase – procedimiento I –, mediante algoritmo C4.5, también se observaron las siguientes reglas:

- [i] Los inmuebles menores a 6500m² tienden a ser declarados a valor subfiscal.
- [ii] los inmuebles mayores a 6500m² – entre mediados de 2005 y 2012 – tienden a no declararse, ocurriendo lo mismo con cocheras, campos, terrenos, parcelas, lotes y propiedades horizontales sin especificación mayores a 6500m² y anteriores a 2005.
- [iii] cuando el funcionario posee una casa, y una participación accionaria superior al 46,3% [100%; 37,9%) o inferior al 37,9 (37,9% ; 0%] prefiere no declarar su valuación, pero tiende a declararla subfiscalmente cuando posee un rango accionario entre el 46% y el 38% (46,3%; 37,9%).
- [iv] Si el FP posee un departamento, entre mediados de 2003 y mediados de 2005, con superficie menor o igual a 26m², tiende a ser declarado a su valuación fiscal, mientras no se declara cuando es anterior a 2003. No obstante, los departamentos mayores a 26m² simplemente tienden a no declararse.

- [v] Los locales comerciales de los FFPP que poseen una valuación mayor a \$76493 $[\infty, \$76493)$ o inferiores a \$10615 $(\$10615, 0]$ tienden a ser declarados a su valor de mercado; a excepción de los locales con rango patrimonial entre $[\$76493, \$10615]$, puesto que en este caso también se tenderá a ser declarado a su valor de mercado, si y solo si, el inmueble se encuentra inscripto a nombre de su cónyuge, y no precisamente a nombre del oficial político, caso en el cual se tenderá a no declararlo.

6. Conclusiones y trabajos futuros

En este trabajo se constituyeron y combinaron procesos híbridos detectores de outliers y ruido configurados para trabajar con BBDD alfanuméricas en BBDD de DDJJ de bienes inmuebles, como caso inédito de uso de técnicas y procesos de minería de datos como herramienta contra la corrupción pública, poniéndose de relieve la potencialidad de los procesos de detección de anomalías al momento de evaluar la calidad de las BBDD públicas por un lado, y el descubrimiento de información sospechosamente portadora de comportamiento corrupto por el otro.

Los atributos superficie patrimonial, valor patrimonial, porcentaje accionario, nombre del bien y año son los que aportaron más entropía al sistema de DDJJ comprometiendo la calidad de la BBDD; dada su baja densidad relativa, la superficie del inmueble es el atributo más sospechoso de contener datos anómalos. Así mismo, las posibles anomalías detectadas; además de revelar inconsistencias en la BBDD, podrían develar indicios de comportamiento corrupto respecto la valuación fiscal inmobiliaria declarada por FFPP, puesto que las características del inmueble podrían condicionar la valuación patrimonial del mismo a su valor fiscal, subfiscal, o simplemente evitando declarar su valor. En este sentido, tanto las reglas de comportamiento como los bins entrada-salida podrían servir de estrategia de investigación cívica y contable en el combate a la corrupción pública, la evasión y la elusión impositiva en FFPP; lacerante, al sopesarse la ejemplaridad ideal del funcionario respecto la comunidad representada.

Futuros trabajos de investigación podrían contemplar la elaboración de variaciones algorítmicas en los procedimientos alfanuméricos propuestos; no descartándose la aplicación de variantes a los procesos alfanuméricos propuestos así como el de procesos de explotación de la información ajenos a la detección de datos anómalos y ruido sobre las mismas BBDD de DDJJ.

7. Referencias

- [1].Ransom J.: Replicating Data Mining Techniques for Development: A Case of Study of Corruption, Lund University, Master Thesis, Master of Science in International Development and Management, <http://lup.lub.lu.se/record/3798253/file/3910587.pdf>, (2013).
- [2].Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X.: The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, 50(3), 559-569 (2011).
- [3].Sowjanya, S., Jyotsna G.: Application of Data Mining Techniques for Financial Accounting Fraud Detection Scheme. *International Journal of Advanced Research in Computer Science and Software Engineering*, Noviembre, 3(11), 717-724 (2013).
- [4].Gómez N., Bello M. A.: Ética, transparencia y lucha contra la corrupción en la administración pública, Manual para el ejercicio de la función pública, 1ra ed.: Oficina Anticorrupción, Ministerio de Justicia y Derechos Humanos de la Nación, Mayo, CABA, <http://www.anticorrupcion.gov.ar/documentos/Libro%20SICEP%20da%20parte.pdf>, (2009).
- [5].DD.JJ. Abiertas: LNDData. Actualizado al 13/1/2014, CABA, <http://interactivos.lanacion.com.ar/declaraciones-juradas/>, (2015).
- [6].Hawkins, D. M.: Identification of outliers, Chapman and Hall., 11, London (1980).
- [7].Kuna H.: Procedimientos de explotación de la información para la identificación de datos faltantes con ruido e inconsistentes, Tesis doctoral, Universidad de Málaga, Marzo (2014).
- [8].Kuna, H., García Martínez, R., Villatoro, F.: Identificación de Causales de Abandono de Estudios Universitarios. Uso de Procesos de Explotación de Información. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, 5, 39-44 (2009).
- [9].Kuna, H., García-Martínez, R. Villatoro, F.: Pattern Discovery in University Students Desertion Based on Data Mining. In *Advances and Applications in Statistical Sciences Journal*, 2(2): 275-286 (2010).
- [10].Kuna, H., Pautsch, G., Rey, M., Cuba, C., Rambo, A., Caballero, S., Steinhilber, A., García-Martínez, R., Villatoro, F.: Avances en procedimientos de la explotación de información con algoritmos basados en la densidad para la identificación de outliers en bases de datos. *Proceedings XIII Workshop de Investigadores en Ciencias de la Computación*. Artículo 3745 (2011).
- [11].Kuna, H., Pautsch, G., Rey, M., Cuba, C., Rambo, A., Caballero, S., García-Martínez, R., Villatoro, F.: Comparación de la efectividad de procedimientos de la explotación de información para la identificación de outliers en bases de datos. *Proceedings del XIV Workshop de Investigadores en Ciencias de la Computación*, 296-300 (2012b).
- [12].Kuna, H., Pautsch, G., Rambo, A., Rey, M., Cortes, J., Rolón, S.: Procedimiento de explotación de información para la identificación de campos anómalos en base de datos alfanuméricas. *Revista Latinoamericana de Ingeniería de Software*, 1(3): 102-106 (2013b).
- [13].Kuna, H., García-Martínez, R., Villatoro, F.: Outlier detection in audit logs for application systems. *Information Systems* (2014).
- [14].Ferreira M.: *Powerhouse: Data Mining usando Teoría de la información*, (2007).

Approximate Nearest Neighbor Graph via Index Construction

Edgar Chávez, Verónica Ludueña, Nora Reyes, and Fernando Kasián

Departamento de Informática, Universidad Nacional de San Luis,
San Luis, Argentina

{vlud, nreyes, fkasian}@unsl.edu.ar

Centro de Investigación Científica y de Educación Superior de Ensenada, México
elchavez@cicese.mx

Abstract. Given a collection of objects in a metric space, the Nearest Neighbor Graph (NNG) associate each node with its closest neighbor under the given metric. It can be obtained trivially by computing the nearest neighbor of every object. To avoid computing every distance pair an index could be constructed. Unfortunately, due to the *curse of dimensionality* the indexed and the brute force methods are almost equally inefficient. This bring the attention to algorithms computing approximate versions of NNG.

The DiSAT is a proximity searching tree. It is hierarchical. The root computes the distances to all objects, and each child node of the root computes the distance to all its subtree recursively. Top levels will have accurate computation of the nearest neighbor, and as we descend the tree this information would be less accurate. If we perform a few rebuilds of the index, taking deep nodes in each iteration, keeping score of the closest known neighbor, it is possible to compute an Approximate NNG (ANNNG). Accordingly, in this work we propose to obtain de ANNNG by this approach, without performing any search, and we tested this proposal in both synthetic and real world databases with good results both in costs and response quality.

Keywords: similarity search, databases, metric spaces, approximate search

1 Introduction

Proximity searching consists in finding objects from a collection near a given query. The literature is vast and there are many specializations of the problem. Similarity search has become a very important operation in applications that deal with unstructured data sources. This has applications in a large number of fields. Some examples are non-traditional databases, text searching, information retrieval, machine learning and classification, image quantization and compression, computational biology, and function prediction. All those applications can be formalized with the *metric space model* [6]. A metric space is composed by a universe of objects \mathbb{U} , and a distance function d , the distance function gives us a dissimilarity criterion to compare objects from \mathbb{U} .

Similarity queries, in metrics spaces, are usually of two types, for a given database $S \subseteq \mathbb{U}$, a query $q \in \mathbb{U}$, and $r \in \mathbb{R}^+$: *range query*: retrieves all elements within distance r to q in S ; and *k-nearest neighbor*: retrieves the k closest elements to q in S . k - $NN(q)$ query is a building block for a large number of problems in a wide number of

application areas. For instance, in pattern classification, the nearest-neighbor rule can be implemented with 1-NN(q)’s [9].

The Nearest Neighbor Graph (NNG) is a graph with S the vertex set, with an edge from u to v whenever v is the nearest neighbor of u . It is often called the all-nearest neighbor problem. It could be generalized to retrieve the k -NN of *all* elements of database: the *All- k -NN* problem. It is a useful operation for batch-based processing of a large distributed point dataset, this will be our focus.

As the distance is considered expensive to compute, it is customary to use the number of distance evaluations as the complexity measure. For general metric spaces, there exist a number of methods to preprocess the database in order to reduce the number of distance evaluations [6], and then by performing n k -NN queries, avoiding the exhaustive search.

However, when the database is very large or the distance is too costly, building an index and then performing an exact k -NN query for each database element could be very expensive. In these cases, an alternative is to settle for the response to approximate similarity queries, which will save runtime at the price of losing accuracy in the response. But, it still could be very expensive, even more if we consider that in this way many calculated distances during the index construction are wasted, because queries do not take complete advantage of these calculations. Thus, it can be considered that an even cheaper way to calculate the approximate nearest neighbors could use directly the distances calculated during the index building, in order to approximate the response, especially if there is a reasonable chance that during the construction each element would be compared with very close elements. Such is the case of the (*DiSAT*).

Therefore, in this work we present a new method to solve the version approximate of the *All-1-NN* problem. This is the particular case of approximate *All- k -NN* problem, when $k = 1$ (*All-1-NN_A*), that uses the construction of a *DiSAT*, without performing any search. Besides, when the precision obtained with the response is not good enough, we propose an inexpensive way to continue improving it, even without to carry out any similarity search.

This paper is organized as follows: Section 2 presents a brief description of some useful concepts. In Section 3 we give a description of the *DiSAT*. Section 4 presents our proposal, and Section 5 contains the empirical evaluation of our proposed solution. Finally, in Section 6 we conclude and discuss about possible extensions for our work.

2 Previous Concepts

In this section we briefly state the problem in a more formal way to continue the discussion. A metric space is composed by a universe of objects \mathbb{U} , and a distance function $d : \mathbb{U} \times \mathbb{U} \rightarrow \mathbb{R}^+$, such that for any $x, y, z \in \mathbb{U}$, $d(x, y) > 0$ (strict positiveness), $d(x, y) = 0 \iff x = y$ (reflexity), $d(x, y) = d(y, x)$ (symmetry), and obeying the triangle inequality: $d(x, z) + d(z, y) \geq d(x, y)$. The smaller the distance between two objects, the more *similar* they are. We have a finite database S , which is a subset of \mathbb{U} and can be preprocessed. Later, given a new object from \mathbb{U} (a query q), we must retrieve all elements found in S close to q , using as few distance computations as possible. Similarity queries, in metrics spaces, are usually of two types, for a given database S with size $|S| = n$, $q \in \mathbb{U}$ and $r \in \mathbb{R}^+$: $(q, r) = \{x \in S \mid d(q, x) \leq r\}$ denote a

range query; and k -NN(q), denotes the k -nearest neighbors, formally it retrieves the set $R \subseteq S$ such that $|R| = k$ and $\forall u \in R, v \in S - R, d(q, u) \leq d(q, v)$. This primitive is a fundamental tool in cluster and outlier detection [4, 10], image segmentation [1], query or document recommendation systems [3], VLSI design, spin glass and other physical process simulations [5], pattern recognition [9], and so on.

The distance is considered expensive to compute (think, for instance, in comparing two fingerprints). Thus, the ultimate goal is to build *offline* an index in order to speed up *online* queries. Different techniques to solve the problem of similarity queries have arisen, in order to reduce these costs, usually based on data preprocessing. All those structures work on the basis of discarding elements using the triangle inequality, and most use the classical divide-and-conquer approach.

A version of the k -NN problem, perhaps less studied, is the *All- k -NN* problem. That is, if $|S| = n$, get the *All- k -NN* is retrieve, efficiently, the k -NN(u_i) for each u_i in S , performing less than $O(n^2)$ distance evaluations. It is a useful operation for batch-based processing of a large distributed point dataset. Consider, for example, a location-based service which recommends each user his or her nearby users, who may be the candidates of new friends. Given that locations of users are maintained by the underlying database, we can generate such recommendation lists by issuing an *All- k -NN* query on the database. In a centralized database environment, we can use the existing *All- k -NN* algorithms.

Some solutions to this problem have been proposed and developed for general metric spaces [14, 15], based on the construction of the *k -nearest neighbors graph* (k NNG). The k NNG is a weighted directed graph connecting each object from the metric space to its k nearest neighbors, that is, $G(S, E)$ such that $E = \{(u, v), u, v \in S \wedge v \in k\text{-NN}(u)\}$. G connects each element through a set of arcs whose weights are computed according to the distance of the corresponding space. Building the k NNG is a direct generalization of the *all-nearest-neighbor* (All-NN) problem, which corresponds to the 1NNG construction problem. The k NNG offers an indexing alternative which requires a moderately amount of memory, obtaining reasonably good performance in the search process. In fact, in low-memory scenarios, which only allow small values of k the search performance of k NNG is better than using classical pivot-based indexing alternative.

The naïve algorithm for *All- k -NN* calculates the distance function d between each $u_i \in S$ and every element of S , so it has quadratic complexity. Even, when we model similarity as a metric space, we are already approximating the real retrieval need of users. In fact, given a dataset, we can use several distance functions, each of them considering some aspects of objects and neglecting others. Likewise, when we design a model to represent real-life objects, we usually lose some information. Moreover, even if we find the proper metric and a lossless object representation, there are high-dimensional metric spaces where solving similarity queries requires reviewing almost all the dataset no matter what strategy we use. In addition, in many applications, the efficiency of the query execution is much more important than effectiveness. That is, users want a fast response to their queries and will even accept approximate results (as far as the number of drops and false hits is moderate). This has given rise to a new approach to the similarity search problem: we try to find the objects relevant to a given query with high probability. An intuitive notion of what this approach aims is that it attempts not to miss many relevant objects at query time.

The goal of the approximate search is to *significantly* reduce search times by allowing some “errors” in the query outcome. This alternative to the “exact” similarity searching is called *approximate similarity searching* [8], and it includes approximate and probabilistic algorithms. The general idea of approximate algorithms is to allow a relaxation on the precision of the query in order to obtain a speed-up the query time complexity. In addition to the query, a precision parameter ϵ is specified to control how far away we want the outcome of the query from the correct result. A reasonable behavior for this kind of algorithm is oncoming asymptotically to the correct answer as ϵ get closer to zero, and complementarily, speed up the algorithm, losing precision, as ϵ moves in the opposite direction. Therefore, the success of an approximation technique is based on the compromise quality / time [16].

To evaluate the performance of an approximate similarity search it must be considered: improvement in efficiency and accuracy of approximate results. The good approximation algorithms should offer large improvements in efficiency and high accuracy of approximate results. But, there must be a trade-off between both. The *improvement in efficiency* can be expressed as:

$$\frac{Cost(Q)}{Cost^A(Q)}$$

where $Cost(Q)$ and $Cost^A(Q)$ are the number of distance evaluations needed to perform an exact query and an approximate query Q , respectively. Q can be a range or a k -NN query.

When performing approximate searches must evaluate the retrieval effectiveness of a method. In an information-retrieval scenario, two measures are used as performance measures: *Recall* and *Precision*. Recall is defined as the number of *relevant objects* retrieved by a search divided by the total number of existing relevant objects. While precision is defined as the number of relevant objects retrieved by a search divided by the total number of objects retrieved by that search. If the R represents the result-set of an exact similarity search query and R^A the result-set returned by the approximation query, these measures can be formally established as:

$$Precision = \frac{|R \cap R^A|}{|R^A|} \quad \text{and} \quad Recall = \frac{|R \cap R^A|}{|R|}.$$

As we are focused on k -nearest neighbor searches, we can observe that given k the precise and approximate response sets both have a fixed cardinalities: k . Thus, the recall and precision measures always return identical values. Therefore, as follows we only use the precision measure.

Another measure to evaluate is the *relative error on distances* [2]. Relative error on distances compares the distances from a query object to the object in the exact and approximate results:

$$\frac{d(o_A, q) - d(o_R, q)}{d(o_R, q)} = \frac{d(o_A, q)}{d(o_R, q)} - 1$$

where o_A is the approximate nearest neighbor and o_R is the real nearest neighbor.

By this way, we computed the ratio between the distance to the object reported by the approximate algorithm and the real nearest neighbor minus 1. In our case, because

we want to compute the *All-1-NN*, the resulting quantity of the average over all the database elements is called the *average relative error on distances*.

3 The Distal Spatial Approximation Tree

The Spatial Approximation Tree (*SAT*) is a proposed data structure [12] based on a concept: approach the query spatially. It has been shown that the *SAT* gives better space-time tradeoffs than the other existing structures on metric spaces of high dimension or queries with low selectivity [12], which is the case in many applications. The Dynamic Spatial Approximation Tree (*DSAT*) [13] is an online version of the *SAT*. It is designed to allow dynamic insertions and deletions without increasing the construction cost with respect to the *SAT*. It is very surprising that *DSAT* is more efficient for searching than the *SAT*. For the *DSAT* the database is unknown beforehand and the objects arrive to the index at random as well as the queries. Then, it arises the *Distal Spatial Approximation Trees (DiSAT)* that improves regarding search performance over *SAT* and *DSAT*. *DiSAT* obtains better behavior on searches just by considering a different construction heuristic from *SAT*, but it maintains the same principles of searching and construction process.

The *SAT* is built as follows. An element a is selected as the root, and it is connected to a set of neighbors $N(a)$, defined as a subset of elements $x \in S$ such that x is closer to a than to any other element in $N(a)$. The other elements (not in $N(a) \cup \{a\}$) are assigned to their closest element in $N(a)$. Each element in $N(a)$ is recursively the root of a new subtree containing the elements assigned to it. From the previous definition of the *SAT*, the starting set for neighbors of the root a , $N(a)$ is empty. Particularly, *SAT* selects the first neighbor between all the elements in $S - \{a\}$, as its closest element and then considers if any other element can become a neighbor by analyzing them in an ordering from nearest to farthest. However, it could be possible to select any database element as the first neighbor. Inversely, *DiSAT* selects the first neighbor as its farthest elements in $S - \{a\}$ and uses the reverse ordering of the other elements to analyze if any of them can become a neighbor. Nevertheless, the same searching algorithm can be used on both trees because both uses the same condition to be a neighbor [12, 7]. This heuristic change of *DiSAT* increases the discarding power of the *SAT* by selecting distal nodes instead of the proximal nodes proposed in the original paper. Please note that this heuristic is the exact opposite of the original ordering in the construction of the *SAT*. Besides, *DiSAT* and *SAT* have the advantage of not having to tune any parameter.

Algorithm 1 gives a formal description of the construction of our data structure. As it can be seen in line 3, *DiSAT* uses farthest-to-nearest order from the root. Searching is done with the standard procedure. When working with hyperplanes to perform data separation it is advisable to use object pairs far from each other as documented in [6] for the *GNAT* and *GHT* data structures. Using the above observations, it is possible to ensure a good separation of the implicit hyperplanes by selecting the first neighbor as the farthest element to the root, and as a secondary effect the covering radii of neighbors are smaller than in *SAT*. Thereby, the partition induced by the *DiSAT* construction on the space has the nice property of obtaining a good data separation, that is useful for our approach to *All-1-NN*.

Algorithm 1 Algorithm to build a *DiSAT* for $S \cup \{a\}$ with root a .

```

BuildTree(Node  $a$ , Set of nodes  $S$ )
1.  $N(a) \leftarrow \emptyset$  /* neighbors of  $a$  */
2.  $R(a) \leftarrow 0$  /* covering radius */
3. For  $v \in S$  in decreasing distance to  $a$  Do
4.    $R(a) \leftarrow \max(R(a), d(v, a))$ 
5.   If  $\forall b \in N(a), d(v, a) < d(v, b)$  Then
6.      $N(a) \leftarrow N(a) \cup \{v\}$ 
7. For  $b \in N(a)$  Do  $S(b) \leftarrow \emptyset$ 
8. For  $v \in S - N(a)$  Do
9.    $c \leftarrow \operatorname{argmin}_{b \in N(a)} d(v, b), S(c) \leftarrow S(c) \cup \{v\}$ 
10. For  $b \in N(a)$  Do BuildTree( $b, S(b)$ )

```

4 Our proposal

As it is aforementioned, we decided to attack the problem of the approximate *All-1-NN*, i. e. to retrieve in an approximate way the near neighbor of *each* item in the database without comparing it against all the others. The idea of this proposal is maintaining for each item, during *construction* of the index, the closest element seen until this moment, obtaining its approximate near neighbor without any search in the index.

We use, as auxiliar structure, a *DiSAT* which do not require any parameter and produces a very good partition on the database. During tree construction we maintain for each object its closest element seen among all with which it was compared. When the construction finalizes we can retrieve for each $o_i \in S$ the approximate nearest neighbor x and its distance $d(o_i, x)$, where $1\text{-NN}_A(o_i) = \{x\}$.

Then, we consider the outcomes as balls centered in o_i with radii $r = d(o_i, x)$. Elements o_i whose balls have larger radii, have their $1\text{-NN}_A(o_i)$ farther away than the rest. This may be its real closest neighbor or not. If it is not its real closest neighbor the reason could be that it has not be compared against enough database objects, and so it is the best it got. We know that tree root compares with each database element, and its neighbors with most of them, but as we go down into the tree, the elements are compared with less objects each time. Then, by trying to improve mainly the neighbors of whose have large balls, we decide ordering all balls, in a decreasing order of its radii, and to rebuilt the *DiSAT* taking the elements in that order¹, the object in the first ball will be the tree root.

In this manner, we assure that by comparing the new root with all the database elements it will achieve its real neighbor. In this way the elements that follow in the order can become the neighbors of the root and thus it could obtain better nearest neighbors. Each time we compare an element o_i with another y that could be a better neighbor, we check if $d(o_i, y) < d(o_i, x)$, being x its current near neighbor, in wich case we update its approximate neighbor as y . If we are willing to pay a more distance calculations, we can repeat this process several times, trying to obtain as good approximate neighbors

¹ In fact, we can avoid the ordering of the elements with respect to the tree root for decreasing the CPU time.

as we can. Finally, we report the better approximate nearest neighbor achieved for each database element. As it can be noticed, we never perform any search on the index.

5 Experimental Results

The experiments consisted in obtaining the approximate *All-1-NN* of each element, only building the *DiSAT*. On the other hand, the exact 1-NN were calculated, building the *DiSAT* and performing the 1-NN search of each item in the database. We use *DiSAT* because in [7] it is shown that it is one of the most competitive index at searches. It is important to notice that there is the cheaper way to obtain the exact 1-NN of all the elements, avoiding the brute force approach.

As we mention previously, if we are willing to spend some distance calculations more, we can iterate the process in order to improve the answer. As we provide an approximate answer, we need to analyze its quality by calculating precision, recall, relative error of differences and complexity of each option. The total cost to obtain the $All-1-NN_A$ of our proposal is the number of distance evaluations done during all index constructions. The total cost of the *All-1-NN* considers all distance evaluations performed for construction and searching. All our results are averaged over 10 executions of the processes performed on different permutations of the datasets.

For the experiments, we consider a set of real-life metric spaces with widely different histograms of distances available from www.sisap.org [11]:

Strings: a dictionary of 69,069 English words. The distance is the *edit distance*, that is, the minimum number of character insertions, deletions and substitutions needed to make two strings equal.

NASA images: a set of 40,700 20-dimensional feature vectors, generated from images downloaded from NASA ². The Euclidean distance is used.

Color histograms: a set of 112,682 8-D color histograms (112-dimensional vectors) from an image database ³. Any quadratic form can be used as a distance, so we chose Euclidean distance.

Besides, in order to analyze how the intrinsic dimensionality affects the behavior of our approach, we experimentally evaluated the different solutions over synthetic metric spaces where we can control the intrinsic dimensionality. We used collections of 100,000 vectors of dimensions 4, 8, and 12, uniformly distributed in the unit hypercube. We did not use explicitly the information of the coordinates of each vector. In these spaces we also use Euclidean distance.

The Figure 1 illustrates the costs of the exact solution against our approximate proposal. We show the cost measured in distance evaluations for each rebuilding of the index. Thus, the first construction is indicated by 0 rebuilding, 1 rebuilding means that we have constructed firstly a *DiSAT* and secondly a *DiSAT* from the balls obtained with the first construction, and so on. As can be noticed, the cost of the exact solutions is shown as constant, because it does not depend of any rebuilding. Figure 1(a) shows the costs for the three real metric spaces. For example, in the plots we name *STRINGS-exact* the cost of *All-1-NN* and *STRINGS-approx* the cost of our $All-1-NN_A$ solution,

² At <http://www.dimacs.rutgers.edu/Challenges/Sixth/software.html>

³ At <http://www.dbs.informatik.uni-muenchen.de/~seidl/DATA/histo112.112682.gz>

for the space of Strings. Besides, as it can be seen, we use the same color for both costs on the same metric space. Alike, Figure 1(b) depicts the same experiments on the three synthetic spaces, designating the spaces of coordinate vectors in dimensions 4, 8, and 12 as C4, C8, and C12, respectively.

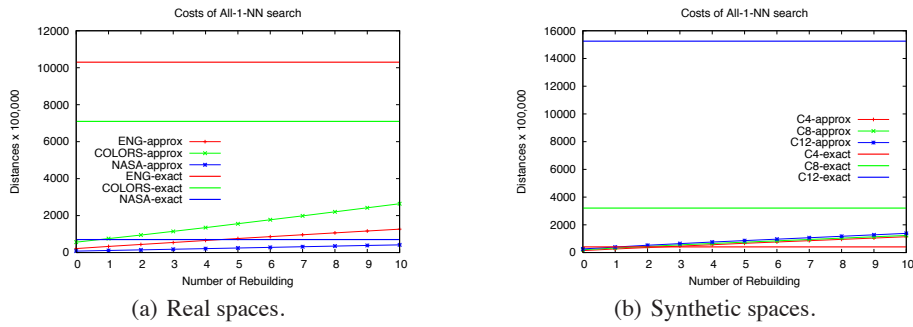


Fig. 1. Comparison of costs of exact and approximate *All-1-NN* for all metric spaces considered.

As it can be noticed, our proposal is significantly less expensive to perform in almost all the metric spaces used than the exact solution. Only, in the space of vectors in dimension 4, the exact alternative surpasses, although not significantly, our solution from the second rebuilding onwards. In order to show more clearly the improvement of costs, Figure 2 illustrates the improvements in efficiency obtained with our approximate solutions as we made more rebuildings of the index. Figure 2(a) shows that in all real the approximate method obtains a very significant efficiency. On the other hand, Figure 2(b) depicts the efficiency achieved over the three synthetic metric spaces. In this case we can observe that the improvement in efficiency is higher as dimension grows, and on dimensions 8 and 12 is always important, but on dimension 4 is inconsiderable.

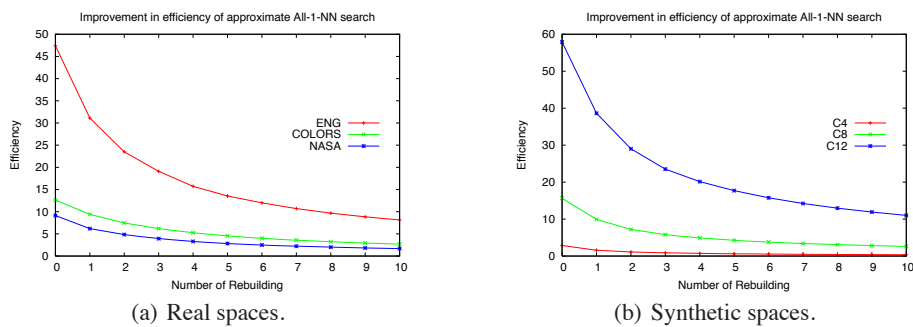


Fig. 2. Improvement in efficiency of *All-1-NN_A*, for all metric spaces considered.

In Figure 3 we show the precision of the response obtained with each reconstruction. After fourth reconstruction, it can observe that the answer exceeds 80% hits in the three real metric spaces (Figure 3(a)). However, in the synthetic spaces it needs more reconstructions to achieve a reasonable answer precision, but as it can be seen at Figure 1 we could spend many more reconstructions and even so to obtain much lower costs to response adequately to $All-1-NN_A$; that is to achieve a high quality solution at a low cost in distance evaluations.

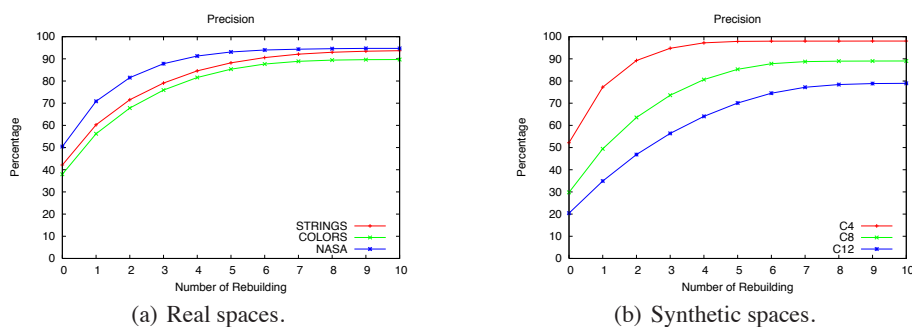


Fig. 3. Precision of the answer of $All-1-NN_A$, for all metric spaces considered.

We also evaluate the quality of the approximate solution by measuring the average relative error on distances. Figure 4 exhibits the error obtained versus the number of rebuilding, for the two kinds of metric spaces used. Over the real metric spaces, Figure 4(a) depicts that the error decreases fast, and as soon we rebuild the tree four times the error is almost zero, but for the Dictionary possibly because it uses a discrete distance. On the other hand, Figure 4(b) exposes the error for the three synthetic spaces used. As it can be seen, as dimension grows the average error decreases more slowly with the reconstructions. For instance, in the space of vectors in dimension 4 (C4) the percentage of error begins lower than 0.3 and achieves close to zero values with the fourth reconstruction.

6 Conclusions

In this paper we tested an alternate approach to computing an approximation to the $All-1-NN$ using a simple heuristic. We have designed an algorithm able to retrieve the $All-1-NN_A$ with a low cost, a very good accuracy, and low error. Our algorithm is based on the construction of the $DiSAT$, an index that was originally proposed only for the common similarity queries. Therefore, in addition to obtaining a good method for solve the $All-1-NN_A$, we have expanded the range of applications of the $DiSAT$.

The novelty of our proposal is that no searches are performed, but only the distances calculated during the construction of the $DiSAT$ are used. Our results are preliminary and encouraging. We obtained good performance with low and medium dimensionality databases, we are aiming at improving the results to tackle higher dimensionalities.

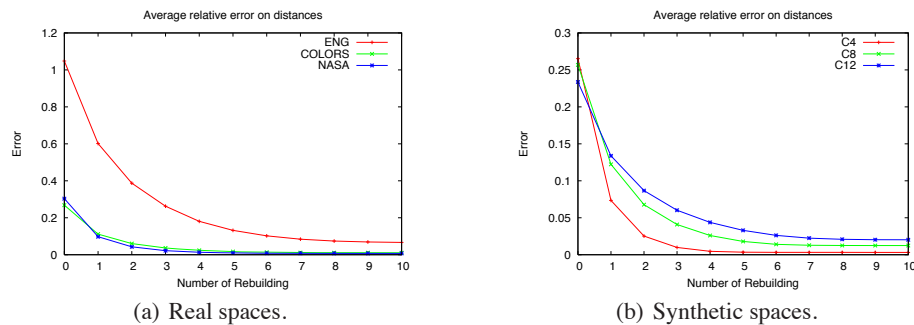


Fig. 4. Average relative error on distances of the answer of $All-1-NN_A$, for all metric spaces considered.

References

1. N. Archip, R. Rohling, P. Cooperberg, H. Tahmasebpour, and S. K. Warfield. Spectral clustering algorithms for ultrasound image segmentation. volume 3750, pages 862–869, 2005.
2. Sunil Arya, David M. Mount, Nathan S. Netanyahu, Ruth Silverman, and Angela Y. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *J. ACM*, 45(6):891–923, November 1998.
3. R. Baeza-Yates, C. Hurtado, and M. Mendoza. Query clustering for boosting web page ranking. pages 164–175, 2004.
4. M. Brito, E. Chávez, A. Quiroz, and J. Yukich. Connectivity of the mutual k -nearest neighbor graph in clustering and outlier detection. *Statistics & Probability Letters*, 35(4):33–42, 1996.
5. P. Callahan and R. Kosaraju. A decomposition of multidimensional point sets with applications to k nearest neighbors and n body potential fields. *JACM*, 42(1):67–90, 1995.
6. E. Chávez, G. Navarro, R. Baeza-Yates, and J. Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, September 2001.
7. Edgar Chávez, Verónica Ludeña, Nora Reyes, and Patricia Roggero. Faster proximity searching with the distal sat. *Information Systems*, 59:15–47, 2016.
8. P. Ciaccia and M. Patella. Approximate and probabilistic methods. *SIGSPATIAL Special*, 2(2):16–19, 2010.
9. R. Duda and P. Hart. Pattern classification and scene analysis. *John Wiley & Sons*, 1973.
10. D. Eppstein and J. Erickson. Iterated nearest neighbors and finding minimal poly-topes. volume 11, pages 321–350, 1994.
11. Karina Figueroa, Gonzalo Navarro, and Edgar Chávez. Metric spaces library, 2007. Available at http://www.sisap.org/Metric_Space_Library.html.
12. G. Navarro. Searching in metric spaces by spatial approximation. *The Very Large Databases Journal (VLDBJ)*, 11(1):28–46, 2002.
13. G. Navarro and N. Reyes. Dynamic spatial approximation trees. *Journal of Experimental Algorithmics*, 12:1–68, 2008.
14. R. Paredes. *Graphs for Metric Space Searching*. PhD thesis, University of Chile, Chile, July 2008.
15. R. Paredes, E. Chávez, K. Figueroa, and G. Navarro. Practical construction of k -nearest neighbor graphs in metric spaces. In *Proc. 5th Workshop on Efficient and Experimental Algorithms (WEA)*, LNCS 4007, pages 85–97, 2006.
16. M. Patella and P. Ciaccia. Approximate similarity search: A multifaceted problem. *J. Discrete Algorithms*, 7(1):36–48, 2009.

An exploratory analysis of methods for extracting credit risk rules

Patricia Jimbo Santana¹, Augusto Villa Monte², Enzo Rucci^{2,3}
Laura Lanzarini², Aurelio F. Bariviera⁴

¹ Carrera de Contabilidad y Auditoría, Facultad de Ciencias Administrativas,
UCE Universidad Central del Ecuador, Quito – Ecuador
prjimbo@uce.edu.ec

²III-LIDI, Fac. de Informática, Universidad Nacional de la Plata,
50 y 120, La Plata, Buenos Aires, Argentina,
{avillamonte, erucci, laural}@lidi.info.unlp.edu.ar

³ III-LIDI, CONICET, Fac. de Informática, UNLP, La Plata, Argentina

⁴Departament of Business, Universitat Rovira i Virgili,
Avenida de la Universitat, 1 Reus, Spain
aurelio.fernandez@urv.cat

Abstract. This paper performs a comparative analysis of two kind of methods for extracting credit risk rules. On one hand we have a set of methods based on the combination of an optimization technique initialized with a neural network. On the other hand there are partition algorithms, based on trees. We show results obtain on two real databases. The main findings are that the set of rules obtained by the first set of methods give a set of rules with a reduced cardinality, with an acceptable precision regarding classification. This is a desirable property for financial institutions, who want to decide credit approval face to face with customers. Bank employees who daily deal with retail customers can be easily trained for selecting the best customers, by using this kind of solutions.

Keywords: credit scoring, classification rules, Learning Vector Quantization (LVQ), Particle Swarm Optimization (PSO).

1 Introduction

The 21st century presents an increase in the development and consumption of goods. The extension of financial services in emerging economies is particularly important. Financial intermediation provides a solution to immediate consumption of durable goods, helping to defer the payment several months or years. This sort of “democratization” in consumption poses a challenge to financial institutions. Whereas mortgage lending applications, due to its comparatively reduced number of borrowers, can be decided at a slower pace, consumer lending needs faster (even instantaneous online) decision procedures. Borrowers want small credits for buying home equipment, a car, a trip, etc. They are eager of a quick answer. From the point of view of the borrowers, they want to receive a quick positive answer to their

applications. On the other side, financial institutions want to find the appropriate rules in order to approve credit application only to good borrowers, *i.e.* those who pay back their financial commitments.

Financial institutions typically ask exhaustive information about the potential client: age, marital status, salary, other debts, job type, etc. This information is gathered in order to be analyzed, using some decision model. The result of this analysis is either to grant or reject the credit.

The increasing number of applicants and data raises the necessity for suitable techniques that deals with the complexity of this multidimensional problem. Timely resolution of credit applications is key element when deciding a credit scoring method. Precisely, the area known as data mining can shed light on this kind of situations.

Data mining comprises a set of techniques that are able to model available information. One of the most important stages in the process is knowledge discovery. It is characterized by obtaining new and useful information without assuming prior hypothesis. One of the preferred techniques by decision makers is the association rule. An example of association rule is an expression: IF condition1 THEN condition2, where both conditions are conjunctions of propositions of the form (attribute = value) and whose solely restriction is that attributes in the antecedent must not be present in the consequent. When a set of association rules presents in the consequent, the same attribute is called a set of classification rules (Witten [15], Hernández & Ramírez [7]). The aim of this paper is to present several alternatives of scoring methods. We believe that the combination of a method for obtaining classification rules and competitive neural network provides an intuitive solution with acceptable levels of errors. Standard methods based on classification trees provide good benchmark for alternative ones. The main advantage of the proposed alternatives is the production of a reduced set of rules, that improves both the transparency and delay in the decision making process of the financial institutions.

The rest of the paper is structured as follows. Section 2 briefly discusses relevant literature on credit risk. Section 3 describes the neural network, metaheuristics, and the proposed method. Section 4 describes data and presents results of a true empirical application and section 5 draws the main implications of our proposal.

2 Brief literature review

One of the oldest papers on bankruptcy prediction is FitzPatrick [5] who, using 13 accounting ratios calculated for 40 firms during three years. In the 1960s, the development of the capital markets in the United States, showed the necessity for more scientific models to assess economic corporate strength. Consequently, the first z-score model by Altman [2] was developed. At that time, the main concern of banks was to classify corporations according to their credit risk, since they were the main clients. However, in the last decades, there has been an increase in consumer credit. Retail banking became a growing industry. Not only there has been a boom in credit card memberships, especially in emerging economies, but also an increase in small consumption credits.

There are several methods to construct rules in order to evaluate the creditworthiness of credit applicants. Computational intelligent techniques produce, exploiting exhaustive credit databases can obtain better results, by capturing subtle characteristics of customers. These techniques, without being exhaustive, include artificial neural networks, fuzzy set theory, decision trees, support vector machines, genetic algorithms, among others. Artificial neural networks is a family of neural networks with different architectures. These architectures include popular models such as back propagation networks, self-organizing maps and learning vector quantization. Decision trees transform data in a tree-shape structure of leaf and decision nodes, and the goal is to test attributes to each branch of the tree, that constitutes a class. Support vector machines search an optimal hyperplane in order to generate a binary classification, maximizing the margin of separation between classes. Genetic algorithms are a set of methods to optimized problems, based on the evolutionary idea of natural selection.

If the goal is to obtain association rules, the a priori method (Agrawal and Srikant [1]) or some of its variants could be used. This method identifies the most common sets of attributes and then combines them to get the rules. There are variants of the a priori method, are usually oriented reduce computation time.

Under the topic classification rules, the literature contains various construction methods based on trees such as C4.5 (Quinlan [14]) or clipped trees as the PART method (Frank and Witten [7]). In both cases, the key is to get a set of rules that covers the examples fulfilling a preset error bound. The methods of construction rules from trees are partitives and are based on different attributes' metrics to assess its ability to cover the error bound.

The original PSO method defined in Kennedy and Eberhart [8] was extended in Lanzarini et al. [10] and [12] in order to obtain classification rules. This extension was later applied to two public databases of credits, in order to obtain a set of rules with low cardinality.

Brown and Mues [4] compared several techniques that can be used for imbalanced credit scoring data sets. Imbalance is a typical feature of credit data sets: in healthy financial institutions the number of defaulting loans is much lower than good performing loans. As a consequence the two classes could be not evenly represented search space. Blanco et al. [3] implements credit scoring modles based on multilayers perceptron approach, and benchmark the performance with linear and quadratic discriminant analysis, using a small sample of a microfinance institution in Peru. They find that neural networks based models outperforms classical discriminant methods.

4 Data and Results

We test alternative methods in two real databases and two consumer credit financial data from UCI Machine Learning Repository [13]. One of the real databases comes from an important savings and credit institution (Banco Solidario) of Ecuador with more than 20 yeas of trajectory in the domestic market. This data comprises credit operations between 2011 until August 2014, with the following attributes: status; date

of application; branch; province; requested amount; authorized amount; purpose of the credit; cash, bank accounts, investments, other assets, liabilities and salary of the applicant; date of verification of information; date of authorization; approval/denial date; cash, bank accounts, investments, other assets, liabilities and salary of the applicants' partner. In case, the applicant is a small business data requested are revenues and expenses of the business. The 'status' variable correspond to the situation of the credit. Applications can be denied or accepted. In case of being accepted, the status is classified between credits that were duly repaid and those with some delay in the payback. In turn, overdue loans are classified, according to the credit procedures manual between those with less than 90 days overdue, and those with more than 90 days overdue (initiation of legal actions). The other real database is from a mutual savings institution of Ecuador, with the same variables described above, with operations between 2011 and 2015.

Using the data described above, we compare the performance of several competing methods that combined a fixed and variable population PSO, initialized with two competitive neural networks (LVQ and SOM [9]). We compare these solutions with C4.5 methods defined by Quinlan [14] and PART defined by Frank and Witten [6]. The way of finding classification rules in proposed and control methods is different. On one hand rules discovering is done after searching task. On the other hand, control methods obtain rules, based in a partition strategy. C4.5 is a pruned tree whose branches are mutually exclusive and allow classifying examples. PART gives as a result a list of rules equivalent to those generated by the proposed classification method, but in a deterministic way. PART operation is based on the construction of partial trees. Each tree is created in a similar manner to that proposed for C4.5 but during the process construction errors of each branch are calculated. These errors allow the selection of the most suitable combinations of attributes.

We performed 30 independent runs of each method. For fixed population PSO, we use a competitive network of 30 neurons, whereas for the variable population case, the size begins with 9 neurons. PART method was executed with a confidence factor of 0.3 for the pruned tree. For other parameters default values were used.

Tables 1, 2, 3, and 4 summarize the results obtained by applying each method in each database. In each case was considered not only the accuracy of coverage of the rule set, but also the "transparency" of the obtained model. This "transparency" is reflected in the average number of rules obtained and the average number of terms used to form the antecedent.

The most important feature of our results, is that the combination of a search algorithm with a competitive neural network, gives a set of rules with a significant low cardinality, *vis-à-vis* the partition algorithms. Although partition algorithms provide more accuracy, this is at expense of a much larger number of rules. In fact, the difference in accuracy between both types of methods is within the range of 1 to 3 percentage points. We have to highlight that the accuracy of the classification based on PSO is very good and comparable to the other methods. However, regarding the number of rules is between 10 and 20 times larger in partition methods. Consequently, there is some sort of trade-off between simplicity and accuracy. Given that credit rules should be simple, in order to give customers a quick answer (for example, in consumer online credits), we believe that competitive search based methods are a good alternative to partition methods.

Table 1. Results on Australian database

Method		TRUE +	TRUE -	False +	False -	Precision	#rules	length antecedent
SOM + PSO	Mean	0.4257	0.4333	0.1097	0.0309	0.8590	3.0167	1.3711
	sd	0.0154	0.0103	0.0069	0.0066	0.0099	0.0461	0.1922
SOM + varPSO	Mean	0.4183	0.4391	0.1071	0.0351	0.8574	3.0000	1.5178
	sd.	0.0132	0.0158	0.0130	0.0077	0.0104	0.0000	0.1085
LVQ + PSO	Mean	0.4201	0.4414	0.1079	0.0306	0.8614	3.0000	1.2667
	sd	0.0179	0.0172	0.0093	0.0065	0.0105	0.0000	0.1207
LVQ + varPSO	Mean	0.4199	0.4382	0.1054	0.0363	0.8582	3.0000	1.5578
	sd	0.0179	0.0172	0.0075	0.0073	0.0092	0.0000	0.1336
C4.5	Mean	0.3910	0.4618	0.0847	0.0625	0.8528	18.2200	4.8394
	sd	0.0121	0.0063	0.0066	0.0120	0.0124	2.0825	0.2810
PART	Mean	0.3564	0.3906	0.1562	0.0969	0.7469	33.3433	2.4926
	sd	0.0136	0.0288	0.0289	0.0134	0.0292	1.5793	0.0934

Table 2. Results on German database

Method		TRUE BAD	TRUE GOOD	False BAD	False GOOD	Precision	#rules	length antecedent
SOM + PSO	Mean	0.1026	0.5993	0.0984	0.1994	0.7019	8.4400	2.1619
	sd	0.0123	0.0183	0.0149	0.0183	0.0153	0.6009	0.1415
SOM + varPSO	Mean	0.1046	0.5954	0.1034	0.1965	0.7000	8.0233	2.0464
	sd	0.0115	0.0135	0.0110	0.0134	0.0162	0.6745	0.1030
LVQ + PSO	Mean	0.0999	0.5997	0.1017	0.1986	0.6996	8.7767	2.1802
	sd	0.0151	0.0171	0.0136	0.0153	0.0133	0.7224	0.1075
LVQ + varPSO	Mean	0.1089	0.5973	0.1057	0.1880	0.7063	8.8867	2.0884
	sd	0.0100	0.0129	0.0103	0.0116	0.0109	0.4918	0.0960
C4.5	Mean	0.1219	0.5894	0.1106	0.1781	0.7113	86.4600	5.6267
	sd	0.0069	0.0070	0.0070	0.0069	0.0079	4.0788	0.1382
PART	Mean	0.1404	0.4385	0.1687	0.2258	0.6967	70.9133	3.0138
	sd	0.0120	0.0091	0.0135	0.0170	0.0139	2.1575	0.0561

Table 3. Results on Cooperativa de Crédito database

Method		TRUE N	TRUE O	False N	False O	Precision	#rules	length antecedent
SOM + PSO	Mean	0.6242	0.1601	0.1253	0.0898	0.7844	3.7867	1.6375
	sd	0.0069	0.0062	0.0057	0.0059	0.0059	0.2980	0.2151
SOM + varPSO	Mean	0.6014	0.1914	0.0947	0.1125	0.7928	4.1533	1.6953
	sd	0.0052	0.0059	0.0057	0.0047	0.0030	0.2801	0.0867
LVQ + PSO	Mean	0.6227	0.1671	0.1191	0.0910	0.7899	3.2933	1.4021
	sd	0.0048	0.0055	0.0051	0.0039	0.0031	0.1837	0.1066
LVQ + varPSO	Mean	0.6029	0.1902	0.0956	0.1114	0.7930	4.3733	1.6553
	sd	0.0056	0.0055	0.0054	0.0053	0.0025	0.2625	0.0567
C4.5	Mean	0.6320	0.1786	0.1075	0.0819	0.8106	114.2600	9.6762
	sd	0.0014	0.0013	0.0013	0.0013	0.0011	6.0543	0.1144
PART	Mean	0.6229	0.1825	0.1036	0.0910	0.8054	42.3567	4.6956
	sd	0.0065	0.0064	0.0064	0.0065	0.0023	2.1661	0.0880

Table 4. Results on Solidario database

Method		TRUE N	TRUE O	False N	False O	Precision	#rules	length antecedent
SOM + PSO	Mean	0.0457	0.8863	0.0228	0.0451	0.9320	4.3967	5.4962
	sd	0.0058	0.0047	0.0045	0.0056	0.0050	0.4895	0.4221
SOM + varPSO	Mean	0.0609	0.8870	0.0210	0.0300	0.9480	3.8600	2.8940
	sd	0.0029	0.0059	0.0051	0.0030	0.0063	0.2415	0.3532
LVQ + PSO	Mean	0.0482	0.8870	0.0219	0.0428	0.9352	4.6067	5.9166
	sd	0.0052	0.0056	0.0055	0.0049	0.0054	0.4193	0.2771
LVQ + varPSO	Mean	0.0565	0.8882	0.0198	0.0346	0.9447	3.8533	3.1013
	sd	0.0043	0.0062	0.0056	0.0043	0.0056	0.2921	0.3395
C4.5	Mean	0.0762	0.9017	0.0073	0.0148	0.9779	153.5733	11.2349
	sd	0.0003	0.0003	0.0003	0.0003	0.0003	5.1687	0.1565
PART	Mean	0.0749	0.9013	0.0077	0.0161	0.9762	80.9400	4.7650
	sd	0.0010	0.0014	0.0014	0.0010	0.0008	2.2034	0.0688

5 Conclusions

We compute several variations of a competing method for credit scoring using a variation of PSO (fixed and variable population), and a neural network. We test our model on two actual credit databases from important retail credit institutions from Ecuador and from two public databases from a repository. Results show that search algorithms allow to reduce significantly the number of rules, required to reach an acceptable and very similar level of classification accuracy.

Future research lines is to explore the incidence of the initial settings such as the speed of growth in population, which helps to determine the antecedent of rules. Results show no significant difference between fixed and variable population PSO. This means that the exploration of the solution space is not satisfactory solved.

Finally, we would like to highlight that the goal of our work is to achieve an intuitive model for credit scoring with a comparable accuracy to popular benchmark models. Our results suggest that the simplification of decision rules generates transparency in credit scoring, which could improve the reputation of financial institutions.

References

1. Agrawal, R., Srikant, R., Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases, VLDB '94, pp. 487–499. Morgan Kaufmann Publishers Inc., San Francisco (1994).
2. Altman, E.I., 1968. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance*, 23(4), pp.589–609.
3. Blanco, A., Pino-Mejías, R. Lara, J. Rayo, S.: Credit scoring models for the micro-finance industry using neural networks: evidence from Peru, *Exp. Syst. Appl.* 40 356–364 (2013).
4. Brown, I., Mues, C.: An experimental comparison of classification algorithms for imbalanced credit scoring data sets, *Exp. Syst. Appl.* 39, 3446–3453 (2012).
5. FitzPatrick, P.J., (1932). A comparison of the ratios of successful industrial enterprises with those of failed companies. *The Certified Public Accountant*, Oct., Nov., Dec.
6. Frank, E., Witten, I. H., 1998. Generating accurate rule sets without global optimization. In: Proceedings of the Fifteenth International Conference on Machine Learning, ICML '98., pp. 144–151. Morgan Kaufmann Publishers Inc., San Francisco.
7. Hernández Orallo, J., Ramírez Quintana, M.J., Ferri Ramírez, C., 2004. *Introducción a la Minería de Datos*. 1ra Edición. Pearson.
8. Kennedy, J. & Eberhart, R., 1995. Particle swarm optimization. In, Proceedings of IEEE International Conference on Neural Networks. pp. 1942–1948 vol.4.
9. Kohonen, T. *Self-Organizing Maps*. Volume 30, Springer Series in Information Sciences. Springer, Heidelberg (2012).
10. Lanzarini, L., Villa Monte, A., Aquino, G., De Giusti, A.: Obtaining classification rules using lvqPSO *Advances in Swarm and Computational Intelligence. Lecture Notes in Computer Science*. Vol 6433, 183-193. Heidelberg: Springer-Verlag Berlin (2015).
11. Lanzarini, L., Villa Monte, A.; Bariviera, A.F., Jimbo Santana, P.: Obtaining Classification Rules Using LVQ+PSO: An Application to Credit Risk. In J. Gil-Aluja et al., eds. *Scientific Methods for the Treatment of Uncertainty in Social Sciences. Advances in Intelligent Systems and Computing*. Springer International Publishing, 383–391 (2015).

12. Lanzarini, L., Villa-Monte, A., Ronchetti, F.: SOM+PSO. A Novel Method to Obtain Classification Rules. *Journal of Computer Science & Technology (JCS&T)*, 15(1), 15-22 (2015).
13. Lichman, M.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA. <http://archive.ics.uci.edu/ml>. Accessed 5 Jan 2015 (2013)
14. Quinlan, J.R.: C4.5: programs for machine learning, Morgan Kaufmann Publishers (1993).
15. Witten, I.H., Eibe, F. & Hall, M.A.: *Data Mining Practical Machine Learning Tools and Techniques* 3rd. ed., San Francisco, CA: Morgan Kaufmann Publishers Inc. (2011).



XI Workshop Arquitectura, Redes y Sistemas Operativos (WARSO)

Comité Científico:

Acosta, Nelson(Argentina)
Ardenghi, Jorge(Argentina)
Bevilacqua Roberto(Argentina)
Buckle, Carlos(Argentina)
Castro Lechtaler, Antonio (Argentina)
Díaz F. Javier(Argentina)
Echaiz, Javier(Argentina)
Fischer, Stefan(Alemania)
García Garino, Carlos(Argentina)
García Villalba, Luis Javier(España)
Marrone, Luis(Argentina)
Matrángolo, Carlos(Argentina)
Mercado, Gustavo(Argentina)
Molinari, Lía(Argentina)
Padovani, Hugo(Argentina)
Steinmetz, Ralf(Alemania)
Tarouco, Liane(Brasil)
Ultreras, Florencio(Chile)
Vendrell Vidal, Eduardo(España)



XI Workshop Arquitectura, Redes y Sistemas Operativos (WARSO)

ID	Trabajo	Autores	
9123	Control difuso de una cámara de germinación	Marcelo Marinelli (UNaM), Nelson Acosta (UNCPBA), Juan Toloza (UNTREF), Carlos Kornuta (UNCPBA)	844
9153	Análisis de impacto en la implementación de DNSSEC en un servidor DNS Recursivo	Ernesto Sanchez (UNSa), Daniel Arias Figueroa (UNSa), Sergio Rocabado (UNSa)	854
9163	Transporte de datos Multicast Confiable con Control de Congestión Dinámico	Gabriel Gomiz (UNLP), Luis Marrone (UNLP)	863
9168	Ecuación de estado generalizada para redes de Petri no autónomas y con distintos tipos de arcos	Orlando Micolini (UNC), Marcelo Cebollada y Verdaguer (UNC), Maximiliano Andres Eschoyez (UNC), Luis Orlando Ventre (UNC), Marcelo Ismael Schild (UNC)	873
9188	Diseño de un componente de simulación para comunicaciones en redes de sensores	Martin Pi Puig (UNLP), Santiago Medina (UNLP), Ary Batista (UNQ), Diego Encinas (UNLP), Fernando Romero (UNLP), Armando De Giusti (UNLP), Fernando G. Tinetti (UNLP)	883
9234	Emulación de elementos de networking interactuando con máquinas virtuales	Carlos Binker (UNLaM), Alejandro Perez (UNLaM), Guillermo Buranits (UNLaM), Eliseo Zurdo (UNLaM)	893
9316	Utilización de los Espacios Blancos: Una solución al problema de la saturación del espectro de frecuencias	Antonio Castro Letchtaler (UBA), Antonio Foti (UNTREF), Alejandro Arroyo Arzubi (EST-IESE), Jorge García Guibout (UNCUYO), Fernanda Beatriz Carmona (UNdeC), Rubén Jorge Fusario (UBA), Alejandro Oliveros (UNTREF)	903

Control difuso de una cámara de germinación

Marcelo Marinelli¹, Nelson Acosta², Juan Toloza³, Carlos Kornuta^{2,4}

¹ Instituto de Biotecnología de Misiones - Universidad Nacional de Misiones

² Facultad Cs. Exactas - Universidad Nacional Centro de la Prov. Buenos Aires

³ Universidad Nacional de Tres de Febrero

⁴ Becario doctoral CONICET

marcelomarinelli@gmail.com, ncornuta@exa.unicen.edu.ar, jmtoloza@untref.edu.ar

Resumen. Se presenta el diseño de un sistema de control difuso para una cámara de germinación de cultivos hidropónicos. El sistema toma mediciones de humedad y temperatura dentro de la cámara y por intermedio de un controlador difuso tipo Sugeno, embebido en una placa Arduino, genera señales de control a un sistema de ventilación forzada y al sistema de control de iluminación artificial. Se realizaron experiencias con una cámara de germinación con este control difuso. Se sembraron semillas de lechuga en la cámara con control difuso y en una bandeja de cultivo convencional obteniéndose resultados de crecimiento superiores en la primera.

Palabras clave: lógica difusa, sistemas embebidos, Sugeno, Cámara de germinación.

1 Introducción

La Teoría de los conjuntos borrosos iniciada por Zadeh¹ [1] a diferencia de la lógica matemática permite utilizar términos lingüísticos de fácil interpretación. Una rama de la lógica difusa se ocupa del estudio de sistemas de control [2]. Los controladores difusos tienen la ventaja de poseer mayor interpretabilidad para poder ser diseñados en función de un experto en una temática determinada. A tal efecto, se pueden utilizar etiquetas lingüísticas para las variables intervinientes como: “muy bajo”, “bajo”, “alto”, “muy alto”. Por otra parte, los conjuntos difusos que definen las variables, tienen valores de pertenencia flexibles. Estas características hacen que los sistemas de controles difusos puedan tener una mejor respuesta que los sistemas reactivos convencionales. Por otra parte, facilita generar un sistema de control basado en la experiencia de un experto, en lugar de basarse en un modelo matemático

En las cámaras de germinación hidropónicas una variable importante, en la germinación de las semillas y el desarrollo de los cultivos, es la ventilación, no sólo por el aporte de CO₂ sino también por el control de temperatura [3], [4]. Con el sistema propuesto el controlador difuso tipo Sugeno toma valores de temperatura y humedad dentro de la cámara de germinación y genera las señales de control para el

sistema de ventilación forzada y el aporte de temperatura provisto por una lámpara incandescente.

2 Sistema de adquisición de datos

Las variables de humedad y temperatura se adquirieron con sensores DHT22 que permiten tomar muestras de estas por una conexión serial.

Para el procedimiento de adquisición de los datos, se desarrolló un software de controlador difuso tipo Sugeno embebido y control de las señales de salida se utilizó una placa Arduino Mega 2560 con un procesador Atmel modelo Atmega2560. Cuenta con 54 pines digitales de entrada / salida (de los cuales 15 se pueden utilizar como salidas PWM), 16 entradas analógicas, 4 puertos seriales, una conexión USB, un conector de alimentación, y un botón de reinicio.

El Atmega2560 tiene 256 KB de memoria flash, para el almacenamiento de código. Para el gestor de arranque (bootloader) se utilizan 8 Kb, también consta 8 KB de SRAM y 4 KB de EEPROM, esta se puede leer y escribir con la librería EEPROM.

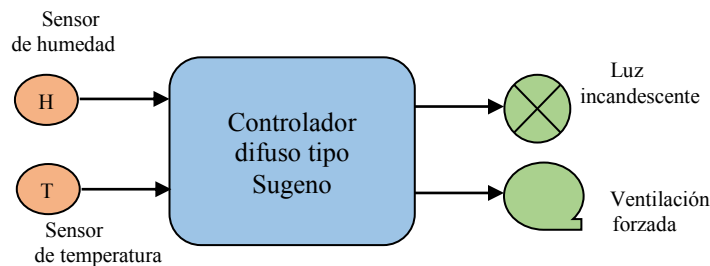


Fig. 1. Diagrama en bloques del sistema de control difuso

Para evitar contacto galvánico entre el sistema digital de la placa Arduino y las etapas de potencia se utilizaron optoacopladores que controlan los sistemas de iluminación artificial y ventilación forzada. En la Figura 1 se puede apreciar un diagrama en bloques del controlador difuso.

2.1 Diseño del controlador difuso

Dado que el controlador difuso reside embebido en la memoria de la placa Arduino, se optó por un sistema de control tipo Sugeno ya que el cálculo de la inferencia difusa se puede diseñar en menos líneas de código que los controladores tipo Mamdani debido a que no poseen etapa de defusificación.

Para el diseño de las particiones difusas se trabajó con datos provistos por expertos en Biología Vegetal del Instituto de Biotecnología de Misiones "María Ebe Reca" (InBioMis), dependiente de la Universidad Nacional de Misiones.

Se definieron los conjuntos difusos de entrada (Figuras 2 y 3) y salida así como el universo de discurso de cada variable (U_{T° , U_H , U_V , U_{LC}) y las etiquetas lingüísticas (Tablas 1 y 2).

2.2 Variable de entrada

2.2.1 Temperatura

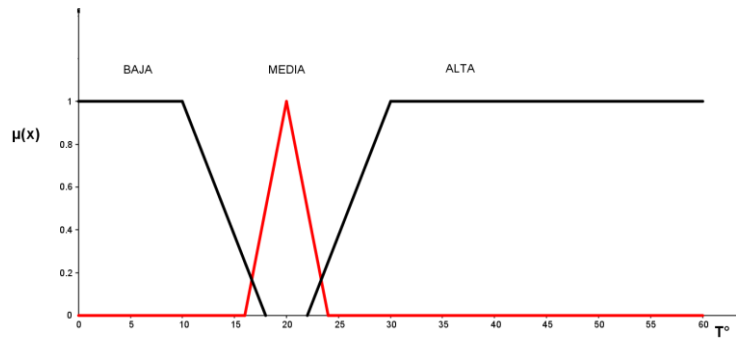


Fig. 2. Conjuntos difusos de la variable de entrada temperatura

Tabla 1. Coordenadas de los conjuntos difusos de la variable de entrada temperatura

Etiqueta	Temperatura ($^\circ\text{C}$)
U_{T°	[0, 60]
BAJA	[0, 0, 12, 18]
MEDIA	[16, 20, 24]
ALTA	[22, 30, 60, 60]

2.2.2 Humedad

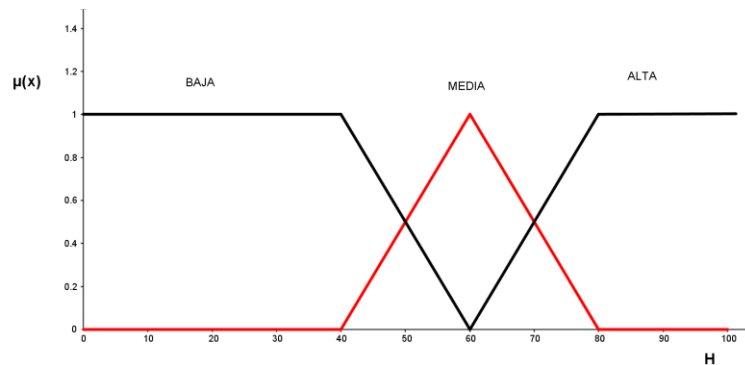


Fig. 3. Conjuntos difusos de la variable de entrada humedad

Tabla 2. Conjuntos difusos de la variable de entrada humedad

Etiqueta	Humedad relativa (%)
U_H	[0, 100]
BAJA	[0, 0, 40, 60]
MEDIA	[40, 60, 80]
ALTA	[60, 80, 100, 100]

2.3 Variable de salida:

2.3.1 Ventilación forzada

La variable de salida correspondiente al control de la ventilación forzada se expresa en porcentaje de potencia (Tabla 3).

Tabla 3. Conjuntos difusos de la variable de salida ventilación

Etiqueta	Potencia %
U_V	[0, 100]
APAGADO	[0]
MEDIA	[50]
ALTA	[100]

2.3.2 Iluminación incandescente

La variable de salida correspondiente al control de iluminación es binaria (Tabla 4).

Tabla 4. Conjuntos difusos de la variable de salida iluminación

Etiqueta	Estado
U_V	[0, 1]
APAGADO	[0]
ENCENDIDO	[1]

2.4 Reglas difusas

La base de reglas difusas, se diseñó en función a lo indicado por un experto en Biología Vegetal que aportó cuales son las relaciones óptimas de las variables intervinientes en el proceso de germinación y crecimiento de cultivos hidropónicos

1. If (temperatura is baja) and (humedad is baja) then (ventilador is apagado) (luz_caliente is encendido)

2. If (temperatura is baja) and (humedad is media) then (ventilador is apagado) (luzcaliente is encendido)
3. If (temperatura is baja) and (humedad is alta) then (ventilador is 50%) (luz_caliente is encendido)
4. If (temperatura is baja) and (humedad is alta) then (ventilador is 100%) (luz_caliente is encendido)
5. If (temperatura is media) and (humedad is baja) then (ventilador is apagado) (luz_caliente is apagado)
6. If (temperatura is media) and (humedad is media) then (ventilador is 50%) (luz_caliente is apagado)
7. If (temperatura is media) and (humedad is alta) then (ventilador is 100%) (luz_caliente is apagado)
8. If (temperatura is alta) and (humedad is baja) then (ventilador is 100) (luz_caliente is apagado)
9. If (temperatura is alta) and (humedad is media) then (ventilador is 100%) (luz_caliente is apagado)
10. If (temperatura is alta) and (humedad is alta) then (ventilador is 100%) (luz_caliente is apagado)

3 Diseño del software del controlador difuso tipo Sugeno

A efectos de definir las subrutinas se presenta un ejemplo de la función que permite calcular la pertenencia de un conjunto difuso triangular, definido por los puntos a, b, c y un valor de la variable de entrada dada por x, se declara el tipo de dato a devolver. En este caso PE_T, todas las variables se definen como tipo *float*.

```
float PE_T(float x, float a, float b, float c)
{
    float pertenencia;
    if (x<a or x>=c)
    {
        pertenencia=0;
    }
    if (x>=a and x<b)
    {
        pertenencia=(x-a)/(b-a);
    }
    if (x>=b and x<c)
    {
        pertenencia=(c-x)/(c-b);
    }
    return pertenencia;
}
```

El cuerpo principal del programa de control difuso de la cámara de germinación comienza con la lectura de puertos y sensores para obtener los valores de las variables del sistema. A continuación, con los valores de las variables de entrada H y T se procede a calcular con las funciones de pertenencia y determinar cuáles son las reglas activas. Como las reglas se componen de la siguiente forma R_n : si x_1 es A_1 e y_2 es B_2 entonces la salida desfusificada z es:

$$z = \frac{x_1 \omega_1 + y_1 \omega_2}{\omega_1 + \omega_2} \quad (1)$$

En donde ω_1 y ω_2 son los pesos de las salidas difusas de las reglas activas.

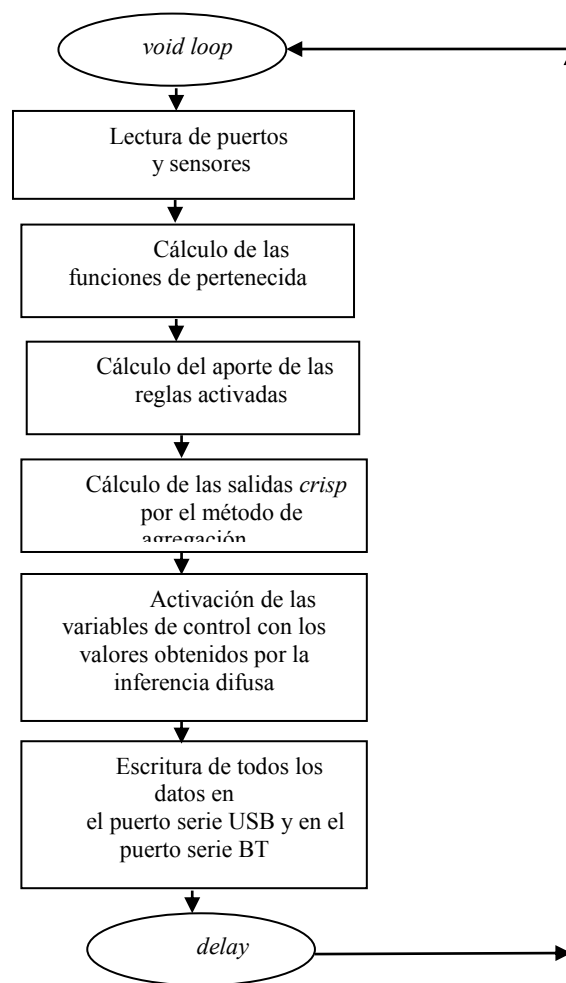


Fig. 4 Diagrama de flujo de la rutina *loop*

Una vez obtenidas las variables de control se procede a enviar las señales correspondientes a la placa de potencia. Luego se envían los valores de las variables por medio de los dos canales de comunicación: el puerto USB y el módulo BT. Antes de regresar al principio del ciclo se activa un retardo de un segundo como indica el diagrama de flujo de la Figura 4.

4 Resultados

Para evaluar la efectividad de la cámara de germinación, se realizó una comparación con un sistema de raíz flotante convencional, montado en una cuba en un recinto adiabático con temperatura controlada por un aire acondicionado y un panel de tubos fluorescentes que proveen de iluminación artificial constante. Se suministraron nutrientes hidropónicos en la misma proporción en ambos recipientes. Para verificar la cantidad de nutrientes presente se midió la CE, con un conductímetro digital marca TDS, obteniendo un valor de 2000 ppm.

Se realizaron mediciones de intensidad lumínica con un luxómetro marca Hepta-instrumentos (CEM), que cuenta con certificación de calibración emitida por el INTI. En la cuba se midieron 1150 lux y en el interior de la cámara 2100 lux.

En la Figura 5, puede apreciarse la diferencia en la calidad de las plántulas en ambos sistemas de germinación.

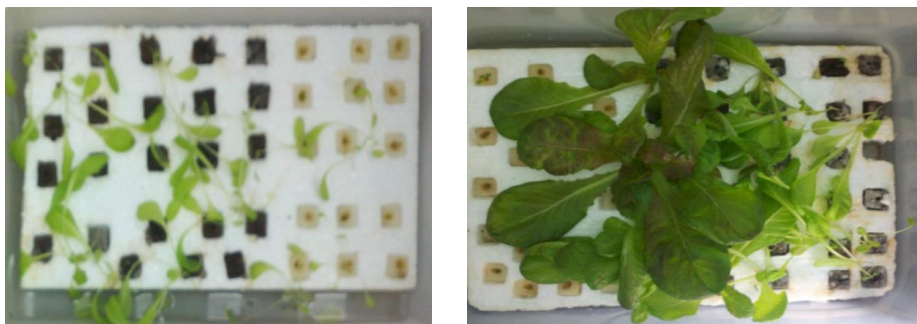


Fig. 5 Evolución de las plántulas a) en una cuba, b) en la cámara de germinación.

Los datos de temperatura y humedad relativa, de cada sistema de germinación se aprecian en los gráficos de la Figuras 6 y 7, como así también, la estabilidad de humedad y temperatura dentro de la cámara.

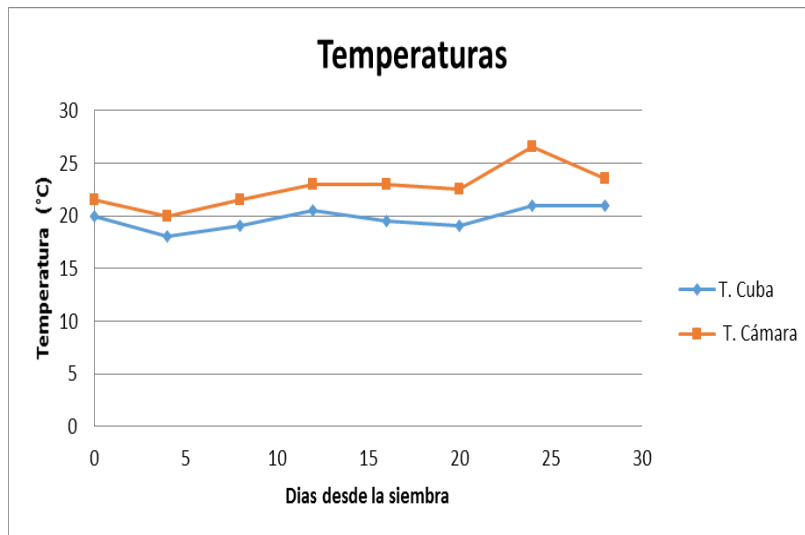


Fig. 6. Temperatura de la cámara en función de los días de siembra

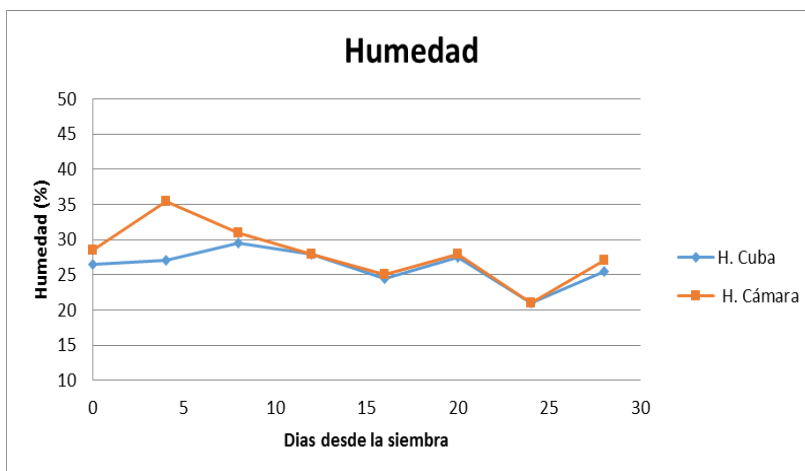


Fig. 7. Humedad de la cámara en función de los días de siembra

En la Figura 8 se observa una plántula extraída de la cámara, la cual tiene una longitud de 30 cm, una radícula de 13 cm y una cantidad de hojas de entre 5 y 7, lo que supera ampliamente las de la cuba a cielo abierto. Estos valores son superiores a los reportados en la literatura por otros autores, tanto en soluciones comerciales como con las fabricadas con el residuo de la destilación para la fabricación de alcohol [5]. Por otra parte, se realizaron experiencias con aplicación de distintos productos bioactivos, en suelo ferralítico rojo compactado, consiguiendo, en promedio, entre 3,97 y 5,71 hojas por planta en suelo a los 30 días [6], valores inferiores a los obtenidos en este trabajo.



Fig. 8. Cultivo obtenido a los 30 días.

Los costos de control, alcanzados en este trabajo, son inferiores a los de otros sistemas hidropónicos como el tipo “Dynamic Root Floating Hydroponic” (DRF) [7] que utilizan sistemas de control digitales, siendo además de difícil implementación en pequeños productores. Otros sistemas de control que utilizan lógica difusa en sistemas embebidos. [8] no muestran resultados de campo mientras que otros investigadores se concentran en mantener estable la cantidad de nutrientes y CO₂ [9], utilizando sensores de elevado costo.

5 Conclusiones y trabajos futuros

Se diseñó un controlador difuso tipo Sugeno para el control de una cámara de germinación hidropónica con el asesoramiento de expertos en biología vegetal del InBioMis. Se utilizó una placa Arduino Mega 2560 para la adquisición de datos y procesamiento de los mismos. Las variables de entrada corresponden a valores analógicos provienen de sensores de humedad y temperatura que se procesan mediante el software de control difuso, generando las señales de control de ventilación forzada e iluminación incandescente.

Los resultados obtenidos muestran que el controlador difuso provee un control del ambiente dentro de la cámara que mejora la producción de los cultivos.

Como trabajo futuro, se propone realizar controladores basados en lógica difusa diseñados para otros tipos de dispositivos de cultivos hidropónicos; como los de circulación película de nutrientes, y del tipo de raíz flotante con agitación de nutrientes.

Referencias

1. Zadeh L.: The concept of a linguistic variable and its application to approximate reasoning". Information Sciences, 1(8), 199–249 (1975)
2. Nguyen H., Prasad N.: Fuzzy Modeling and Control: Selected Works of Sugeno. CRC Press, Boca Raton, Estados Unidos. (1999)
3. Mohamed M. & Alsadon A.: Influence of ventilation and sucrose on growth and leaf anatomy of micropropagated potato plantlets. Scientia Horticulturae, 123, 295–300. (2010)
4. Domingues D., Hideaki W . Takahashi B., Camara C. & Nixdorf. S.: Automated system developed to control pH and concentration of nutrient solution evaluated in hydroponic lettuce production. Computers and Electronics in Agriculture. 84, 53–61. (2012)
5. Dos Santos, J. D., Lopes da Silva, A. L., Da Luz Costa, J., Scheidt, G. N., Novak, A. C., Sydney, E. B., & Soccol, C. R.: Development of a vinasse nutritive solution for hydroponics. Journal of environmental management, 114, 8–12. doi:10.1016/j.jenvman.2012.10.045 (2013)
6. Elein Terry Alfonso, Josefa Ruiz Padrón, Tamara Tejada Peraza, Inés Reynaldo Escobar, M. M. D. de A.: RESPUESTA DEL CULTIVO DE LA LECHUGA (Lactuca PRODUCTOS BIOACTIVOS CROP RESPONSE OF LETTUCE (Lactuca sativa)., 28–37.(2010)
7. Kao, T. THE DYNAMIC ROOT FLOATING HYDROPONIC TECHNIQUE : YEAR-ROUND PRODUCTION OF VEGETABLES IN ROC ON TAIWAN, (1987)
8. Ponce, P., Pontecorvo, F. L., Tena, C., Muñoz, K. C., Valencia, D. Y., Salzar, G. I., & Rojas, M. A.: Mixed Hidropony Automatization Based on a Fuzzy Logic System, 1–6. (2005)
9. Liu, C., Li, Z., & Zeng, X.: Fuzzy control system for hydroponics cultivating. 2008 7th World Congress on Intelligent Control and Automation (pp. 7804–7808). IEEE. doi:10.1109/WCICA.2008.4594146. (2008)

Análisis de impacto en la implementación de DNSSEC en un servidor DNS Recursivo

Ernesto Sánchez¹, Daniel Arias Figueroa¹, Sergio Rocabado¹, Javier Díaz²

¹ Centro de Investigación en Informática Aplicada (C.I.D.I.A.). Universidad Nacional de Salta. Argentina.

² Universidad Nacional de la Plata. Provincia de Buenos Aires. Argentina
{esanchez, daaf, srocahad}@cidia.unsa.edu.ar

Abstract. Las extensiones de Seguridad para DNS (DNSSEC) proveen autenticación del origen e integridad de los datos intercambiados a través del protocolo DNS. Las mejoras que ofrece DNSSEC radican principalmente en el uso de una jerarquía de “firmas criptográficas” que permite proteger el flujo de información intercambiado entre Servidores Autoritativos, Servidores DNS Recursivos y Clientes DNS. En el presente trabajo se muestra la perspectiva desde un cliente DNS, es decir, como es el comportamiento y que costo en términos de tiempos de respuesta y carga de tráfico, de un servidor DNS Recursivo que debe realizar validaciones DNSSEC. Para poder exponer los aspectos antes citados, se comparó el comportamiento de un servidor DNS Recursivo tradicional, (sin características de seguridad), frente a un servidor DNS Recursivo DNSSEC

Keywords: DNSSEC, Extensiones de Seguridad para DNS, Estudio comparativo, Sistema de Nombres de Dominio, Universidad Nacional de Salta

1 Introducción

El despliegue a nivel mundial de las extensiones de seguridad para el Sistema de Nombres de Dominio (DNSSEC) ha alcanzado a la fecha el 88%, considerando que son 1405 los dominios de nivel superior en la zona raíz [1]. La implementación de tales extensiones ha sido relativamente lenta en términos de tiempo, si se tiene en cuenta que en el año 2010 se anunció el firmado de la zona raíz. Esto se debe principalmente a la naturaleza jerárquica del sistema DNS, donde cada eslabón de la cadena es administrado de forma independiente por entidades privadas, públicas, ISPs, etc. Otro aspecto crítico, que influye en la migración a DNSSEC, es la naturaleza del servicio de resolución de nombres, fundamental para el funcionamiento de Internet.

Otros aspectos no menos importantes, son los relacionados directamente con las nuevas características de DNSSEC, tales como, nuevos registros de recursos, criptografía de clave pública, proceso de validación de confianza entre zona padre y zona hijo, entre otros. Estos últimos influyen de manera directa en costos adicionales en términos de carga de tráfico y tiempos de respuesta en procesos de resolución de

nombres, donde se involucra a servidores DNS de diferentes características (De nivel superior, Autoritativos, Recursivos y Resolvers).

En el presente trabajo se muestra la perspectiva desde un cliente DNS, es decir, como es el comportamiento y que costo en términos de tiempos de respuesta y carga de tráfico, de un servidor DNS Recursivo que debe realizar validaciones DNSSEC. Para poder exponer los aspectos antes citados, se comparó el comportamiento de un servidor DNS Recursivo tradicional, (sin características de seguridad), frente a un servidor DNS Recursivo DNSSEC.

Las pruebas realizadas y resultados obtenidos, se obtuvieron en el marco de trabajo final de tesis de postgrado de la Universidad Nacional de la Plata. "Un estudio comparativo en extensiones de seguridad para el Sistema de Nombres de Dominio" (Expte 3300-2113/10-000)

2 Características de DNSSEC

En términos generales y según se describe en el RFC 4033: DNS Security Introduction and Requirements, las Extensiones de Seguridad para DNS (DNSSEC) proveen autenticación del origen e integridad de los datos intercambiados a través del protocolo DNS. Las mejoras que ofrece DNSSEC radican principalmente en el uso de una jerarquía de "firmas criptográficas" que permite proteger el flujo de información intercambiado entre Servidores Autoritativos, Servidores DNS Recursivos y Clientes DNS.

Para poder proveer los aspectos de seguridad anteriormente citados, DNSSEC hace uso de nuevos Registros de Recursos y una particular infraestructura de clave pública, basada en la construcción de una "cadena de confianza", necesaria para la validación de los datos en el proceso de consulta/respuesta DNS.

2.1 Nuevos Registros de Recursos

Los Registros de Recursos para DNSSEC son: [2].

- **DNSKEY:** Registro de Recurso habilitado para almacenar claves públicas, que posteriormente serán usadas por DNSSEC en procesos de autenticación.
- **RRSIG:** Contiene la firma para un conjunto de Registros de Recursos (RRset) con un nombre particular, clase y tipo. El registro RRSIG se genera en el proceso de firmado de una zona utilizando la clave privada y cuyo par (clave pública) es almacenada en el registro DNSKEY.
- **NSEC:** Permite validar la estructura de una zona y los Registros de Recurso que esta contiene.
- **DS:** Permite crear una cadena de confianza o de autoridad de una zona padre firmada, hacia una zona hija firmada. DS está relacionado con el Registro

DNSKEY, ya que contiene un resumen (hash o digesto) de la clave (KSK) almacenada en éste último.

2.2 Cadena de confianza

El proceso de construcción de una cadena confianza es fundamental para la implementación de DNSSEC en una jerarquía DNS, ya que sin ésta característica, cada Servidor Recursivo configurado con DNSSEC, debería tener un punto de entrada seguro (SEP) por cada dominio seguro en Internet, lo que claramente haría imposible un despliegue a escala global de tales extensiones de seguridad. [9].

La siguiente ilustración permite observar los procesos involucrados en la creación de la cadena de confianza:

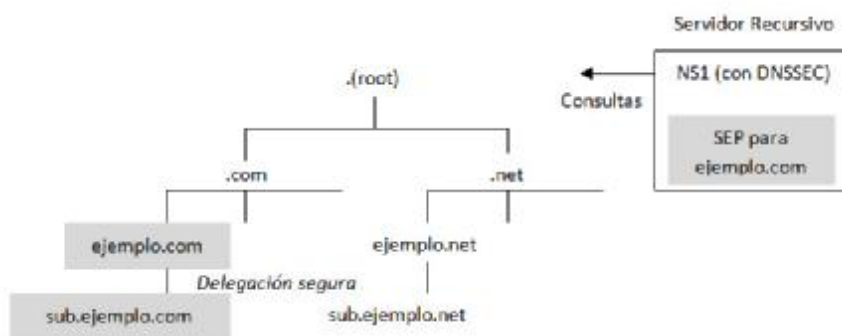


Fig. 1. Creación de cadena de confianza

Tanto el dominio ejemplo.com como sub.ejemplo.com se encuentran asegurados, es decir que para que pueda ocurrir una delegación segura es requisito previo haber asegurado la zona hija (sub.ejemplo.com). El punto de entrada seguro para ejemplo.com cubre las zonas seguras que son delegadas a partir de él, a través de una delegación segura creando una cadena de confianza provista por el uso del Registro de Recurso DS.

Una cadena de confianza puede ser construida tanto hacia arriba como hacia abajo en una jerarquía DNS, por lo que si el dominio de nivel superior .com fue asegurado, el dominio ejemplo.com puede unirse a la cadena.

Continuando con el ejemplo de la figura, el servidor NS1 (configurado con DNSSEC), podría ahora requerir un nuevo SEP para el dominio .com, y este único SEP cubriría ahora los dominios .com, ejemplo.com, así como sub.ejemplo.com. Desde Julio 2010 la zona raíz se encuentra firmada y a la fecha, son 1257 los dominios de nivel superior que fueron firmados, de los cuales, 1245 tienen puntos de anclaje seguros publicados como registros DS en la zona antes mencionada [1]. Es importante resaltar que en Junio del año 2015, el dominio .ar, se encuentra asegurado y publicado en la zona raíz. [3].

2.3 Clave de Zona (ZSK) y Clave de Claves (KSK)

En los procesos de delegación y posterior validación de claves criptográficas de firmado, la siguiente clasificación de claves se hace necesaria a fin de facilitar las tareas operacionales llevadas a cabo por DNSSEC. Según se describe en el RFC 4641 [4], las claves usadas para el firmado de registros asociados a un dominio pueden ser de dos tipos, ZSK (Zone Signing Key) o KSK (Key Signing Key), donde la primera tiene por función la de proteger los Registros de Recursos individuales de una Zona dada, mientras que la KSK se encarga de proteger la ZSK. Operacionalmente se almacenan en un registro DNSKEY y se distinguen mediante el bit llamado SEP, presente en la porción RDATA del Registro de Recurso DNSKEY.

Algunas de las motivaciones para un uso separado de claves son: La KSK puede configurarse con longitudes de clave mayores, lo que la convierte en una clave de mayor fortaleza. Operacionalmente tiene poco impacto en consumo de recursos, ya que solo se usa para el firmado de una pequeña porción de datos de una zona dada. Por otro lado, dado que la KSK sólo se utiliza para firmar un conjunto de claves, ésta puede actualizarse con menos frecuencia que otros datos en la Zona y ser almacenada en una localización diferente de la ZSK.

3 Escenario de prueba utilizado

Según se expuso en 2.2, para poder garantizar la autenticidad e integridad de una respuesta DNS, en primera instancia se debe verificar la cadena de confianza desde el último eslabón de la cadena, hasta llegar al nodo raíz, es por esto que se configuró una jerarquía DNS, utilizando la herramienta de virtualización VMWare Player, según se muestra en la siguiente figura. Para la obtención de los datos se crearon dos instancias, la primera, basada en DNS tradicional y la segunda con extensiones de seguridad (DNSSEC)

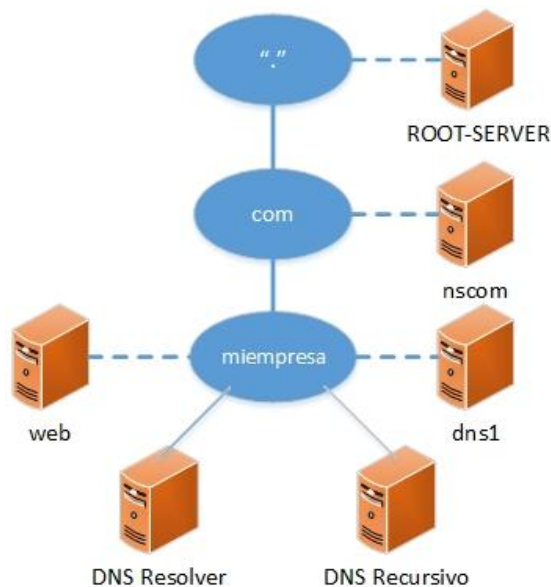


Fig. 2. Escenario de prueba

La siguiente tabla muestra detalles de la configuración utilizada, tales como direccionamiento IP y funcionalidad.

Tabla 1. Configuración escenario de prueba.

Nombre servidor	Dirección IP	Observaciones
ROOT-SERVER	10.0.0.254	Servidor DNS Raíz
nscom	10.0.0.253	Servidor TLD .com
dns1	10.0.0.2	Servidor Autoritativo miempresa.com
web	10.0.0.100	Servidor web miempresa.com
recursivo	10.0.0.11	Servidor Recursivo Cache
resolver	10.0.0.12	Cliente DNS

Todos los servidores están basados en Debian Linux con software DNS BIND versión 9.8.4

3.1 Generación de consultas DNS tradicional

Para el análisis de tráfico DNS se generaron consultas al dominio www.miempresa.com desde el servidor resolvente, el cual retransmite las mismas al servidor recursivo. El primero ejecuta la herramienta Wireshark para la captura de los

paquetes de datos intercambiados. La siguiente tabla resume los datos que ilustran dos situaciones, donde la primera, el servidor recursivo, para responder a la solicitud del resolver, debe consultar al resto de los servidores, (Sin cache), mientras que en el segundo caso, responde a la consulta con los datos ya almacenados en cache, como consecuencia del primer caso.

Tabla 2. Datos obtenidos DNS tradicional.

DNS Tradicional	Consultas	Tiempo de rta	Bytes enviados	Bytes recibidos
Sin cache	4	0,0108 seg	341	514
Con cache	1	0,0039 seg	77	128

3.2 Configuración de escenario DNSSEC

Tomando como base el escenario presentado en la figura 2, se implementó DNSSEC en todos los servidores involucrados en la jerarquía DNS, es decir que, se configuró el servidor recursivo para validación DNSSEC, se firmaron zona raíz, zona .com y zona miempresa.com. El proceso se completó creando la cadena de confianza a partir de la publicación del registro DS desde zona hija a zona padre.

La implementación de DNSSEC para el escenario propuesto, se resume en los siguientes pasos:

- Generación de claves de firmado pública/privada ZSK y KSK.
- Publicación de claves pública ZSK y KSK en archivo de zona.
- Firmado de archivo de zona con clave privada ZSK.
- Publicación de archive DS en zona padre. Esto se realizó de zona .com a zona raíz y de zona miempresa.com a zona .com
- Refirmado de zona padre.
- Configuración de clave pública de validación inicial (KSK), para todos los servidores de la jerarquía, incluidos recursivo y resolver. Esta clave se obtuvo de lo generado en zona raíz (ROOT-SERVER).

Cabe resaltar que la configuración DNSSEC se realizó utilizando los comandos nativos provistos por BIND, siendo otras alternativas posibles, DNSSEC Tools, [6] y OpenDNSSEC, [7].

A modo de ejemplo se presenta la siguiente figura, la cual describe el recorrido de la cadena de confianza, la cual resulta de la configuración antes descripta.

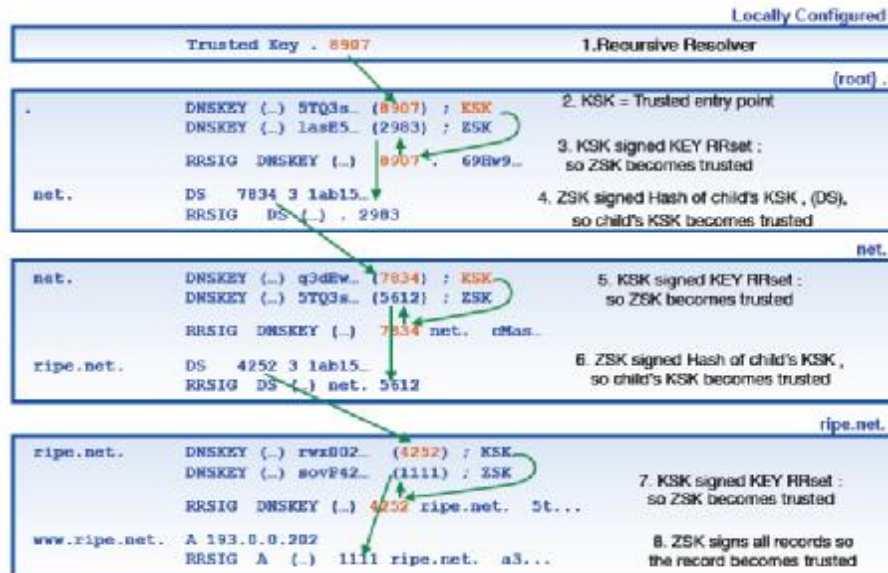


Fig. 3. Recorrido de cadena de confianza DNSSEC. [5]

3.3 Generación de consultas DNSSEC

Para la obtención de los datos en el proceso de consulta/respuesta, se usó la misma metodología del escenario DNS tradicional, es decir se generó una solicitud al dominio www.miempresa.com, en primera instancia sin cacheo y la segunda ya con datos almacenados en cache.

De los datos obtenidos de las pruebas, surge la primera observación importante, la misma se corresponde con el proceso de validación de la cadena de confianza, la cual se realiza en sentido contrario al proceso de consulta, la siguiente figura ilustra a modo de ejemplo dicha observación:

- **DNS Query goes from top down**



- **DNSSEC Validation goes bottom up**



Fig. 4. Flujo de consulta y validación DNSSEC. [8]

Finalmente y con el propósito de realizar un estudio comparativo en cuanto a carga de tráfico y tiempos de respuesta, se registraron los siguientes datos:

DNSSEC	Consultas	Tiempo de rta	Bytes enviados	Bytes recibidos
Sin cache	9	0,0297 seg	727	6705
Con cache	1	0,0024 seg	77	128

Partiendo desde un estado en donde la cache local del servidor recursivo se encuentra vacía, se necesitan nueve consultas con un tiempo total para la resolución de la consulta de 0,0297 seg, con 727 bytes enviados y 6705 bytes recibidos. Lo que representa tres veces más de tiempo de respuesta, mientras que para el tráfico enviado el incremento es de 2,5 veces más y otras 10 veces más de carga de tráfico de salida, en comparación con el escenario DNS tradicional. En cuanto los datos de la consulta son almacenados en cache, la performance para la resolución mejora notablemente, demostrando que la funcionalidad de almacenamiento en cache es sumamente útil.

4 Conclusiones

Desde la perspectiva de un cliente DNS recursivo, se distinguen dos características fundamentales, la primera es que la funcionalidad de almacenamiento en cache, “absorbe” de manera eficiente, el impacto de la implementación de las extensiones de seguridad para DNS. Sin embargo, la puesta en marcha de tales aspectos de seguridad no se alcanza de manera inmediata, ya que requiere estar altamente familiarizado con el protocolo y el funcionamiento del mismo.

A la fecha, siguen surgiendo casos de implementación y documentación al respecto, lo que sumado a la adopción para el dominio .ar, permitirán continuar con las pruebas de performance sobre tráfico de consulta/respuesta sobre Internet.

Referencias

1. ICANN Research, TLD DNSSEC Report, http://stats.research.icann.org/dns/tld_report/
2. ARENDS, R., AUSTEIN, R., LARSON, M., MASSEY, D., AND ROSE, S. RFC 4034: Resource Records for the DNS Security Extensions, Marzo 2005.
3. NIC Ar. <https://nic.ar/Enterate/Noticias/primer-ceremonia-de-dnssec-para-la-zona-ar>
4. KOLKMAN, O, GIEBEN, R. RFC 4641: DNSSEC Operational Practices, Setiembre 2006.
5. RIPE Network Coordination Centre. DNSSEC Training Course. <https://www.ripe.net/support/training/material/dnssec-training-course/DNSSEC-Slides-Single.pdf>.
6. DNSSEC Tools Project. <http://www.dnssec-tools.org/>
7. Open DNSSEC Project. <https://www.opendnssec.org/>.
8. Sudan Network Operators Group. Hands on DNS and DNSSEC. <http://www.sdnog.sd/images/SdNOG-2/DNSSEC/DNSSEC-course.pdf>.
9. Ron Aitchison: Pro DNS and BIND. APRESS, New York (2011)

Transporte de datos Multicast Confiable con Control de Congestión Dinámico

Gabriel Gomiz – Luis Marrone

Laboratorio de Investigación en Nuevas Tecnologías Informáticas
Facultad de Informática - UNLP
Calles 50 y 120, 1900 La Plata, República Argentina
gabriel.gomiz@gmail.com
lmarrone@linti.unlp.edu.ar
<http://www.linti.unlp.edu.ar>

Resumen La naturaleza de la transmisión multicast hace que sea un mecanismo no orientado a conexión, por lo que la mayoría de los protocolos multicast se desarrollan sobre la capa de transporte UDP y sobre este es necesario construir un esquema de confiabilidad. Para aplicaciones de “streaming” de audio o vídeo, la pérdida ocasional de un segmento es aceptable, pero al transmitir datos críticos (archivos, programas, etc.) es necesario un mecanismo que garantice confiabilidad. Por lo tanto se presenta en esta publicación una propuesta de transmisión multicast confiable con control de congestión, como alternativa a los mecanismos disponibles en la actualidad.

Keywords: multicast, PGM, congestion, ACK

1. Introducción

Muchas aplicaciones necesitan semánticas confiables de transmisión de datos para funcionar correctamente y recibir una secuencia de paquetes sin pérdidas, ni duplicados y en el orden exacto en el que fueron transmitidos. Un protocolo de transporte confiable resuelve estas cuestiones y oculta dicha complejidad proporcionando una abstracción confiable a las aplicaciones.

Todos estos protocolos utilizan las mismas ideas para cumplir con los requisitos de confiabilidad: redundancia para resolver pérdidas de paquetes, almacenamiento (buffers) en los receptores para poder reordenar y el uso de temporizadores y retransmisiones. Debido a que las aplicaciones grupales de comunicación tienen una amplia variedad de requerimientos de confiabilidad, se han desarrollado múltiples protocolos multicast confiables, ninguno de los cuales es un estándar dominante así como lo es TCP para las transmisiones unicast confiables. [13]

Existen, fundamentalmente, dos aproximaciones para los protocolos de Multicast confiable:

1. ACK: hacer que todos los receptores confirmen la recepción de los paquetes
2. NACK: hacer que los receptores solo informen cuando detectan pérdida de paquetes

Si es necesario conocer que absolutamente todos los receptores recibieron todos los datos, entonces es probable que se necesite un protocolo basado en ACKs. Por otro lado, si para la aplicación en cuestión no es importante si alguno de los receptores no recibió la totalidad de los datos, entonces los protocolos basados en NACKs pueden ser mas adecuados ya que escalan mucho mejor.

1.1. Mecanismos basados en ACKs

Debido a que un grupo multicast puede tener un arbitrario número de miembros y a que los protocolos tradicionales utilizan acks para garantizar confiabilidad, esto requeriría que el transmisor pueda manejar un número arbitrario de acks. Desafortunadamente, esto deriva en un problema conocido como “ACK implosion”. Este problema ha sido el foco de múltiples investigaciones en los últimos años y existen varias estrategias para solucionarlo o, al menos, intentar disminuirlo.

Para superar el problema de la implosión de las confirmaciones, los protocolos de multicast confiable aplican una aproximación jerárquica en donde el multicast se restringe a una única fuente. Antes de enviar los datos, se establece un árbol de reenvío (“forwarding tree”) desde la fuente hasta todos los

miembros del grupo y se determinan puntos de confirmación (“acknowledgement points”). Un punto de confirmación consiste de un router en el árbol de reenvío que acepta realizar copias de caché de los datos y procesar las confirmaciones de los routers o las estaciones que estén por debajo en el árbol. Si se requiere una retransmisión, el punto de confirmación obtiene una copia desde su caché.

Otra estrategia para mejorar los protocolos basados en ACK es la de combinar varios ACKs en un solo paquete.

1.2. Mecanismos basados en NACKs

Muchos esquemas de multicast confiable, en vez de utilizar confirmaciones positivas (ACKs), utilizan confirmaciones negativas (NAKs); esto quiere decir que las estaciones receptoras no responden salvo que algún datagrama se pierda o no sea recibido. Este esquema permite una mejor escalabilidad para el ambiente en el que usamos multicast ya que se pasa la carga de control de errores del transmisor a los destinatarios. La estación que transmite envía los datagramas sin esperar las confirmaciones. La idea es evitar el envío de mensajes de estado (ACKs) cuando todo está normal y así mejorar la eficiencia del protocolo [3].

Para permitir a una estación detectar que un datagrama no ha sido recibido, se debe asignar a cada uno un número único de secuencia. Cuando la estación detecta una pérdida envía un NACK para solicitar la retransmisión del datagrama perdido. El NACK debe ser propagado hacia arriba en el árbol (en dirección a la fuente) hasta que el mismo llega a un punto de confirmación. En este punto se procesa el NACK y se retransmite el datagrama hacia abajo en el árbol.

Cada punto de confirmación utiliza el mismo esquema para asegurarse de que tiene una copia de todos los datagramas de la secuencia. Eventualmente, al seguir cada uno de los puntos de confirmación, llegaremos a la fuente de la transmisión.

Con esta aproximación, el diseño de la topología y los puntos de confirmación son cruciales para el éxito del esquema de multicast confiable. Sin los suficientes puntos de confirmación, un datagrama perdido puede causar una implosión de NACKs. En particular, si un router tiene demasiados descendientes, un datagrama perdido puede causar que el mismo sea sobrecargado con pedidos de retransmisión. Desafortunadamente, la selección automática de los puntos de confirmación es un problema complejo y muchos protocolos multicast confiables requieren configuración manual. Por esto, la transmisión multicast se aplica mejor a servicios que tienden a existir por largos períodos de tiempo, topologías que no cambian rápidamente y situaciones donde los routers intermedios acuerdan servir como puntos de confirmación [4].

Como vemos, los mecanismos basados en NACKs pueden sufrir también problemas de implosión, los cuales son resueltos con técnicas de supresión de NACKs y/o soluciones basadas en temporizadores, como hace por ejemplo el protocolo SRM.

1.3. Grupo de trabajo RMT

La IETF formó el grupo RMT (Reliable Multicast Transport Working Group) para establecer un conjunto estándar de protocolos de transporte multicast confiables. Este grupo creó un conjunto de directivas que los desarrolladores de los protocolos propuestos deberían satisfacer. Una vez implementados, se espera que sean de uso general para una amplia variedad de aplicaciones.

Es claro que diferentes aplicaciones van a requerir diferentes aspectos de confiabilidad. A diferencia del modelo de transporte unicast, donde TCP es prácticamente universal, es poco probable que un único protocolo de transporte multicast confiable sea aceptable para todo tipo de aplicaciones. El foco del grupo RMT apunta a tres tipos de módulos:

- “Nack Oriented Reliable Multicast Protocol” (NORM) [5] que utiliza confirmaciones negativas (NAKs) para garantizar confiabilidad. Este protocolo evita el problema de la implosión de los NAKs haciendo que todos los receptores también escuchen y procesen los NAKs de otras estaciones y/o utilizando demoras probabilísticas para evitar NAKs redundantes. NORM puede trabajar en modelos multicast con un único transmisor y también con múltiples transmisores. Este protocolo seguramente será más útil en aplicaciones de transferencia de archivos.

Pros: simplicidad; el emisor no necesita conocer la cantidad de receptores; el emisor no necesita llevar cuenta del estado de los receptores; escalabilidad para aplicaciones y topologías de red donde es prohibitivo construir una infraestructura de entrega multicast sobre la capa básica IP multicast. [12]

Cons: como varios de los protocolos basados en NAKs, es difícil para el emisor saber cuando puede liberar los buffers de transmisión; se necesitan mecanismos adicionales a nivel de sesión si el emisor necesita saber si un determinado receptor ha recibido todos los datos.

- “Tree ACKnowledgement based protocol” (TRACK) [6] que utiliza un árbol para controlar la retro-alimentación y las *reparaciones* (retransmisión de paquetes perdidos). Este protocolo tiene la característica de devolver al transmisor una confirmación de cada receptor. Esto puede ser importante, por ejemplo, en aplicaciones financieras donde el receptor *debe pagar* por los datos. En este modelo el transmisor necesita una confirmación de que puede *cobrar* a los receptores. TRACK puede no ser una buena alternativa en redes donde el camino de retorno del receptor al transmisor no sea confiable.
- “Asynchronous Layered Coding” (ALC) [7] que utiliza técnicas de forward error-correction (FEC) y no requiere de retro-alimentación.

2. Alternativas de transporte multicast

2.1. Multicast IP

Pros: estándar y altamente disponible. Respeto la filosofía *extremo a extremo*.

Cons: no tiene garantías de confiabilidad, no es *amigable* con los routers

2.2. Scalable Reliable Multicast (SRM)

Scalable Reliable Multicast (SRM)[8] es un marco de trabajo de multicast confiable para usar a nivel aplicación y orientado a sesiones. En este marco de trabajo todo el tráfico es multicast. Este marco de trabajo puede ser resumido describiendo sus 3 tipos de paquetes:

- **latidos (heartbeats):** cada miembro envía periódicamente un heartbeat que incluye el número de secuencia del último paquete enviado. Estos paquetes son utilizados por los demás miembros para poder detectar la pérdida de paquetes, comparando el número de secuencia del heartbeat con el número de secuencia del último paquete recibido.
- **NACKs:** cuando se detecta un paquete perdido, se envía un paquete NACK a todos los miembros. Como todos los miembros ven este NACK, todos pueden participar en la reparación de los datos.
- **reparación (repair):** cada miembro de la sesión mantiene un caché con los últimos paquetes de datos recibidos (y enviados) y si reciben un NACK de un paquete que tienen en el caché, retransmiten el paquete a todo el grupo como una reparación.

Para minimizar el número de NACKs y reparaciones, estas dos operaciones se preceden por un “exponential back-off”. Esto significa que en vez de enviar el paquete directamente, el cliente debe esperar un tiempo aleatorio (basado en la distancia al nodo en cuestión) y si en este tiempo otro nodo envía el paquete correspondiente, este cancela su propia transmisión.

SRM está diseñado para cumplir con la definición minimal de multicast confiable, esto es, entrega eventual de todos los datos a todos los miembros del grupo, sin asegurar un determinado orden de entrega. Los autores creen que si fuera necesario asegurar un determinado orden de entrega, sería fácil de agregar una capa sobre SRM que de esto se ocupe.

SRM está fuertemente basado en el modelo de entrega de grupo que es la pieza central del protocolo multicast IP. En multicast IP, las fuentes de datos simplemente envían a la dirección multicast del grupo sin necesitar conocimiento previo de los miembros del grupo.

Para poder recibir datos enviados al grupo, los receptores anuncian que están interesados a través de un mensaje *multicast join* en la subred local y no necesitan conocimiento previo de los miembros del grupo y/o transmisores activos. Cada receptor se une o deja el grupo de manera individual, sin afectar la transmisión de los datos a cualquier otro miembro.

SRM extiende el concepto de grupo multicast maximizando la información y los datos compartidos entre todos los miembros, y fortalece la individualidad de la pertenencia haciendo a cada miembro responsable por la correcta recepción de sus propios datos.

Finalmente, SRM intenta seguir los principios centrales del diseño de TCP/IP. Primero, SRM requiere sólo el modelo básico de entrega del protocolo IP – mejor esfuerzo con posible duplicación y/o re-ordenamiento de paquetes – y construye la confiabilidad en una base punto-a-punto. No se requiere ningún cambio o soporte especial de la red IP que existe por debajo. En segundo lugar, de una forma similar a TCP que setea temporizadores y ventanas de control de congestión, los algoritmos de SRM dinámicamente ajustan sus parámetros de control basándose en la performance observada dentro de una sesión. Esto permite a las aplicaciones que utilizan el marco de trabajo de SRM a adaptarse a un rango amplio de tamaños de grupo, topologías y anchos de banda de los enlaces mientras mantiene una performance alta y robusta. [8]

Pros: útil para replicar datos con poca probabilidad de cambios. No sufre de implosiones de ACK o NAK.

Cons: limitado a escalabilidad de mejor esfuerzo; no puede llevar cuenta de la pertenencia a los grupos; el throughput es inestable si la tasa de fallos de la red no es desestimable.

2.3. Reliable Multicast Transport Protocol (RMTP)

RMTP [9] provee una entrega ordenada y completa de una secuencia de datos desde una fuente a un grupo de receptores. RMTP está basado en una jerarquía de múltiples niveles en la cual los receptores se agrupan en jerarquías de regiones locales, con un receptor designado (DR) en cada región. RMTP define DRs que reúnen los mensajes de estado de los nodos de la región RMTP local y proveen las reparaciones (retransmisiones de datos perdidos), si estos están disponibles. Los receptores dirigen los mensajes administrativos usando unicast hacia el DR. De esta manera, el DR provee recuperación local y consolidación del tráfico de control al siguiente DR de la jerarquía si los datos solicitados no estuvieran disponibles.

RMTP provee entrega confiable de una única fuente a un grupo de receptores sin saber exactamente la identidad de los mismos. Los receptores envían sus mensajes de estado periódicamente al transmisor/DR que retransmite los paquetes que se puedan haber perdido. Dado que los ACKs se envían periódicamente desde los receptores, aunque uno de estos ACKs se pierda, el transmisor/DR no debe realizar nada especial, porque otro ACK reflejando el estado actualizado será generado por el mismo receptor. Por lo tanto, la generación periódica de mensajes de estado provee al protocolo RMTP de una característica inherente de tolerancia a fallos y no debe ser considerado como una sobrecarga ya que simplifica la recuperación de errores.

El enfoque jerárquico utilizado en RMTP junto con la decisión de diseño de no llevar cuenta explícitamente de los receptores, dan un alto grado de escalabilidad al protocolo. Si algunos receptores están muy lejos del transmisor, este no debe preocuparse por estos, ya que el DR correspondiente será responsable de manejar los ACKs y de retransmitir los paquetes perdidos. Además, una característica es clave para la escalabilidad de RMTP: la información de estado que se mantiene en el transmisor es independiente del número

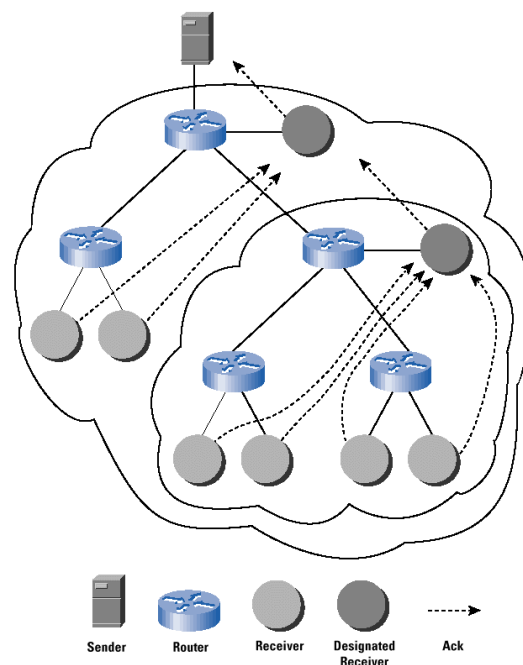


Figura 1. [10]

de receptores. El precio que RMTP paga por esta escalabilidad es el caché adicional que tienen que mantener tanto el transmisor como los DRs.

Pros: su estructura jerárquica brinda un comportamiento predecible; se adapta automáticamente a fallos en los servidores o a servidores que no pueden alcanzarse. Útil para transmisión de archivos multicast.

Cons: comparte las mismas características que SRM. Requiere que los DRs se definan estáticamente.

2.4. Pragmatic General Multicast (PGM-OPGM-PGMCC)

Pragmatic General Multicast (PGM) es un protocolo de transporte multicast confiable diseñado principalmente para aplicaciones que requieran transmisión de datos multicast ordenada y libre de duplicados desde múltiples transmisores hacia múltiples receptores.

La principal ventaja de PGM sobre los protocolos multicast tradicionales es que para cada receptor en el grupo, PGM garantiza que el mismo recibe todos los paquetes a partir de las transmisiones y posibles retransmisiones, o es capaz de detectar una pérdida de paquetes de datos que es irrecuperable.

PGM se diseñó específicamente como una solución para aplicaciones multicast cuyos requerimientos de confiabilidad sean básicos. Su meta central de diseño es la simplicidad de operación siempre teniendo en cuenta la escalabilidad del protocolo y la eficiencia en el uso de la red.

En PGM, no existe la noción de pertenencia a un grupo; se provee una entrega multicast confiable de datos dentro de una ventana de transmisión. El árbol de PGM se construye utilizando elementos de red (NEs) que manejan PGM sobre el árbol multicast existente. Una fuente PGM envía a los receptores paquetes de datos multicast (ODATA) en una secuencia. Cuando los receptores detectan paquetes que faltan en la secuencia, envían un paquete unicast a su padre en el árbol PGM. Los padres confirman la recepción del NAK a todos sus hijos con una confirmación multicast del NAK (paquete conocido como NCF). Los paquetes de reparación (RDATA) son generados por la fuente o por un reparador local designado (DLR) en respuesta a un NAK; esto significa que los NEs nunca guardan ODATA o proporcionan paquetes de reparación. Un paquete de reparación es el paquete perdido reenviado o un paquete FEC, dependiendo de los parámetros de la sesión. Antes de enviar un NAK, los receptores ejecutan un back-off aleatorio y suprimen el NAK si reciben el correspondiente NCF, los datos o los datos de reparación.

En cuanto a implementaciones de PGM es importante considerar **OpenPGM**. Es una implementación de código abierto del protocolo PGM que facilita el desarrollo de aplicaciones distribuidas escalables que intercambian datos a través de la red y con asistencia de la misma. OpenPGM es un transporte de datos, no impone ningún formato de mensaje o lógica de comunicación de alto nivel. OpenPGM puede ser ejecutado en varias plataformas de hardware y de software permitiendo así una plataforma de red heterogénea.

Otra optimización que realizan los receptores es demorar intencionalmente por un tiempo mínimo aleatorio la emisión de un NAK para evitar que varios receptores emitan el mismo NAK simultáneamente. Opcionalmente, es posible utilizar FEC (Forward Error Correction) donde se utilizan códigos de bloque lineales para generar h paquetes de paridad a partir de k paquetes originales de datos, de forma que tomando k paquetes cualquiera, del total de $(h+k)$ paquetes, se pueden decodificar los k paquetes originales.

En el modo básico de operación de PGM simplemente se realiza un límite en la tasa de transferencia del emisor. Para realizar control de congestión el emisor debe recibir información acerca del estado de la transmisión. Para este fin, PGM soporta 3 tipos opcionales de feedback al emisor:

1. carga del peor enlace (medido por los NEs)
2. carga del peor camino punto-a-punto (medido por los NEs)
3. carga del peor camino punto-a-punto (medido por los receptores)

Sin embargo, PGM no especifica como se ajustará esta tasa; el esquema de control de congestión se deja a criterio de la implementación.

Como trabajos relacionados, se han propuesto varios esquemas de control de congestión complementarios a PGM. Uno de éstos es llamado **PGMCC** [11]; esquema de control de congestión de tasa única cuyos objetivos se centran en la escalabilidad, estabilidad y rápida respuesta a la variación de las condiciones de la red.

En el esquema de tasa única que usa PGMCC, todos los receptores obtienen la misma tasa de datos y el emisor se adapta al receptor más lento del grupo. Se conoce que estos esquemas tienen la limitación de que un único receptor lento puede tirar abajo la tasa de envío para todo el grupo. El comportamiento es muy similar al control de congestión de TCP, ya que utiliza un ciclo de control basado en ventana que se ejecuta entre el emisor y un receptor seleccionado, llamado "ACKER". El rol del ACKER es proveer al emisor de una rápida retroalimentación de la misma forma que lo hace un receptor en TCP. La implementación de esta variante requiere modificaciones tanto en los emisores como en los receptores.

3. Nuestra propuesta - PGMCOOP

La utilización de PGM en esquemas 1 a N incorpora confiabilidad a las transmisiones, pero, al utilizar una tasa de transferencia fija, no permite aprovechar el ancho de banda completo que hubiera disponible en la red, ya que no posee un mecanismo para poder incrementarlo, así como tampoco disminuye el mismo en el caso de existir congestión.

Aquí se presenta una variante del protocolo, que llamaremos PGMCOOP, cuya novedad con respecto a PGM es incorporar un mecanismo de ajuste dinámico de la tasa de transferencia del emisor, en función de los NAKs / tasa de NAKs que va recibiendo. La simplicidad del diseño y la implementación es uno de los objetivos fundamentales de esta variante, y para cumplir con esto se buscó una estrategia donde solo sea necesario modificar la lógica de emisión y que el esquema sea transparente para los receptores que podrán seguir trabajando con el protocolo PGM puro.

Una estrategia que se utilizó antes de desarrollar PGMCOOP, fue usar un protocolo multicast desarrollado ad-hoc que llamamos internamente MDP. Se implementó sobre multicast UDP simple y tenía 2 etapas:

1. Envío de datos usando IP multicast con tasa fija de transferencia.
2. Fase de reparación posterior con estrategia unicast para cada receptor que necesite reparaciones.

El comportamiento del protocolo MDP con respecto a la cantidad de bloques perdidos es proporcional al crecimiento de la tasa de transferencia. A medida que incrementamos la misma y nos acercamos al límite de potencia de procesamiento de los receptores, los bloques perdidos se incrementan. Este comportamiento evidencia dificultades generadas por el uso de una parametrización estática (tasa de transferencia) y por no disponer de un mecanismo de control de congestión, además de la ineficiencia generada a partir de potencialmente repetir la transmisión de los mismos bloques de reparación en un esquema unicast. PGMCOOP intenta solucionar estos inconvenientes.

Para el control de congestión se prefirió utilizar una estrategia diferente a la de PGMCC, realizando modificaciones mínimas a PGM para poder cambiar la tasa de transferencia en forma dinámica y de esta forma intentar mejorar su comportamiento. Debido a que este protocolo será utilizado en múltiples escenarios que poseen distintas características de equipamiento de red y emisores y receptores con características totalmente heterogéneas, es necesario que el mismo pueda adaptarse a cada red regulando la tasa de transmisión para no sobrecargar a los receptores, evitando así que se produzcan demasiados NAKs y se deba realizar un alto número de retransmisiones.

Se adoptó una estrategia de "congestion avoidance", con un esquema basado en el algoritmo de incremento aditivo y reducción multiplicativa (AIMD). Esta estrategia evita la congestión combinando el crecimiento lineal de la ventana de congestión con una reducción exponencial de la tasa de transferencia cuando ésta es detectada.

Se utilizan los NAKs como indicadores de congestión, el emisor reducirá la tasa de transferencia por un factor multiplicativo; por ejemplo dividir a la mitad la ventana de transmisión. En ausencia de NAKs la ventana se incrementa linealmente. El resultado de esto es una curva con forma de dientes de sierra ya que este algoritmo va probando la red para detectar el ancho de banda disponible, de manera similar a lo que ocurre con el algoritmo para evitar congestión en TCP.

3.1. Diseño

Cada sesión de envío del emisor trabajará enviando datos a los receptores via paquetes de datos (ODATA) con PGM, con una ventana de transmisión $cwnd$, que le permitirá enviar sin restricciones hasta agotada dicha ventana. Luego del envío de cada bloque, el emisor deberá chequear cual es la tasa

actual de recepción de paquetes NAK. Si la tasa de NAKs es menor al umbral `nak_th` durante toda la duración de la ventana `cwnd`, entonces se ampliará la ventana de transmisión utilizando el factor lineal aditivo (*a*). Por el contrario, si se detecta una tasa de NAKs mayor al umbral `nak_th` en ese período, reduciremos la ventana de transmisión multiplicando la misma por el factor de decremento (*b*) y el emisor entrará en un estado de control de congestión (CC). En el estado de control de congestión, el emisor podrá seguir enviando datos, pero sin incrementar la ventana de transmisión `cwnd` y deberá controlar luego de la transmisión de cada paquete la tasa actual de NAKs; es decir que dentro del estado de control de congestión el límite `cwnd` no se utilizará; el emisor, sin embargo, utilizará una ventana de control de congestión `cc_cwnd`. Esta ventana se usa para “limpiar” el estado de congestión ya que el emisor deberá controlar que no haya recibido ningún paquete NAK durante esta ventana para poder salir del estado de control de congestión. Una vez que el emisor no haya recibido ningún NAK durante una ventana del tamaño definido para el control de congestión se podrá volver al estado normal de envío. Los siguientes parámetros que controlarán la transmisión serán configurables por parte del usuario/desarrollador que utilice las funciones del protocolo:

cwnd: tamaño inicial de ventana de transmisión

cc_cwnd: ventana de transmisión en la que no se deben recibir NAKs para salir de estado CC

nak_th: umbral límite de la tasa de NAKs

a: factor de incremento lineal ($a > 0$)

b: factor de decremento exponencial ($0 < b < 1$)

r: tasa de transferencia inicial de transmisión (valor predeterminado: 100.000 bps)

Algoritmo de envío A continuación se presenta, en lenguaje de pseudo-código, el algoritmo que se utilizará en la solución:

```

1: mientras existan datos para enviar
2:   pgm_send(sock, buffer)
3:   si estado==CC
4:     si nak_rate < nak_threshold
5:       si se completó cc_cwnd
6:         estado=NORMAL
7:       sino
8:         avanzar ventana cc_cwnd
9:     sino
10:      reiniciar ventana cc_cwnd
11:   sino
12:     si nak_rate > nak_threshold
13:       estado=CC
14:       cwnd = cwnd * b
15:       cc_cwnd = cc_cwnd * b
16:       continuar
17:   si se completó cwnd
18:     si estado==NORMAL y nak_rate < nak_threshold
19:       cwnd = cwnd + a
20:       cc_cwnd = cc_cwnd + a
21:   sino
22:     avanzar ventana cwnd
23: fin bucle

```

De esta manera, el emisor podrá ir incrementando la tasa de transferencia linealmente, cada vez que complete una ventana `cwnd` sin que la tasa de NAKs haya superado el umbral. Cuando la tasa de NAKs sea mayor al umbral configurable, deberá reducir la tasa de transferencia y luego entrará en la fase de control de congestión. Cuando haya pasado una ventana `cc_cwnd` completa donde la tasa de NAKs sea menor a la mitad del umbral definido, entonces podrá salir del estado de control de congestión y volver a trabajar con la ventana `cwnd`. Este proceso se repetirá sucesivamente y la tasa de transferencia debería converger al ancho de banda disponible en la red para la transmisión multicast.

3.2. Implementación

Los receptores solo implementarán la función normal de recepción con la notificación de los NAKs respectivos a los paquetes faltantes que hayan detectado como cualquier receptor normal PGM. Este diseño también tiene la ventaja de que no es necesario modificar la lógica interna del protocolo PGM o sus funciones de bajo nivel. El esquema de control de congestión se puede construir, por ejemplo, sobre la implementación OpenPGM del protocolo PGM, utilizando sus primitivas.

Se realizaron 2 modificaciones a la librería OpenPGM para poder satisfacer las necesidades de nuestra implementación: una función para poder consultar los contadores internos de la librería (mas precisamente el valor de los NAKs recibidos en el emisor para servir de dato de entrada del algoritmo de control de congestión) y un método para poder modificar la tasa máxima de transferencia que regula el throughput de envío del emisor dentro de una sesión determinada.

4. Escenario de pruebas

La investigación y desarrollo realizados surgen de la necesidad de mejorar el módulo de comunicaciones de un sistema de sucursales, que fué realizado con desarrollo propio y a medida, por la Gerencia de Sistemas de la Cooperativa Obrera Ltda. Este módulo es necesario para el envío de comandos de control a las cajas, para el envío de datos y para monitoreo del estado de cada una de ellas. En la Fig. 2 se muestra el esquema tipo y sobre el cual se realizaron las pruebas.

En particular el requerimiento de comunicaciones implementado, fue un mecanismo para poder enviar tablas de datos desde el SERVER a todas las cajas y además el envío de nuevas versiones de la aplicación (archivos de varios MBs), que por motivos de performance se decidió hacer el envío bajo multicast. Utilizar multicast en una red LAN nos daría la ventaja de realizar solo una transmisión total y, con un bajo número de reparaciones, ocupar muchos menos tiempo el ancho de banda de la red de la sucursal (la cual en muchos casos está limitada a equipamiento no switcheado y con bajas velocidades $\approx 10\text{ Mb/s}$); esta ventaja sería más evidente sobre la solución unicast a medida que creciera el número N de cajas y se reduciría considerablemente el tiempo total de transmisión.

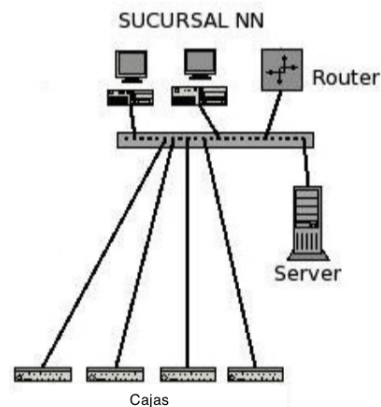


Figura 2.

4.1. Transmisión

En la etapa de preparación se realizan algunas tareas previas a la transmisión y que simplificarán los pasos posteriores. Una vez que se tiene el archivo final a transmitir, se calcula el tamaño total del mismo (`file_size`). La transmisión se realiza en bloques. Estos bloques permiten la reparación posterior, ya que los receptores pedirán a la fuente los bloques no recibidos/perdidos haciendo referencia por número de bloque. En esta etapa, se calcula el número total de bloques (`blocks`) en función de `file_size` y el tamaño de bloque (`MAX_DATA_SIZE`). Como último paso de esta etapa, se calcula un valor de hash (`md5sum`) de los contenidos del archivo. Este hash se utiliza para chequear la consistencia del archivo recibido en los receptores y decidir si se utilizará luego en capa de aplicación o deberá ser descartado por errores en algún bloque.

Se dividieron las pruebas en dos grupos de escenarios: aquellos 3 escenarios que tienen redes de 10 Mb/s con repetidores (hubs), escenarios A, B y C y los otros 3 que poseen el segmento de red implementado con switches de 100 Mb/s y tecnología full duplex, escenarios D, E y F. En el caso de 10 Mb/s se realizaron pruebas con tasas iniciales de transferencia de 200, 400, 600 y 800 KB/s. Para cada combinación de escenario y tasa inicial de transferencia se realizaron 5 pruebas de transferencia completas con cada uno de los 3 protocolos. Esto permitió analizar los resultados tomando promedio de

la tasa de transferencia y del tiempo de transmisión y luego se calculó varianza y desviación estándar de estos resultados usando la tasa de transferencia como variable aleatoria. Para los escenarios de 100 Mb/s se utilizó el mismo esquema de pruebas, pero usando como tasas de transferencia iniciales los siguientes valores: 2000, 4000 y 6000 KB/s.

Los protocolos de prueba fueron MDP desarrollado internamente y descrito brevemente en la Sección 3, PGM y nuestra propuesta PGMCOOP.

5. Resultados obtenidos

En principio, el análisis de los resultados del protocolo experimental desarrollado en el presente trabajo puede considerarse positivo. Como vemos en el gráfico que tomamos a modo de ejemplo de uno de los escenarios planteados, en la mayoría de las pruebas realizadas, PGMCOOP fue el protocolo que logró las mejores tasas de transferencia. Ver Fig. 3. El escenario es el C con 8 cajas.

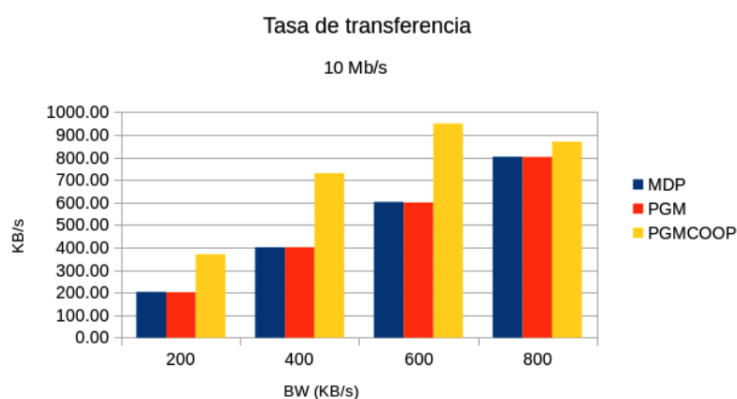


Figura 3.

Las pruebas se realizaron utilizando parámetros de crecimiento de la tasa de transferencia y tamaños de ventanas muy conservadores. Por eso, por ejemplo para tasas iniciales de 200 KB/s, solo se pudo lograr en promedio alrededor de 320 KB/s, pero podrían utilizarse parámetros que permitan un comportamiento mucho más agresivo del protocolo.

Los mejores rendimientos de PGMCOOP se obtienen utilizando como tasa de transferencia inicial el 50 % del ancho de banda disponible en el segmento de red.

Este comportamiento se produce porque la estrategia de aumentar la tasa de transferencia termina siendo contraproducente para estos casos donde la misma está cerca del límite de capacidad de los receptores.

Con la estrategia implementada, el protocolo pierde un intervalo de tiempo inicial hasta que se dispara el control de congestión, se disminuye la tasa de transferencia y el envío se estabiliza. Pero este proceso tiene un impacto en los resultados, suficiente para que un envío estable usando PGM a 800 KB/s termine dando mejores resultados.

En el escenario C, PGMCOOP claramente fue el protocolo de mejores resultados, en 200 y 400 KB/s casi duplicando las tasa de transferencias de MDP y PGM y en 600 KB/s logrando un 50 % más.

De estos resultados se puede concluir que resulta adecuada la estrategia de usar ventanas de transferencia para observar la tasa de NAKs que ingresan y subir la tasa de transferencia mientras que la misma no supere el umbral configurado.

En la mayoría de los casos de prueba de los escenarios de 100 Mb/s, también es el protocolo que logró los mejores resultados, pero no supera a los otros 2 protocolos tan claramente como en los casos anteriores. Al utilizar tasas de transferencia iniciales más exigentes, se llega más rápidamente a los límites de capacidad de los receptores. Ver Fig. 4. El escenario considerado es el F con 20 cajas.

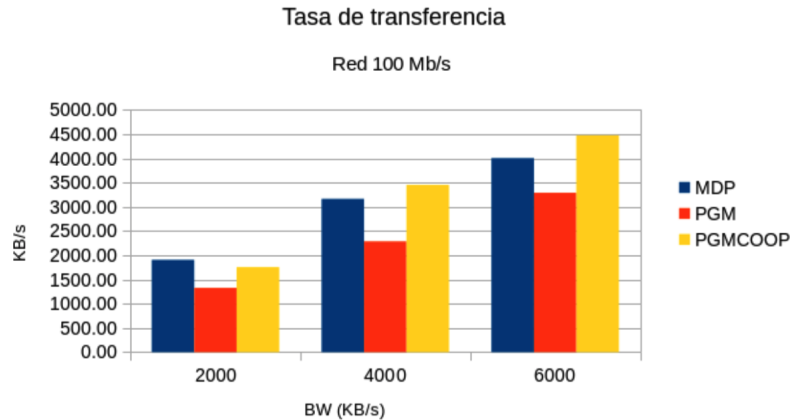


Figura 4.

Referencias

1. Banikazemi, Mohammad.:IP Multicasting: Concepts, Algorithms, and Protocols. Survey Paper, Ohio State University (1997)
2. Davies, Joseph.:Understanding IPv6, (2003)
3. Diot, Christophe; Dabbous, Walid; Crowcroft Jon.: Multipoint Communication: A Survey of Protocols, Functions and Mechanisms, (2004).
4. Comer, Douglas.: Internetworking with TCP/IP, (2000)
5. Adamson, B., Bormann, C., Handley, M., Macker, J.: NACK-Oriented Reliable Multicast (NORM) Transport Protocol, RFC 5740, IETF, (November 2009)
6. Whetten, B., Chiu, D., Paul, S., Kadansky, M., Taskale, G.:TRACK Architecture, A Scalable Real-Time Reliable Multicast Protocol, Internet Draft, Internet Engineering ask Force, (July 2000).
7. Luby, M., Watson, M., Vicisano, L.: Asynchronous Layered Coding (ALC) Protocol Instantiation, RFC 5775, (April 2010)
8. Floyd, S., Jacobson, V., Liu, C., McCanne, S., Zhang, L.: A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing, ACM SIGCOMM 95, (August 1995).
9. Shiroshita, T., Sano,T., Takahashi,O., Yamanouchi, N.:Reliable Multicast Transport Protocol, INTERNET-DRAFT, IETF, (September, 1997)
10. Paul,S.,Sabnani, K. K., Lin, J. C. -H.,Bhattacharyya, S.: Reliable multicast transport protocol (RMTP), IEEE Journal on Selected Areas in Communications (Volume:15 , Issue: 3), (1997-2002).
11. Rizzo, L.,Vicisano, L., Handley, M.: PGMCC single rate multicast congestion control: Protocol Specification, <https://tools.ietf.org/html/draft-ietf-rmt-bb-pgmcc-03.txt>, (2004).
12. Adamson B., Bormann C., Handley M., Macker, J.: NACK-Oriented Reliable Multicast (NORM) Protocol Building Blocks, (2001)
13. Speakman, T., Crowcroft, J., Gemmell, J., Farinacci, D. , Lin, S., Leshchiner, D., Luby, M., Montgomery, T. , Rizzo, L., Tweedly, A., Bhaskar, N., Edmonstone, R., Sumanasekera, R., Vicisano, L.: PGM Reliable Transport Protocol Specification, RFC 3208, (2001)

Ecuación de estado generalizada para redes de Petri no autónomas y con distintos tipos de arcos

Dr. Ing. Orlando Micolini¹, Geol. Marcelo Cebollada y Verdaguer¹, Ing. Maximiliano Eschoyez¹, Ing. Luis Orlando Ventre¹, Ing. Marcelo Ismael Schild¹.

Laboratorio de Arquitectura de Computadoras (LAC) FCEFyN
Universidad Nacional de Córdoba
{orlando.micolini, marcelo.cebollada.y.verdaguer, maximiliano.eschoyez, luis.ventre, marcelo.schild }@unc.edu.ar

Abstract. Este trabajo propone generalizar la ecuación de estado de las redes de Petri, para representar redes de Petri no autónomas con distintos tipos de brazos y semánticas temporales. Esta ecuación generalizada facilita la implementación por hardware de un IP-Core para la ejecución de la red. La solución expresada por esta ecuación de estado da origen a un algoritmo que preserva el modelo original, facilita su ejecución en paralelo y permite abordar problemas de mayor tamaño y complejidad. Además, se aborda un caso donde se pone de manifiesto las ventajas de incluir eventos, distintos tipos de brazos y semánticas temporales.

Abstract. This work proposes to generalize the state equation of a Petri Net, with the objective of represent different types of arcs and time semantics in non-autonomous Petri nets. This generalized equation facilitates the hardware implementation of an IP-Core to execute the network.

The solution that these extended state equation express raises an algorithm that preserves the original model, facilitates parallel execution and allows addressing problems bigger in size and complexity. In addition, exposes a case of application where highlights the advantages of including events, different types of arcs and timed semantics.

Palabras clave: Red de Petri no autónomas, Ecuación de estado, Procesador de Petri, IP-Core.

Keywords: non-autonomous Petri net, state equation, Petri Processor, IP-Core.

1 Introducción

La implementación de hilos en los sistemas informáticos permite explotar los recursos de las arquitecturas multicore [1]. Estos hilos cooperan y se ejecutan concurrentemente. Las aplicaciones diseñadas de esta manera tienen una complejidad intrínseca adicional con respecto a los programas secuenciales. Esta complejidad se manifiesta en el diseño, detección de errores, testing, validación y mantenimiento [2]. Por lo mencionado, es necesario introducir mecanismos de control, como los semáforos, que

penalizan los tiempos de ejecución. Por todo esto es conveniente generar una solución formal que garantice y facilite el desarrollo e implementación del sistema.

Investigaciones recientes mostraron que los modelos obtenidos con redes de Petri (RdP) facilitan la implementación de sistemas de manera directa, utilizando software, procesadores o IP-Core que ejecutan RdP [3, 4]. Procesadores de RdP no-autónomas (PP) se presentaron en [5, 6] como IP-Cores implementados en una FPGA Spartan-6.

La ejecución y simulación de sistemas complejos requieren modelado de RdP con mayor semántica. Para ampliar la capacidad semántica de las redes se incluyen eventos, guardas, distintos tipos de brazos (inhibidores, lectores, etc.) y semánticas temporales [7]. Esto amplía el dominio de los problemas, el tamaño y la complejidad.

Si bien en [8] se presenta una extensión de la ecuación de estado, los autores no consideran eventos, guardas y tiempo. Además, introducen variables en la matriz de incidencia lo que dificulta la implementación como circuito combinacional.

En el presente trabajo se generaliza la ecuación de estado de la RdP, usando el concepto de inhibición de disparos, para obtener las distintas ampliaciones semánticas e implementarlas en un sistema combinacional. De esta forma, se aprovechan más efectivamente los recursos de las FPGA, manteniendo todas las propiedades de las RdP temporales y no autónomas.

2 Fundamentos y Notación

2.1 Ecuación de estado

Las RdP se utilizan para representar gráficamente el comportamiento dinámico de un sistema. Por lo tanto, las características gráficas de las RdP pueden explotarse para inspeccionar la dinámica de un sistema. Este enfoque es adecuado para sistemas pequeños. Sin embargo, la metodología gráfica no es eficiente cuando los sistemas son grandes y complejos. En este artículo, desarrollamos una ecuación de estado para representar modelos de estos sistemas e implementarlos como IP-Core.

Es importante introducir la ecuación de estado de las RdP para extenderla con distintos tipos de brazos, eventos, guardas y tiempo. Con esta ecuación es posible obtener el siguiente estado del sistema. Esta es una manera más simple que la metodología gráfica para analizar la evolución de los sistemas.

La ecuación de estado de una RdP, con n plazas y m transiciones, con brazos con peso mayor o igual a uno y marca inicial M_0 , es:

$$M_{j+1} = M_j + I * \sigma. \quad (1)$$

Siendo: I matriz de Incidencia, con dimensión $n \times m$, σ el vector disparo, con dimensión $m \times 1$, M_0 el vector de marcado inicial, M_j el vector de marcado actual, M_{j+1} el vector de marcado del estado siguiente, todos con dimensión $n \times 1$.

Los elementos i_{ij} de la matriz I se obtienen,

$$i_{ij} = w(t_i, p_j) - w(p_j, t_i). \quad (2)$$

Donde w son los pesos de los arcos (valores enteros con signo):

- Los arcos de la plaza- i a la transición- j , son $-w(p_j, t_i)$.
- Los arcos de la transición- j a la plaza- i , son $w(t_i, p_j)$.

Cuando se dispara una transición sensibilizada, se puede calcular el siguiente estado usando (1).

La ecuación de estado representa matemáticamente el comportamiento dinámico del sistema. La matriz de incidencia caracteriza el comportamiento del sistema. Por lo tanto, la matriz de incidencia representa el sistema en sí.

2.2 Transición sensibilizada

Una transición está sensibilizada si todos los lugares de entrada a la transición tienen una marca igual o mayor al peso del arco que une cada plaza con la transición. Lo que se expresa como:

$$M(p_i) \geq w(p_j, t_i), \forall p_j \in I(t_i).$$

Las transiciones sensibilizadas se expresan como un vector binario E de dimensión $m \times 1$. Cada componente del vector indica con uno la transición sensibilizada y un cero la que no.

$$E = (\text{sign}(S^0), \text{sign}(S^1), \dots, \text{sign}(S^{n-1})).$$

Donde la relación $\text{sign}(S^i)$ de cada vector S^i es:

- S^i es igual a $M_j + I^i$, un vector con el estado que se obtendría de disparar la i -ésima transición una vez.
- I^0, I^1, \dots, I^{n-1} , son las columnas de la matriz I
- $\text{sign}(S^i)$, es binario siendo cero si alguna de las componentes de S^i es negativa, de otra forma es uno.

2.3 Disparo de una transición

Una transición se puede disparar solo si esta sensibilizada. Cuando se dispara se calcula el nuevo estado M_{j+1} y el nuevo vector de transiciones sensibilizadas E .

2.4 Interpretación de la matriz de incidencia

De la semántica de disparo de una RdP, se interpreta la matriz de incidencia como la evaluación conjuntiva entre las columnas (transiciones) y las restricciones que imponen las filas (plazas). Es decir, en una matriz de incidencia de dimensión $m \times n$ se evalúan n combinaciones de m variables lógicas, como se expresa en la siguiente ecuación:

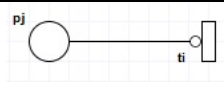
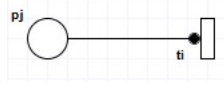
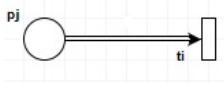
$$e_i = \left(\bigwedge_{h=0}^{m-1} M(p_h) \geq i_{hi} \right), \forall i = 0, \dots, n-1.$$

Donde i_{hi} son los elementos de la matriz I y $M(p_h)$ es la marca en la plaza h . El resultado son las componentes del vector E .

3 Extensión de la ecuación de estado

3.1 Arcos inhibidores, lectores y reset

Los arcos inhibidores, lectores y reset conectan una plaza con una transición, aumentando la capacidad de expresión de la RdP. Cabe destacar que los arcos inhibidores y lectores posibilitan modelar prioridades.

Arco	Representación	Consecuencia sobre la semántica
inhibidor		Si $h(p_j, t_i)$ es un arco inhibidor y hay token en p_j , entonces t_i no está sensibilizada.
lector		Si $l(p_j, t_i)$ es un arco lector y no hay token en p_j , entonces t_i no está sensibilizada. El disparo de t_i no consume los token de p_j .
reset		Si $r(p_j, t_i)$ es un arco reset y se dispara t_i se consumen todos los token de p_j . El disparo de t_i pone a cero la marca de la plaza p_j .

3.2 Guardas

Una guarda es una variable lógica asociada a una transición que aumenta la capacidad de expresión. Gráficamente, la guarda se representa con una etiqueta junto a la transición. Si $g(t_i)$ es una guarda, para disparar t_i se requiere que la transición esté sensibilizada y el valor de la guarda sea verdadero. Las guardas posibilitan comunicar la RdP con el medio haciéndolas no autónoma.

3.3 Eventos

Un evento se almacena en una cola que se asocia a una transición [3], aumentando la capacidad de expresión de la RdP. Gráficamente, la cola se representa como una etiqueta junto a la transición. La cola c_i , es un contador, que se incrementa cuando llega un evento y se decrementa cuando la transición asociada se dispara. Para que la transición esté sensibilizada se requiere que la cola tenga al menos un evento. Los eventos comunican la RdP con el medio para hacerla no autónoma.

3.4 Tiempo

Hay distintas semánticas temporales, como las propuestas en [9-11]. En este trabajo hemos tomado las RdP con tiempo introducidas en [12]. Estas imponen limitaciones de tiempo sobre los disparos, como un intervalo de tiempo asociado a cada transición.



Es decir, cada transición activa tiene asociada un cronómetro implícito que mide el tiempo transcurrido desde la última vez que se sensibilizó.

Una transición habilitada puede dispararse si el valor cronometrado, asociado a la transición, está dentro del intervalo de tiempo predeterminado. Por ejemplo, la etiqueta de intervalo $[\alpha_i, \beta_i]$ asociada a t_i , tiene como instante de inicio α_i y β_i como el instante de fin.

3.5 Política de selección de disparo

Se refiere a la elección de la próxima transición a disparar de entre todas las sensibilizadas. Para una RdP, incluidas las no autónomas o temporales, si esta elección es aleatoria el sistema resulta no determinístico. Para que el sistema sea determinístico hay diferentes soluciones, como la inclusión de: prioridades, probabilidades, arcos inhibidores, arcos lectores, etc.

Con el fin de evitar conflictos en las transiciones usamos la política de servidor único. Es decir, calculamos el nuevo estado considerando solo un disparo de una transición. Si se requieren más de uno, se realiza una secuencia de disparos.

4 Ecuación de estado extendida

4.1 Consideraciones para la generalización de la ecuación

En todas las ampliaciones de la semántica expuesta, destacamos que siempre se trata de determinar cuáles transiciones están sensibilizadas. Una vez determinada la transición a disparar, el disparo se realiza con la ecuación de estado original, excepto para el brazo reset que retira todos los token de la plaza.

Para representar matemáticamente la existencia de los brazos enumerados anteriormente se requiere de una matriz para indicar la conexión plaza transición. Estas matrices son similares a la matriz I . Los términos de las matrices son binarios, dado que los pesos de los arcos son uno.

Es decir, para que una transición esté sensibilizada se requiere:

- Si tiene brazo inhibidor que la plaza no tenga token.
- Si tiene brazo lector que la plaza tenga uno o más token
- Si tiene guarda que el valor de la guarda sea verdadero.
- Si tiene evento que tenga uno o más eventos en la cola asociada.
- Si tiene etiqueta con intervalo de tiempo, que el contador se encuentre en el intervalo.

4.2 Nueva ecuación de estado

De las consideraciones anteriores observamos que para disparar una transición se requiere la conjunción lógica entre todas las condiciones enumeradas en el punto anterior y el vector E de transiciones sensibilizadas.

Vector de transiciones des-sensibilizadas por arco inhibidor B , es un vector de valores binarios de dimensión $m \times 1$, que indica con un cero cuales transiciones están inhibidas por el arco y uno las que no, y se obtiene:

$$B = H * Q.$$

- Donde H es una matriz de dimensión $m \times n$ y Q un vector binario de dimensión $n \times 1$.
- H es la matriz de incidencia que relaciona las plazas que conectan las transiciones con un brazo inhibidor, los términos h_{ij} de la matriz son uno si hay brazo de la plaza p_i a la transición t_j y cero si no lo hay.
- Las componentes q_i del vector Q se obtienen de aplicar la relación

$$q_i = \text{cero}(M(p_i)).$$

- La relación $\text{cero}(M(p_i))$, es cero si la marca en la plaza p_i es distinta de cero, en otro caso un uno.

Vector de transiciones des-sensibilizadas por arco lector L , es un vector de valores binarios de dimensión $m \times 1$, que indica con un cero cuales transiciones están inhibidas por el arco y un uno las que no, y se obtiene:

$$L = R * W.$$

- Donde R es una matriz de dimensión $m \times n$ y W un vector binario de dimensión $n \times 1$.
- R es la matriz de incidencia que relaciona las plazas que conectan las transiciones con un brazo lector, los términos r_{ij} de la matriz son un uno si hay brazo de la plaza p_i a la transición t_j y un cero si no lo hay.
- Las componentes w_i del vector W se obtienen de aplicar la relación

$$w_i = \text{uno}(M(p_i)).$$

- La relación $\text{uno}(M(p_i))$, es uno si la marca en la plaza p_i es distinta de cero, en otro caso un cero.

Vector de transiciones des-sensibilizadas por guarda G , es un vector de valores binarios de dimensión $m \times 1$, que indica con un cero cuales transiciones están inhibidas por la guarda y un uno las que no, y se obtiene directamente de las guarda.

Vector de transiciones des-sensibilizadas por evento V , es un vector de valores binarios de dimensión $m \times 1$, que indica con un cero cuales transiciones están inhibidas porque no hay evento solicitando el disparo de la transición. Las componentes v_i del vector V se obtienen de aplicar la relación:

$$v_i = \text{uno}(\text{bufferDeEventos}_i).$$

La relación *uno(bufferDeEventos_i)*, es uno si hay uno o más eventos en el buffer asociado con la transición t_i , en otro caso cero.

Vector de transiciones des-sensibilizadas por tiempo Z, es un vector de valores binarios de dimensión $m \times 1$, que indica con un cero cuales transiciones están inhibidas porque no se ha alcanzado o se ha superado el intervalo de tiempo transcurrido desde de que la transición fue sensibilizada.

$$Z = Tim(q(E, B, L, G, clk), intervalos).$$

- E es el vector de sensibilizado, B es el vector de no-sensibilizado por arco inhibidor, L es el vector de no-sensibilizado por arco lector y G es el vector de no-sensibilizado por guarda.
- La relación $Tim(q, intervalos)$ es un vector binario de $m \times 1$, el valor de la componente i es uno si q_i , que es un contador, tiene valor en el intervalo indicado por la componente $intervalos_i = [\alpha_i, \beta_i]$ de otra forma es cero.
- Para que el contador q_i arranque la ecuación e_i and b_i and l_i and g_i debe ser uno de otra forma el contados se pone a cero, este contador se incrementa en una unidad de tiempo por cada pulso de reloj, de la base de tiempo (clk).
- La matriz $intervalos$ es de dimensión $m \times 2$ y contiene el límite inferior el límite superior de la ventana de tiempo para el disparo.

Vector de transiciones reset A, es un vector de valores enteros de dimensión $m \times 1$, que tiene el valor de la marca de la plaza que se quiere poner a cero, mientras que las otras componentes son uno.

La matriz Re expresar matemáticamente los arcos de reset. El arco que une la plaza p_i con la transición t_j se indica con un uno en la componente re_{ij} de la matriz de otro modo es cero.

$$A = Marca(Re * M_j).$$

La multiplicación $Re * M_j$ es un vector con cero sin no hay brazo reset y el valor de la marca si hay brazo reset.

La relación $Marca$ es uno si la componente es cero en otro caso el valor de la componente.

El vector A se multiplica elemento a elemento con el vector de disparo σ , operación que indicamos con $\#$. El vector σ tiene un uno en la transición que se quiere disparar por lo que obtenemos la cantidad de disparos necesarios para sacar todas las marcas de la plaza a resetear en otro caso solo un disparo.

4.3 Vector de sensibilizado extendido y ecuación de estado

Este vector de sensibilizado extendido Ex se obtiene de la conjunción lógica de todos los vectores anteriores.

$$Ex = E \text{ and } B \text{ and } L \text{ and } V \text{ and } G \text{ and } Z.$$

Para introducir el brazo reset hay que multiplicar elemento a elemento (#) a el vector que resulte de la conjunción por A . Por lo que la ecuación de estado extendida resulta:

$$M_{j+1} = M_j + I * ((\sigma \text{ and } Ex)\#A).$$

La importancia de esta ecuación de estado es que es de simple implementación como circuito combinacional en una FPGA.

5 Caso de Aplicación

5.1 Sistema de producción modelado con RdP extendida

La Figura 1 muestra el uso de arcos inhibidores y lectores para modelar prioridad y secuencias en un sistema de fabricación. Consideremos el proceso de fabricación representado en la Figura 1(a) donde, una máquina procesa tres tipos de trabajos. El trabajo J1 tiene una prioridad más alta que J2 y J3, y debe haber esperando al menos un J2 para procesar J3.

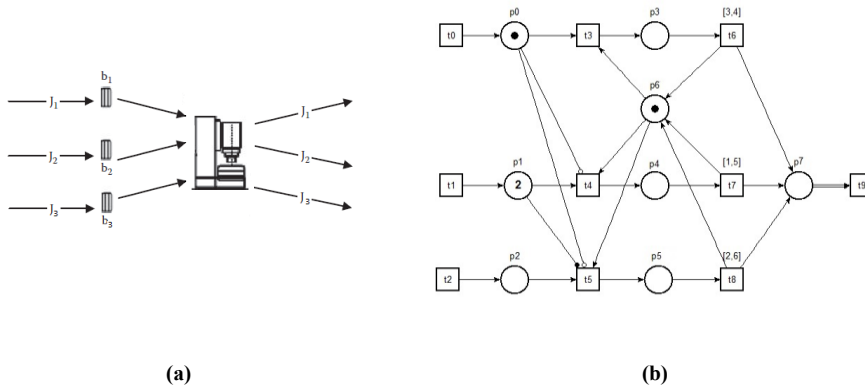


Figura 1: (a) Sistema de producción y (b) la RdP que lo modela

La RdP de la Figura 1(b) muestra el modelo del sistema. En el modelo t_0 , t_1 y t_2 son las llegadas de solicitudes de trabajos J1, J2 y J3, los token en p_0 , p_1 y p_2 representan estas solicitudes. Un token en alguna de las plazas p_3 , p_4 o p_5 representa que se está realizando el trabajo. Las transiciones t_6 , t_7 y t_8 tienen asociada la etiqueta de tiempo, que es el requerido por cada trabajo y su disparo corresponde a la finalización de este. La plaza p_6 representa la máquina y la plaza p_7 totaliza la cantidad de trabajos realizados. El disparo de t_9 , dado que hay un arco reset entre p_7 y t_9 , pone a cero la plaza p_7 .

A continuación expresamos las matrices y vectores del modelo de la Figura 1.

Matriz I									Matriz de brazos inhibidores B								
-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	-1	-1	-1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	-1	-1	-1	0	0	0	0	0	0	0	0	0

Matriz de brazos lectores L									Matriz de brazos Reset R								
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Matriz intervalos temporales	Vector inhibidos por tiempo	Vector de Marcado	Vector de transiciones sensibilizadas	Vector de transiciones inhibidas por arco inhibidor	Vector de transiciones inhibidas por arco lector	Vector de transiciones sensibilizadas extendido
<i>intervalos</i>	<i>Z</i>	<i>M_j</i>	<i>E</i>	<i>B</i>	<i>L</i>	<i>Ex</i>
0	0	1	1	1	1	1
0	0	2	1	1	1	1
0	0	0	1	1	1	1
0	0	0	1	1	1	1
0	0	0	1	0	1	0
0	0	0	0	0	1	0
3	4	1	0	1	1	0
1	5	0	0	1	1	0
2	6	0	0	1	1	0
0	0	0	0	1	1	0

La implementación de esta ecuación requiere de 6 matrices. Para obtener matrices de menor dimensión, con el consiguiente ahorro de recursos, es posible aplicar técnicas de división de redes como las presentadas en [13].

6 Conclusiones

Este trabajo presentó la ecuación de estado y el vector de transición sensibilizada de RdP generalizada para múltiples tipos de arcos, eventos, guardas y tiempo. Esta ecuación y vector son de lógica conjuntiva lo que facilita la implementación con circuitos combinacionales en un IP-Core. Se ha tenido la precaución de no introducir variables en las matrices, dado que esto dificulta la implementación y validación de la ecuación de estado en el circuito combinacional.

Esta nueva ecuación de estado facilitó el desarrollo de software - hardware, que implementa y ejecuta RdP temporales y no-autónomas. Este software - hardware se

utilizó para obtener los estados del sistema y la ejecución de la lógica concurrente y paralela del programa.

La ecuación de estado generalizada se obtuvo con la conjunción de vectores. Para esto se ha obtenido un vector para cada tipo de arco, etiquetas con tiempo, eventos y guardas. Esto consiguió la modularidad del software – hardware según sea la red que se desea implementar.

Como trabajo futuro, se está diseñando un nuevo PP con arquitectura pipeline basado en esta ecuación, lo que permitirá ejecutar sistemas de mayor complejidad.

Referencias

1. David R. Martinez, R.A.B., M. Michael Vai, *High Performance Embedded Computing Handbook A Systems Perspective* 2008, Massachusetts Institute of Technology, Lincoln Laboratory, Lexington, Massachusetts, U.S.A.: CRC Press.
2. Domeika, M., *Software Development for Embedded Multi-core Systems* 2008, 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA Linacre House, Jordan Hill, Oxford OX2 8DP, UK.
3. Micolini, O., *ARQUITECTURA ASIMÉTRICA MULTI CORE CON PROCESADOR DE PETRI*, in *Informatica* 2015, UNLaP: La Plata, Argentina.
4. Moutinho, F. and L. Gomes, *Distributed Embedded Controller Development with Petri Nets: Application to Globally-Asynchronous Locally-Synchronous Systems*. Vol. 150. 2015: Springer.
5. Micolini, O., J. Nonino, and C.R. Pisetta. *IP Core Para Redes de Petri con Tiempo*. in *CASIC 2013*. 2013.
6. M. Pereyra, N.G., M. Alasia and O. Micolini, *Heterogeneous Multi-Core System, synchronized by a Petri Processor on FPGA*. IEEE LATIN AMERICA TRANSACTIONS, 2013. **11**: p. 218-223.
7. Diaz, M., *Petri Nets Fundamental Models, Verification and Applications* 2009, NJ USA: John Wiley & Sons, Inc.
8. Baškocagić, C. and S. Kurtulan, *Generalized state equation for Petri nets*. WSEAS TRANSACTIONS on SYSTEMS, 2011. **10**(9): p. 295-305.
9. Sifakis, J., *Performance evaluation of systems using nets*. Springer-Verlag Berlin Heidelberg, 1979. **84**: p. 307-319.
10. Roussopoulos, J.E.C.y.N., *Timing requirements for time-driven systems using augmented Petri nets*. IEEE transactions on Software Engineering, 1983. **9**(5): p. 603-616.
11. Jensen, K. and L.M. Kristensen, *Coloured Petri Nets Modelling and Validation of Concurrent Systems*. Springer 2009.
12. Merlin, P.M., *A Study of the Recoverability of Computing Systems*, 1974, University Microfilms, : University of California.
13. Micolini, O., M. Cebollada, and L.O. Ventre. *Localidad estructural, criterio de división para la ejecución de redes de Petri no autónomas en IP-Core*. in *XXI Congreso Argentino de Ciencias de la Computación (Junin, 2015)*. 2015.

Diseño de un componente de simulación para comunicaciones en redes de sensores

Pi Puig Martín¹, Medina Santiago¹, Batista Ary², Encinas Diego¹, Romero Fernando¹,
De Giusti Armando^{1,3}, Tinetti Fernando G.^{1,4}

¹Instituto de Investigación en Informática LIDI (III-LIDI), Facultad de Informática, Universidad Nacional de La Plata, 50 y 120 2do piso, La Plata, Argentina.

²Universidad Nacional de Quilmes, Roque Sáenz Peña 352, Bernal, Argentina.

³CONICET – Consejo Nacional de Investigaciones Científicas y Técnicas

⁴CIC – Comisión de Investigaciones de la Pcia. de Buenos Aires

{mpipuig, smedina}@lidi.info.unlp.edu.ar, ary.batista@unq.edu.ar, {dencinas, fromero, degiusti, fernando}@lidi.info.unlp.edu.ar

Abstract. Los Sistemas Distribuidos de Tiempo Real deben ejecutar algoritmos en plazos de tiempo adecuados a los requerimientos de la implementación. Durante la etapa de verificación y validación del hardware, pueden encontrarse medidas que impliquen la realización de cambios o modificaciones en los mismos. Una manera de disminuir la complejidad y la probabilidad de errores en la generación de hardware es desarrollar una simulación específica de éstos. Se propone el diseño, implementación y validación de un modelo de simulación de un sistema de comunicaciones Controller Area Network, utilizando el entorno de simulación Proteus. La finalidad del modelo es predecir el comportamiento de la transmisión en diferentes escenarios.

Keywords: Modelado, Simulación, Comunicaciones en Sistemas de Tiempo Real, Controller Area Network.

1 Introducción

La infraestructura de comunicación en un sistema de tiempo real distribuido debe cumplir con requerimientos no funcionales [15] muy diferentes a los demandados por otros sistemas. Estos requerimientos son: exactitud, fiabilidad, flexibilidad y estructural.

En el requerimiento de exactitud se puede notar la necesidad de conseguir una latencia corta y un jitter mínimo, siendo esta la principal diferencia entre los sistemas de comunicaciones de tiempo real y los otros. La latencia tiene una duración que inicia con la lectura de los sensores en cada uno de los nodos y finaliza con la salida del actuador correspondiente, teniendo en cuenta el procesamiento requerido. El jitter es la diferencia entre las latencias del peor y mejor caso, y un valor alto de este parámetro puede producir efectos negativos para calcular el instante de aceptación de un mensaje.

La fiabilidad engloba características que necesitan poseer los sistemas de comunicaciones para aportar seguridad a la comunicación (esto se consigue por medio de redundancia). También, un protocolo de comunicaciones de tiempo real debe ser lo suficientemente flexible para soportar cambios sin la necesidad de requerir modificaciones en el software. Los requerimientos estructurales de la red están determinados no solo por consideraciones técnicas (requerimientos funcionales) sino también económicas.

Uno de los primeros pasos en la fase de diseño de estos sistemas es efectuar un análisis de funcionamiento de los mismos. En la actualidad esto se consigue por medio de técnicas de simulación; de esta manera se permite estudiar, entre otros aspectos:

- 1) la interacción entre los distintos componentes
- 2) protocolos de comunicaciones
- 3) modos de operación
- 4) potencia consumida de los distintos subsistemas.

Debe quedar claro que el propósito de la simulación funcional no es representar el funcionamiento exacto de los componentes ya que se perdería la ventaja de abstracción de las características físicas [1].

En este trabajo se propone un modelo de simulación de un dispositivo de comunicaciones CAN [2], utilizando el entorno de simulación Proteus [3] que actualmente no ofrece este protocolo. Luego la validación del modelo se realizará por medio de un sistema físico con el cual se contrastarán los datos obtenidos con las pruebas en el simulador.

El trabajo está organizado de la siguiente forma: en la segunda sección se describirán algunas características del protocolo de comunicaciones CAN. En la tercera sección se explica la metodología utilizada para desarrollar el modelo del sistema. En la cuarta sección se muestran el desarrollo de la simulación. En la quinta sección se explica la validación y los resultados obtenidos. Finalmente, en la sexta sección las conclusiones y trabajos futuros.

2 Protocolo CAN

Para este trabajo se ha elegido el protocolo CAN el cual ha sido formalmente definido por el estándar ISO 11898 [4]. Este protocolo fue diseñado por la compañía Robert Bosch GmbH en Alemania y el propósito inicial era crear una red más rápida y resistente a interferencias dentro del ámbito automotriz. El estándar ofrece una comunicación determinística, con prioridades y detección de errores. La topología del bus consiste generalmente en un medio de transmisión, formado por un par de cables, y estaciones o nodos. Aunque la especificación del medio de transmisión del cable no está indicada en la norma original puede ser adaptada dependiendo el área de aplicación. La codificación utilizada es de Non Return to Zero (NRZ) con stuffing sobre un canal de señales diferenciales (balanceado) lo que asegura mensajes compactos con un mínimo número de transiciones y una alta resistencia a las

perturbaciones externas. En la Figura 1 puede verse el conexionado típico de los nodos a la red y los elementos que la componen.

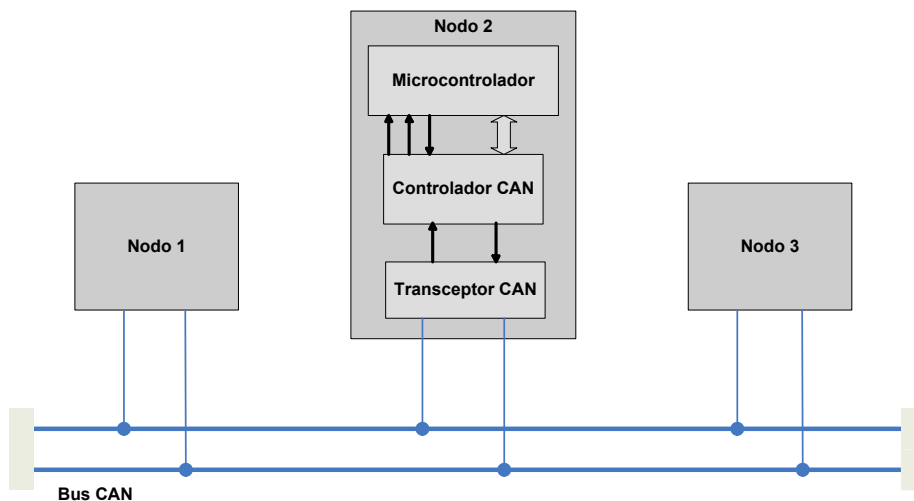


Fig. 1.

CAN es conocido como un protocolo de comunicaciones serie de alta confiabilidad, robustez y performance; además es apropiado, mediante un desarrollo de capa de aplicación, para el control de sistemas distribuidos de tiempo real. Sus principales características son:

- Priorización de mensajes.
- Sistema Multi-maestro.
- Configuración flexible.
- Velocidad de transmisión media (hasta 1 Mbit/s).
- Señalización y detección de fallas.

El mecanismo de transacción usado por el protocolo CAN para transferir datos es similar al modelo productor consumidor en donde cada nodo envía mensajes sin requerir confirmación y de la forma multicast o broadcast. El tipo de acceso al bus CAN es multi-maestro; por este motivo, el derecho a acceder al bus no es asignado por un nodo central pero de todas formas todos los nodos pueden intentar acceder al mismo tiempo. El arbitraje se resuelve por un procedimiento que hace uso de los identificadores que se encuentran al mismo tiempo en el bus.

La transferencia de mensajes se manifiesta y controla por cuatro tipos de trama diferentes: tramas de datos, tramas remotas, tramas de error y tramas de sobrecarga. En la Figura 2 se puede observar los campos de la trama de datos.

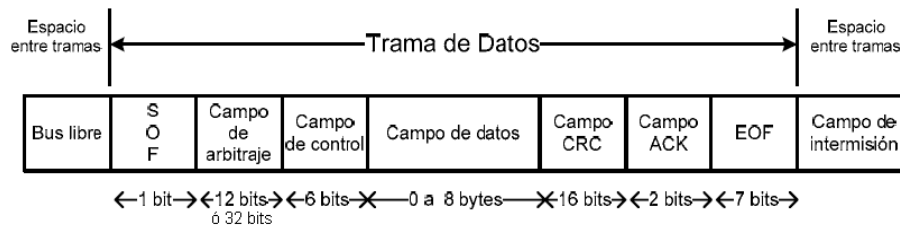


Fig. 2.

3 Metodología

3.1 Emisión de mensajes

El modelo de simulación se realiza considerando un análisis de planificación de transmisión de mensajes para redes CAN pero imponiendo restricciones propias del desarrollo de un simulador.

El modelo de planificación CAN utilizado es el propuesto en [5] y en sus trabajos previos. Un concepto de planificación importante es que cada mensaje es encolado por una tarea, proceso o interrupción software que se ejecuta en el procesador host. Esta tarea es invocada por un evento o consulta de estado con una cantidad de tiempo limitada para encolar el mensaje que esté listo para transmitir. Este tiempo varía entre 0 y J_m , donde J_m es denominado el jitter de encolamiento del mensaje.

Generalmente los modelos de planificación de prioridad fija como es el caso del protocolo CAN, presentan jitters a pesar de poseer sólo tareas periódicas [6]. Existen varios trabajos que realizan un análisis de planificabilidad del protocolo [7] [8] [9] e introducen al jitter como un parámetro necesario para efectuar la planificación de mensajes que se envían periódicamente.

La presencia de un jitter aleatorio provoca un retardo del tiempo de transmisión, en [10] se realiza un análisis estocástico de los tiempos de respuesta y se propone una función de distribución uniforme para el jitter. Pero si se considera que el jitter de un componente está formado por la contribución de distintos subcomponentes, que poseen a su vez otros jitters, puede apelarse al teorema del límite central. El teorema del límite central establece que la función distribución de probabilidad de la suma de un número suficientemente grande de variables aleatorias, con distintas distribuciones, puede aproximarse a una normal o gaussiana [11]. Esta es una de las razones, para proponer como función de distribución de probabilidad del jitter a la normal o gaussiana.

3.2 Ventanas de lectura y escritura

El envío y recepción de mensajes está implementado por medio del concepto de ventanas de escritura y lectura. Las ventanas están diseñadas para tener un intervalo de tiempo fijo y representar las restricciones que poseen los mensajes cuando llegan

en un momento o periodo en el que no pueden transmitir a pesar de poseer una alta prioridad. Por lo tanto, cuando un mensaje llega durante un intervalo de tiempo de ventana de lectura debe esperar hasta el inicio de ventana de escritura para competir con otros posibles mensajes en el momento de arbitraje.

3.3 Arbitraje

Cada nodo escribe el bit correspondiente a su identificador durante el periodo de ventana de escritura y luego, en la ventana de lectura, se verifica que el bit presente en el bus corresponde al escrito. Si este bit difiere, significa que existe un mensaje emitido por un nodo con mayor prioridad. Entonces los nodos con menor prioridad se dedican a recibir el mensaje que se está transmitiendo en el canal.

Es decir, un nodo logra atravesar el proceso de arbitraje si puede escribir por completo su identificador. De esta forma obtiene el control del bus y procede a emitir su mensaje.

3.4 Reloj global

La sincronización entre los nodos de la red está dada por una señal provista por el entorno de simulación. Esta señal se emite a intervalos regulares sobre un canal de sincronización y determina el inicio y fin de las ventanas de escritura y lectura.

Por lo tanto, se asume un reloj global suponiendo que los métodos Hard Synchronization y Resynchronization [12] son suficientes para sincronizar las señales de reloj de cada nodo en el sistema real.

3.5 Herramientas utilizadas

Para el modelado y la implementación de la simulación, se utilizó un conjunto de herramientas, las cuales se listan a continuación.

- Proteus v7.7
- CCS Compiler v5.0

Luego, dentro de la suite de simulación Proteus, se decidió utilizar microcontroladores del tipo PIC16F877A como componentes básicos de implementación de los nodos y controladores CAN.

Para la implementación del sistema físico, el microcontrolador seleccionado fue el AT89S52, el controlador CAN elegido fue el SJ1000 y el transceptor CAN el componente PCA82C250.

4 Desarrollo

4.1 Implementación

En el sistema básicamente existen dos módulos principales: el Nodo y el Bus. Como sucede físicamente, el Nodo modelado está compuesto por un microcontrolador y un controlador CAN. En este modelo de simulación no se ha tenido en cuenta al transceptor CAN ya que este componente agrega un retardo fijo del orden de los nanosegundos, el cual es un valor pequeño en comparación al tiempo necesario para la transmisión de un bit (1 μ seg).

En la Figura 3 se muestra el diagrama de comportamiento de la implementación para el envío de un paquete de datos CAN.

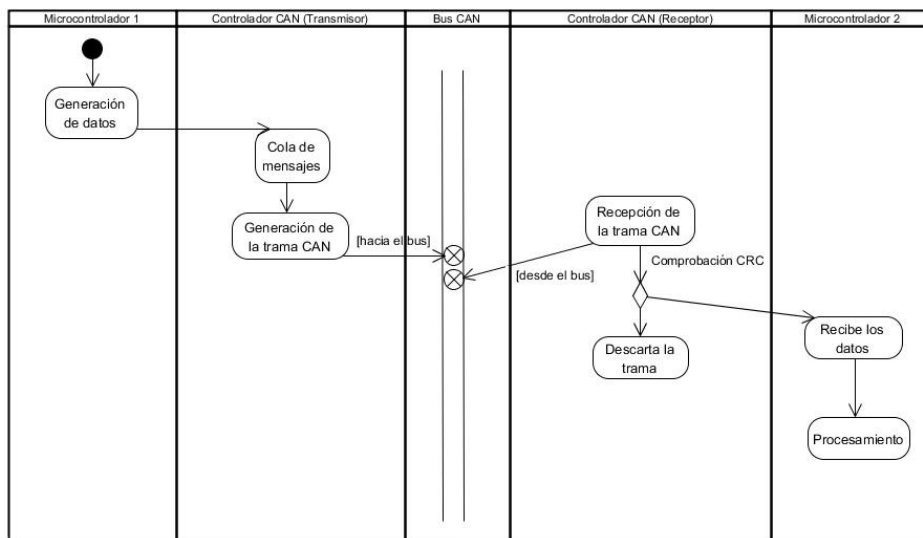


Fig. 3.

4.2 Modelo de programación

En la Figura 4 se muestra un diagrama de actividades de los componentes principales del sistema de simulación.

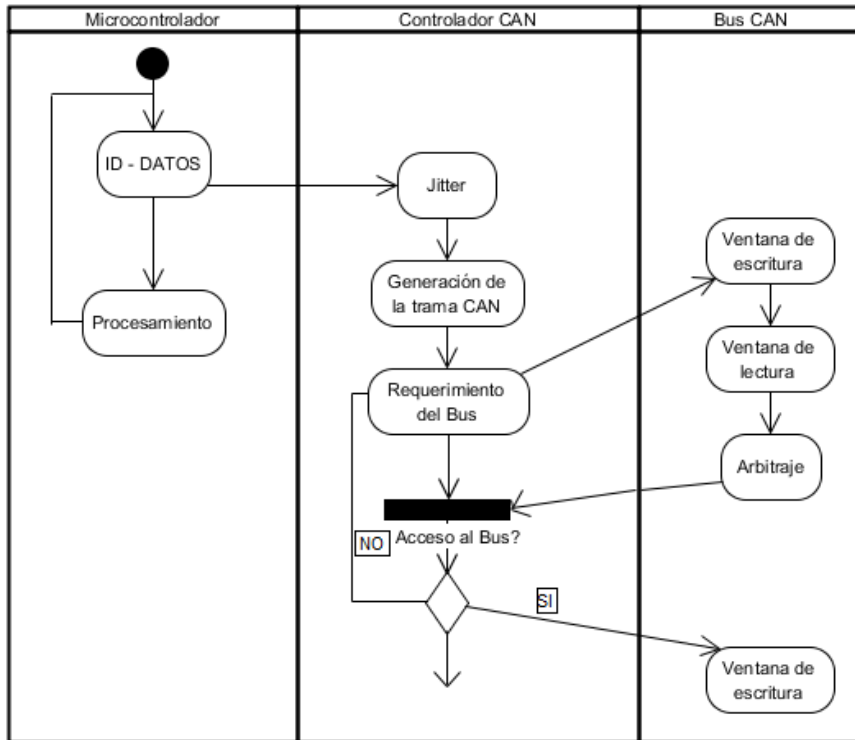


Fig. 4.

El microcontrolador cumple las funciones de sensor del sistema y de host del controlador CAN. En este componente se generan el dato e identificador del mensaje. Estos parámetros se transmiten al controlador CAN por medio del protocolo RS-232.

El controlador CAN junto con el bus, es uno de los principales componentes del sistema de simulación. Por medio de diferentes funciones se completan los campos de la trama de datos, se introduce el dato a transmitir y se agrega el jitter correspondiente al buffer de transmisión. Como se ha mencionado en la sección 3.1 se propone representar al jitter con una función de distribución normal. Se implementó una función que aplica la transformada de Box-Muller [13] para la generación de números aleatorios con la función de densidad de probabilidad normal o gaussiana.

El bus de datos (Bus CAN) se realizó utilizando compuertas lógicas del tipo OR para permitir a cada nodo incrementar el valor de la ventana actual. Y una compuerta del tipo AND que reinicia el bus cada vez que el reloj global vuelve a cero el canal de sincronización.

5 Resultados

5.1 Métricas obtenidas con el sistema físico

Seleccionando una periodicidad de envío de 4 mseg, la tasa de transmisión programada fue de 3,96 mseg. Esto es consecuencia de la arquitectura del microcontrolador (tamaño de registros), el proceso de transmisión del controlador CAN y la frecuencia de los cristales en cada nodo.

En general, el tráfico en redes de datos tradicionales se analiza a partir del throughput, latencia y jitter [14] de la red. Aunque en una red para sistemas de tiempo real estricto estos parámetros son necesarios para cuantificar la calidad de servicio (QoS), por las características específicas de las redes en el protocolo CAN medir el throughput es poco significativo. De hecho el throughput es fijo y de 1 Mbps, ya que es la máxima velocidad de transferencia que admite el protocolo. Es por esto que el parámetro que se evalúa es el Inter Arrivals Time (IAT) de mensajes.

La cantidad de mensajes enviados fue de 37339 aproximadamente en 150 seg (tasa de 3,96 mseg). En la Figura 5 se puede ver las métricas obtenidas.

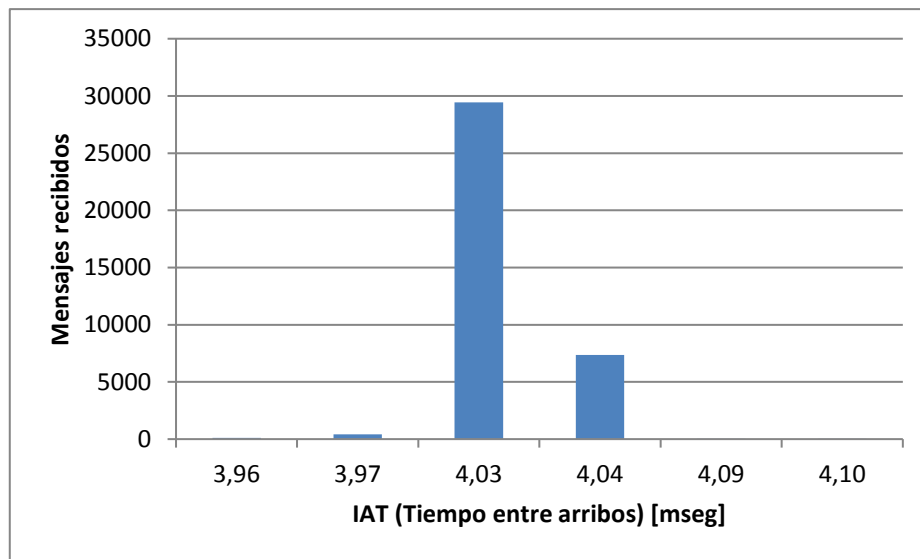


Fig. 5.

Como puede observarse en la Figura 5, el mayor porcentaje de mensajes se recibe a un IAT promedio de 4,03 mseg y el resto de los mensajes no se desvían más de 70 μ seg de ese valor.

5.2 Métricas obtenidas con el modelo de simulación

El modelo de simulación agrega un overhead en la ejecución comprendido por: tiempos de procesamiento para implementar el CRC del protocolo CAN, arbitraje (por medio de ventanas de lectura y escritura) y el jitter de distribución gaussiana.

El escenario que se simuló fue realizado en base al sistema físico, un nodo que transmite a una tasa periódica de 3,96 mseg un mensaje de 8 bytes de datos a un nodo receptor. Nuevamente la prueba consistió en enviar 37339 mensajes. En la Figura 6 pueden verse las métricas obtenidas.

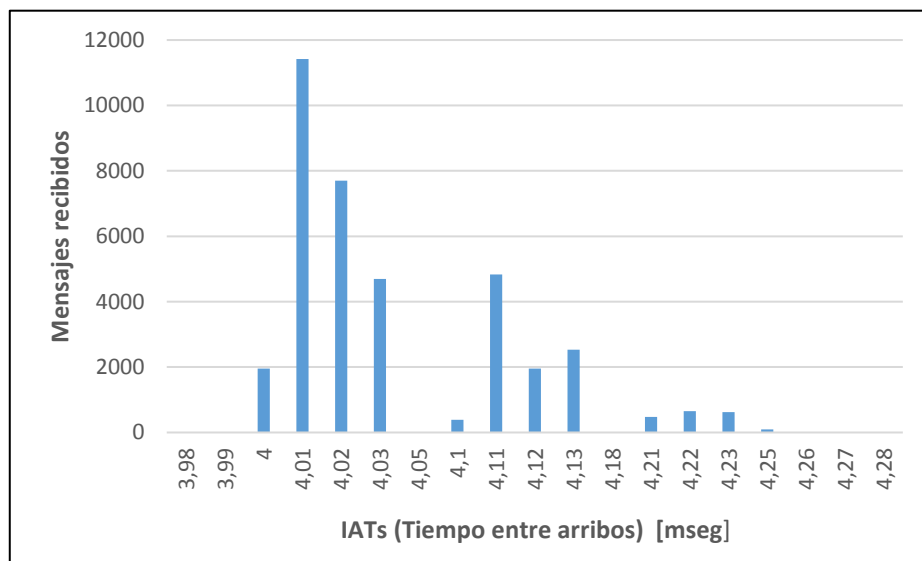


Fig. 6.

En la Figura 6 puede verse como principal diferencia con el sistema físico, que el valor del IAT promedio es de 4,05 mseg con una desviación máxima de 230 μ seg. Pero por otra parte aumenta el número de IATs, permitiendo obtener una mayor resolución. Aun así puede observarse un comportamiento similar en los dos modelos.

6 Conclusiones

De la observación de los resultados del modelo de simulación se puede decir que cumple con la especificación del protocolo CAN y de esta forma es validado. Además, de la comparación de los resultados de la ejecución de simulación con los obtenidos con el sistema físico se puede afirmar que con los mismos datos de entrada los dos sistemas producen salidas similares.

Desarrollar la implementación necesaria para que el modelo de simulación soporte otros tipos de tramas CAN, mejorar el escalamiento del bus y analizar otras

distribuciones de probabilidades para el jitter, serían los próximos pasos para proseguir con el presente trabajo.

Bibliografía

- [1] EICKHOFF J. (2009). *Simulating Spacecraft Systems*. Springer.
- [2] Robert Bosch GmbH. *CAN Specification 2.0*. 1991
- [3] Proteus. <https://www.labcenter.com>. 2016
- [4] ISO 11898: Road Vehicles – Interchange of digital information – Controller Area Network (CAN) for high speed communication. 1993.
- [5] R. Davis, «Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised.,» *Real-Time Systems*. Springer, vol. 35, n° 3, pp. 239-272, 2007.
- [6] A. Burns, *Real-time systems and programming languages*, Addison Wesley, 2009.
- [7] R. Davis, «Controller Area Network (CAN) Schedulability analysis with FIFO queues,» *23rd Euromicro conference on Real-Time Systems*, pp. 45-56, 2011.
- [8] P. Yomsi, «Controller Area Network (CAN): Response time analysis with offsets,» *9th IEEE International workshop on Factory Communication Systems*, pp. 43-52, 2012.
- [9] N. Navet, «Controller Area Network (CAN) Schedulability analysis for messages with arbitrary deadlines in FIFO and work-conserving queues,» *9th IEEE International workshop on Factory Communication Systems*, pp. 33-42, 2012.
- [10] M. Di Natale, *Understanding and Using the Controller Area Network Communication Protocol. Theory and Practice*, Springer, 2012.
- [11] J. Devore, *Probabilidad y estadística para ingeniería y ciencias*. Sexta Edición, Thomson Learning, 2005.
- [12] Philips semiconductors, *Application note. Determination of Bit Timing Parameters for the CAN Controller SJA 1000*. AN97046, 1997.
- [13] Box, G. E. P.; Muller, Mervin E. *A Note on the Generation of Random Normal Deviates*. *Ann. Math. Statist.* 29. 1958
- [14] J. Kurose, *Redes de Computadores*. 2º Edición, Pearson-Addison Wesley, 2004
- [15] Hermann Kopetz. *Real-Time Systems. Design Principles for Distributed Embedded Applications*. Second Edition. Springer. 2011. ISSN 1867-321X e-ISSN 1867-3228 ISBN 978-1-4419-8236-0 e-ISBN 978-1-4419-8237-7

Emulación de elementos de networking interactuando con máquinas virtuales

Carlos Binker¹, Alejandro Pérez¹, Guillermo Buranits¹, Eliseo Zurdo¹

¹Universidad Nacional de La Matanza, Florencio Varela 1903 (B1754JEC) -- San Justo,
Buenos Aires, Argentina

{cbinker, aperez, gburanits, }@unlam.edu.ar;
ezurdo@alumno.unlam.edu.ar

Abstract. En este trabajo se pretende mostrar la emulación de elementos de networking tales como switches y routers interactuando con máquinas virtuales que emplean sistemas operativos diversos como ser Windows, Mac OS X, Linux en diferentes distribuciones, etc. A tal efecto se emplea una plataforma de software libre denominada GNS3 (Graphical Simulator Network 3). Después de hacer un análisis más pormenorizado de dicha plataforma con sus programas asociados se mostrará un ejemplo de laboratorio en donde se simula una red de una empresa que se conecta con otras dos sucursales mediante enlaces punto a punto, como podría ser el caso de una red de un banco (la casa central con dos sucursales).

Keywords: GNS3, VMware, IOS images, Appliances, Virtual Machine

1 Introducción

En los últimos años el enorme crecimiento registrado en el ámbito del networking, sobre todo teniendo en cuenta el avance vertiginoso de forma exponencial de Internet, ha traído aparejado un importante incremento de las estructuras de hardware. Por esta razón y dada la necesidad de reducir físicamente estas enormes estructuras es que ha nacido un nuevo paradigma denominado *Virtualización* [1]. Clásicamente si por ejemplo se requería montar una granja de servidores, era necesario contar con muchas cajas físicas en donde cada caja constituía un servidor brindando un determinado servicio. Con la virtualización en cambio en una sola caja física se podrían albergar múltiples servidores *lógicos* con la consiguiente disminución de las estructuras de hardware. Pero claro, esto no resulta gratuito, se requieren procesadores con alta potencia de cómputo (varios núcleos para procesamiento paralelo) y mucha capacidad de memoria RAM, como así también gran capacidad de almacenamiento de disco rígido. De esta manera nace el concepto de *Virtual Machine* [2], en castellano máquina virtual, es decir de una estructura lógica que emula los componentes básicos de hardware (disco duro, cpu, ram, placas de red, lector de dvd, audio, video, etc.) y que

corre un determinado sistema operativo. La máquina virtual es en esencia software que utiliza los recursos de hardware de una máquina principal denominada *host*. Cuando se instala una máquina virtual en una máquina *host*, para el usuario ésta se ve como si fuera una máquina física más con las mismas características de una máquina *host*, en donde los recursos a asignar a la máquina virtual son totalmente configurables. La máquina virtual instalada en una máquina *host* se denomina comúnmente *guest*. Por lo tanto, la máquina virtual utiliza un sistema operativo, denominado también *guest* que puede ser Windows, Mac OS X o cualquier distribución de Linux. Existen varias plataformas que permiten crear máquinas virtuales, entre las más populares tenemos *VirtualBox de Oracle* [3], *VMware Workstation* [4], *qemu* [5] (que es nativa en los sistemas Linux), etc. Pero no sólo las computadoras con sus correspondientes sistemas operativos pueden ser virtualizadas, sino que el concepto de virtualización se extiende también a los elementos de *internetworking*, como ser los switches, en donde por citar un ejemplo las *vlans* (Virtual Lans) constituyen switches lógicos que conforman redes LAN independientes que están en una misma estructura física, es decir en un único switch físico. El *GNS3* [6] además de permitir trabajar con los distintos elementos de *networking* y las máquinas virtuales, básicamente computadoras, incorpora un nuevo paradigma que es la posibilidad de incorporar *Appliances* [7]. Un *appliance* consiste en una estructura de hardware que puede emularse a través de una máquina virtual y ser utilizada en una plataforma de simulación, como el *GNS3*, comportándose como si se tratara del equipo físico real, con la consiguiente reducción de costos, ya que en realidad se tiene una virtualización del equipo real, pero que da la apariencia de estar trabajando con el equipo físico real verdadero. Así por ejemplo podemos tener *Appliances* de costosos equipos simulados en esta plataforma y también resulta beneficioso para las empresas porque aumentan sus ventas.

1.1 Ventajas y desventajas de emplear una plataforma *GNS3* frente a la utilización de hardware de red real

Con el *GNS3* podemos emular el comportamiento de una red real utilizando recursos de software (*Appliances*) en vez de los costosos equipos de hardware tal como lo haríamos en forma tradicional. Por lo tanto, algunas de las ventajas que podríamos citar son las siguientes:

- Realizar pruebas de diferentes topologías complejas de red previa a la puesta en producción del equipamiento real.
- Probar actualizaciones y cambios de configuraciones de software en los equipos virtualizados antes de realizarlas en equipos reales y productivos, evitando complicaciones e inconvenientes en la infraestructura real de los clientes.
- Facilitar la enseñanza y el aprendizaje al poder crear laboratorios virtuales idénticos a los entornos reales, permitiendo a los estudiantes reproducir los diferentes escenarios en sus computadoras personales.

Por otro lado, como desventajas de la plataforma *GNS3* podríamos citar las siguientes:

- No se puede probar ni medir la performance de la infraestructura real de red ya que la misma al estar virtualizada presenta una degradación natural con respecto a la estructura original.
- Se requiere de un hardware bastante potente para emular configuraciones complejas de red, esto significa concretamente gran poder de procesamiento y de memoria RAM.

2 Descripción de la plataforma GNS3 y sus programas asociados

El GNS3 es un simulador gráfico de redes que permite emular dispositivos de la línea CISCO. Entre los dispositivos que pueden emularse tenemos routers, switches, firewalls y el mayor valor agregado lo constituye la implementación de máquinas virtuales. En la figura 1 observamos la pantalla principal de trabajo de la plataforma:

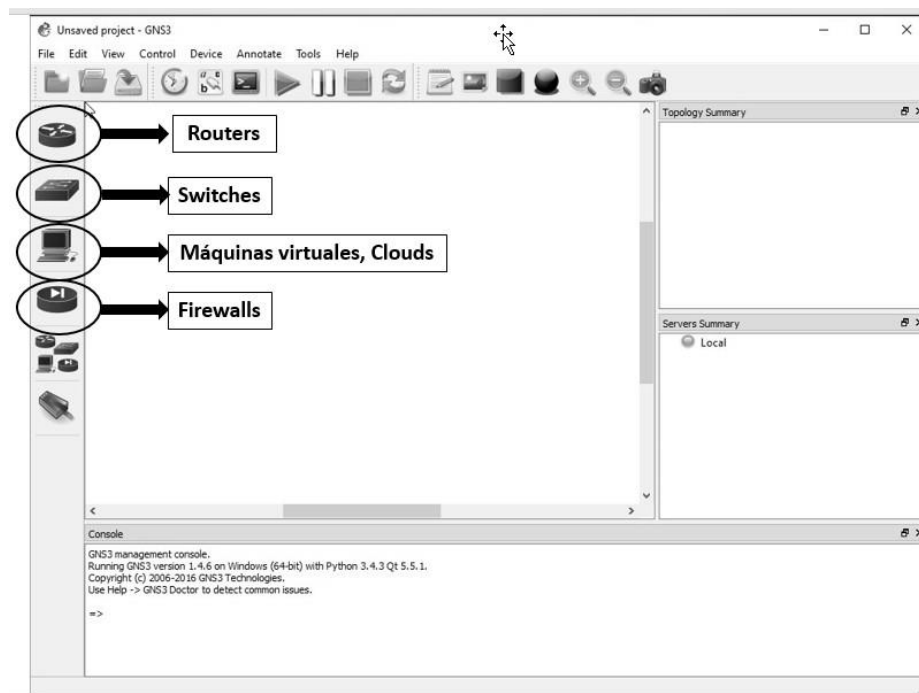


Fig. 1. Pantalla principal de trabajo de la plataforma GNS3

Según la Fig. 1 podemos elegir distintos tipos de routers, switches, máquinas virtuales o Firewalls. Entre los routers, se recomienda el uso de las siguientes plataformas:

c3640, c3660, c3725, c3745, y c7200. El programa asociado a la emulación de los routers CISCO se denomina *Dynamips* [8].

2.1 Configuración

La configuración del GNS3 se lleva a cabo editando las preferencias, en donde se encuentran los siguientes elementos a destacar (ver Fig. 2):

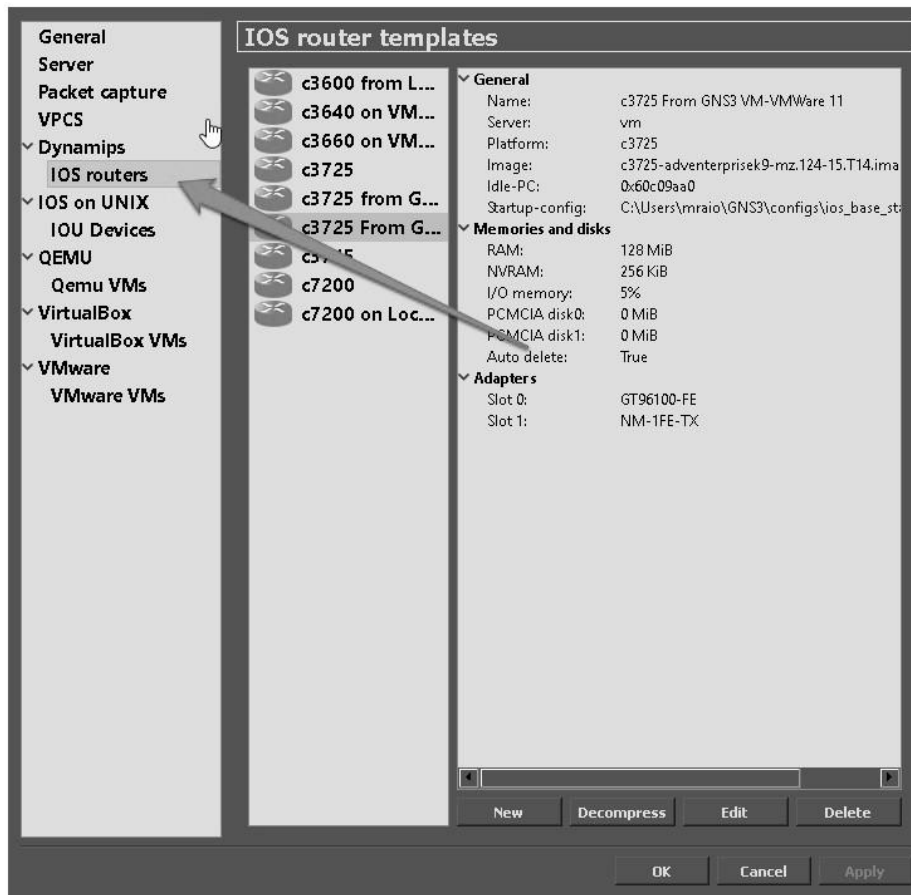


Fig. 2. Configuración de las preferencias de la plataforma GNS3

1. *Server*: en el server se guardará toda la configuración, puede elegirse la opción local, server que emplea la dirección de local host 127.0.0.1 y puerto TCP 8000, GNS3 Virtual Machine server, que emplea una como servidor una máquina virtual que suministra GNS3 o remote server, que permite configurar un servidor externo.

2. *IOS Routers*: desde aquí se selecciona el IOS del router CISCO correspondiente; las posibles plataformas se ven en la Fig.2
3. *Qemu VMs, Virtual Box VMs, VMware VMs*: desde aquí se seleccionan las diferentes máquinas virtuales a emplear en las topologías de red. En el desarrollo experimental que explicaremos más adelante utilizamos VMware Workstation.

3 Diseño de una experiencia de laboratorio empleando GNS3

Se sintetizará mediante el GNS3 la siguiente topología de red:

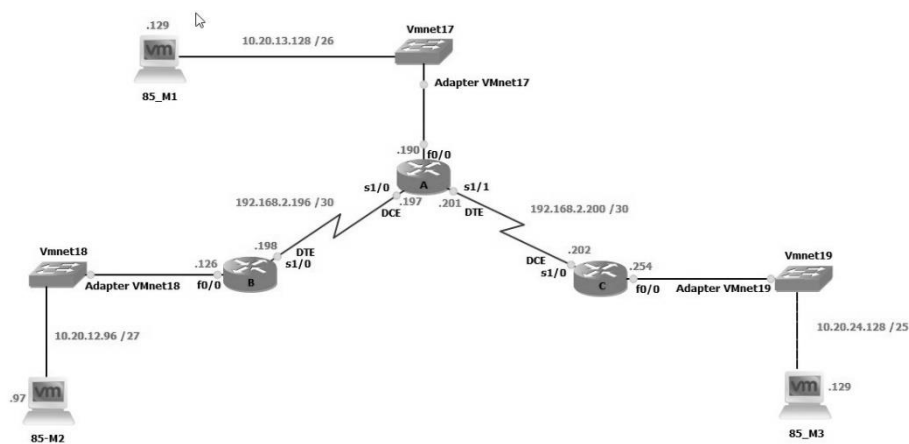


Fig. 3. Topología de red a implementar mediante GNS3

En esta topología tenemos tres routers cisco 3725 (denominados A, B y C). El router A enlaza con los routers B y C mediante un enlace punto a punto simulado a través de una conexión back to back (conexión sincrónica empleando un cable DTE y un DCE en cada extremo del router). Tenemos tres redes LAN y dos redes WAN en la topología. Las direcciones de red y de hosts con sus respectivas máscaras son las siguientes:

LAN A	10.20.13.128 /26	WAN A-B	192.168.2.196 /30
LAN B	10.20.12.96 /27	WAN A-C	192.168.2.200 /30
LAN C	10.20.24.128 /25		

Table 1. Direcciones y máscaras de las redes LAN y WAN de la Fig. 3

Routers	LAN f0/0	WAN s1/0	WAN s1/1
A	10.20.13.190 /26	192.168.2.197 /30	192.168.2.201 /30
B	10.20.12.126 /27	192.168.2.198 /30	
C	10.20.24.254 /25	192.168.2.202 /30	
Máquinas Virtuales	IP host		

85_M1	10.20.13.129 /26
85_M2	10.20.12.97 /27
85_M3	10.20.24.129 /25

Table 2. Direcciones de hosts y máscaras correspondientes a la topología de la Fig. 3

Los switches denominados Vmnet17, Vmnet18 y Vmnet19, corresponden a switches virtuales que utiliza el VMware y su explicación la daremos en el siguiente apartado.

4 Desarrollo de la experiencia

En este apartado se aportará la experiencia de vislumbrar la *interacción* del mundo del networking con las máquinas virtuales. Para ello primero haremos una síntesis preliminar de las herramientas utilizadas y finalmente presentaremos las capturas más relevantes de los resultados obtenidos.

4.1 Herramientas utilizadas

Para la síntesis de la topología de red mencionada con anterioridad emplearemos los siguientes elementos de hardware y software:

- Servidor HP Proliant DL-120 G6 con 12 GB de RAM
- Sistema operativo host Windows 10
- Plataforma de software GNS3 versión 1.5.1
- VMware Workstation 12 Pro version 12.1.1 build-3770994
- Tres routers 3725
- Tres máquinas virtuales con sistema operativo Debian versión 8.5 sin entorno gráfico, denominadas 85_VM1, 85_VM2 y 85_VM3.
- Tres switches virtuales denominados VMnet17, VMnet18 y VMnet 19 aportados por la aplicación VMware Workstation.

Cada router 3725 posee las siguientes placas de hardware:

1. Slot 0: una NM-1FE-TX, identificada como f0/0
2. Slot 1: una NM-4T, identificadas como s1/0, s1/1, s1/2 y s1/3

La configuración de cada máquina virtual es la siguiente:

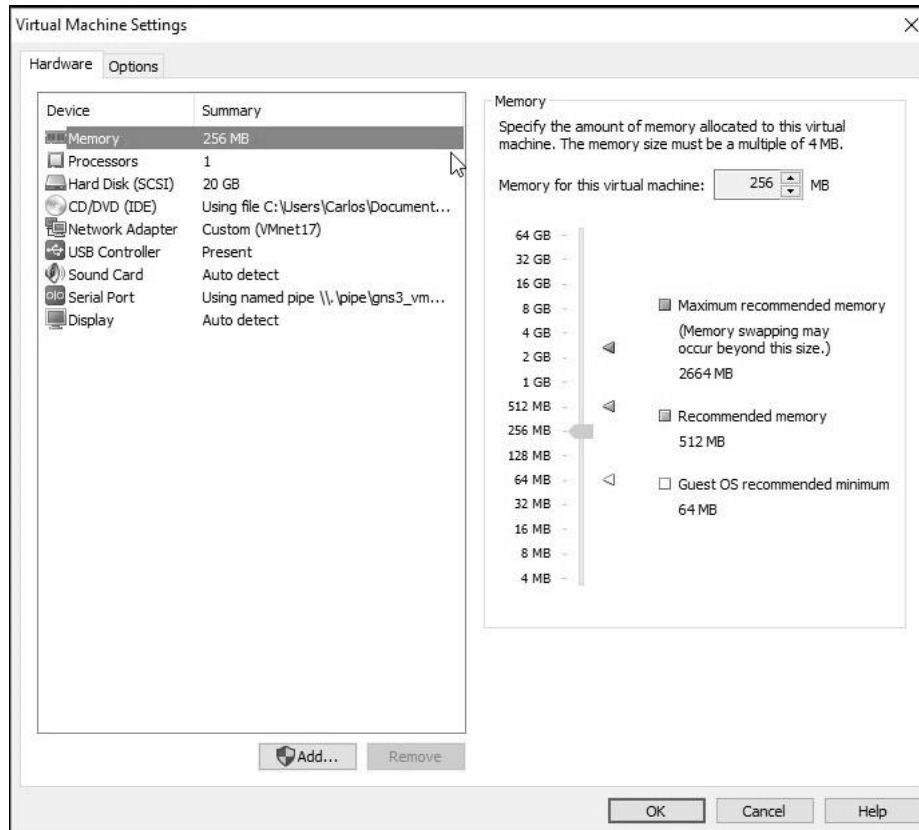


Fig. 4. Configuración de hardware correspondiente a la Máquina Virtual 85_VM1

Observese que se han asignado sólo 256 MB de ram a cada Máquina virtual (no se hace uso del entorno gráfico, ya que no resulta necesario para esta experiencia). Las configuraciones para las máquinas virtuales 85_VM2 y 85VM_3 es idéntica a la de la85_VM1 excepto que ambas máquinas se encuentran conectadas a los switches virtuales VMnet 18 y VMnet 19 respectivamente.

4.2 Configuración de los routers CISCO 3725

En todos los routers se ha configurado RIP versión 2, ya que estamos trabajando con subnetting. A continuación se muestran las tablas de ruteo correspondientes a cada router:

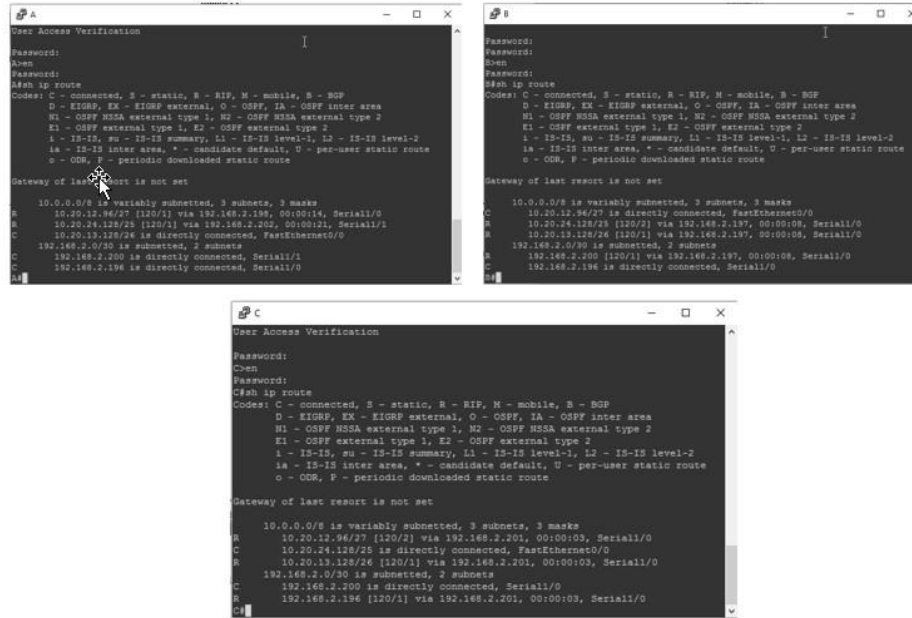


Fig. 5. Tablas de ruteo correspondientes a los routers A, B y C

4.3 Vinculación de los elementos de red con las máquinas virtuales

Para establecer la comunicación entre los dispositivos de red y las máquinas virtuales, en este caso los routers, se debe emplear un *cloud* como elemento auxiliar. En la solapa Ethernet del cloud se elige con qué interface de red se van a vincular los routers con las máquinas virtuales. Como las máquinas virtuales en VMware están asociadas precisamente a los switches denominados VMnet, es que se eligen estas interfaces. Cabe destacar que para que todo esto funcione apropiadamente deberán configurarse las placas de red tanto desde el GNS3, como desde el VMware. A tal efecto el VMware posee un editor de redes que simplifica mucho esta tarea. Con posterioridad, al cloud puede cambiarse el símbolo que lo identifica, siendo en este caso el más apropiado el de un switch.

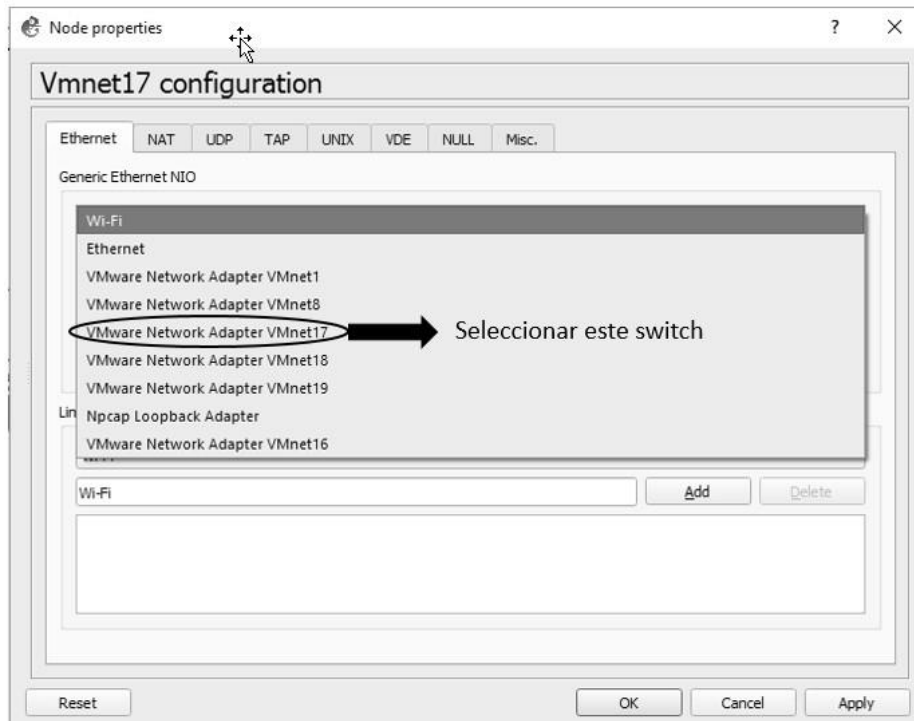


Fig. 6. Configuración del Cloud VMnet17

Idéntica situación deberá realizarse para vincular las dos máquinas restantes 85_VM2 y 85_VM3, es decir hay que crear dos clouds más y asociarlas respectivamente a las interfaces de red vinculadas a los switches VMnet18 y VMnet19 respectivamente.

4.4 Resultados alcanzados

A continuación, mostramos tres capturas realizadas desde cada máquina virtual enviando ping hacia las otras máquinas virtuales. Con esto se corrobora el funcionamiento del laboratorio.

```

root@85_M1:~# ping 10.20.24.129
PING 10.20.24.129 (10.20.24.129) 56(84) bytes of data:
64 bytes from 10.20.24.129: icmp_seq=1 ttl=62 time=45.9 ms
64 bytes from 10.20.24.129: icmp_seq=2 ttl=62 time=57.4 ms
64 bytes from 10.20.24.129: icmp_seq=3 ttl=62 time=67.1 ms
64 bytes from 10.20.24.129: icmp_seq=4 ttl=62 time=69.0 ms
64 bytes from 10.20.24.129: icmp_seq=5 ttl=62 time=66.3 ms
64 bytes from 10.20.24.129: icmp_seq=6 ttl=62 time=63.5 ms
64 bytes from 10.20.24.129: icmp_seq=7 ttl=62 time=63.8 ms
64 bytes from 10.20.24.129: icmp_seq=8 ttl=62 time=49.0 ms
64 bytes from 10.20.24.129: icmp_seq=9 ttl=62 time=69.5 ms
64 bytes from 10.20.24.129: icmp_seq=10 ttl=62 time=69.0 ms
64 bytes from 10.20.24.129: icmp_seq=11 ttl=62 time=85.0 ms
64 bytes from 10.20.24.129: icmp_seq=12 ttl=62 time=70.8 ms
64 bytes from 10.20.24.129: icmp_seq=13 ttl=62 time=70.0 ms
64 bytes from 10.20.24.129: icmp_seq=14 ttl=62 time=53.2 ms
^C
--- 10.20.24.129 ping statistics ---
14 packets transmitted, 14 received, 0% packet loss, time 1314ms
rtt min/avg/max/mdev = 45.946/65.038/85.041/9.745 ms
root@85_M1:~# _

root@85_M2:~# ping 10.20.24.129
PING 10.20.24.129 (10.20.24.129) 56(84) bytes of data:
64 bytes from 10.20.24.129: icmp_seq=1 ttl=61 time=60.4 ms
64 bytes from 10.20.24.129: icmp_seq=2 ttl=61 time=99.5 ms
64 bytes from 10.20.24.129: icmp_seq=3 ttl=61 time=84.7 ms
64 bytes from 10.20.24.129: icmp_seq=4 ttl=61 time=83.3 ms
64 bytes from 10.20.24.129: icmp_seq=5 ttl=61 time=99.0 ms
64 bytes from 10.20.24.129: icmp_seq=6 ttl=61 time=69.4 ms
64 bytes from 10.20.24.129: icmp_seq=7 ttl=61 time=68.8 ms
64 bytes from 10.20.24.129: icmp_seq=8 ttl=61 time=99.3 ms
^C
--- 10.20.24.129 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7086ms
rtt min/avg/max/mdev = 60.429/83.108/99.581/14.564 ms
root@85_M2:~#

root@85_M3:~# ping 10.20.13.129
PING 10.20.13.129 (10.20.13.129) 56(84) bytes of data:
64 bytes from 10.20.13.129: icmp_seq=1 ttl=62 time=61.3 ms
64 bytes from 10.20.13.129: icmp_seq=2 ttl=62 time=68.2 ms
64 bytes from 10.20.13.129: icmp_seq=3 ttl=62 time=68.2 ms
64 bytes from 10.20.13.129: icmp_seq=4 ttl=62 time=54.5 ms
64 bytes from 10.20.13.129: icmp_seq=5 ttl=62 time=68.4 ms
64 bytes from 10.20.13.129: icmp_seq=6 ttl=62 time=59.5 ms
64 bytes from 10.20.13.129: icmp_seq=7 ttl=62 time=53.2 ms
64 bytes from 10.20.13.129: icmp_seq=8 ttl=62 time=50.1 ms
^C
--- 10.20.13.129 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7067ms
rtt min/avg/max/mdev = 50.193/59.721/68.450/7.259 ms
root@85_M3:~# _
    
```

Fig. 7. Prueba de Ping entre las máquinas virtuales

5 Conclusiones y trabajo futuro

1. A través del GNS3 es posible conectar una estructura virtual de red con redes físicas por medio de los clouds, lo que hace a esta plataforma sumamente potente.
2. Esta es una presentación sencilla en donde se pretende mostrar la enorme performance de este simulador e insistimos se pone el énfasis en la interacción con las máquinas virtuales, que otros simuladores no pueden realizar, como por ejemplo el paquet tracer de CISCO.
3. Se avizora un enorme futuro para la implementación de redes basadas en IPv6.
4. Lo único negativo a destacar es que para hacer un uso fructífero de GNS3 debe tenerse un hardware poderoso y mucha RAM, se aconseja un mínimo de 16 Gb.

6 Referencias

1. Virtualización con VMWare: Manuales USERS (Spanish Edition). Publicación: 20 de enero de 2013 - ISBN: 978-9871857715.
2. Virtualization Essentials 1st Edition. Publicación: 1º mayo 2012. ISBN 13: 978-1118176719
3. <https://www.virtualbox.org>
4. <http://www.vmware.com/mx/products/workstation.html>
5. http://wiki.qemu.org/Main_Page
6. <https://www.gns3.com>
7. <https://www.gns3.com/marketplace/appliances>
8. GNS3 Network Simulation Guide. Publicación: 25 Octubre 2013. ISBN 13: 978-1782160809.

Utilización de los Espacios Blancos: Una solución al problema de la saturación del espectro de frecuencias

Antonio Castro Lechtaler^{1,2}, Antonio Foti^{3,4}, Alejandro Arroyo Arzubi¹, Jorge García Guibout⁵, Fernanda Carmona⁶, Rubén Fusario^{1,2}, Alejandro Oliveros³.

¹EST - Facultad de Ingeniería del Ejército, C1426 Ciudad de Buenos Aires; ²CISTIC/FCE - Universidad de Buenos Aires, C1120 Ciudad de Buenos Aires; ³Universidad Nacional de Tres de Febrero, B1674 Sáenz Peña, Provincia de Buenos Aires; ⁴Universidad Nacional del Oeste, B1718, San Antonio de Padua, Provincia de Buenos Aires; ⁵Instituto Tecnológico Universitario - Universidad Nacional de Cuyo, M5500 Mendoza, Provincia de Mendoza; ⁶Universidad Nacional de Chilecito, F5360 Chilecito, Provincia de la Rioja} {Argentina}

[antonio.castrolechtaler,antonio.foti,aarroyoarzubi,fbcarmona64,rfusario}@gmail.com](mailto:antonio.castrolechtaler,antonio.foti,aarroyoarzubi,fbcarmona64,rfusario@gmail.com); aoliveros@untref.edu.ar; jgarcia@itu.uncu.edu.ar

Abstract. La necesidad de proveer el Servicio Universal de Comunicaciones en áreas rurales como en zonas de baja densidad poblacional que no cuentan ningún servicio de comunicaciones ha ido llevando a buscar, dentro de las nuevas tecnologías, aquellas que usando los espacios blancos evitan aumentar la congestión del espectro de frecuencias. Esas están basadas en dos técnicas conocidas como *Cognitive Radio* y *Software Defined Radio*. Las frecuencias utilizadas por los Sistemas Digitales de Televisión por Radiodifusión aparecen como una oportunidad para desplegar este tipo de equipos, que combinan técnicas de telecomunicaciones con aplicaciones informáticas a través del uso de software específico. Estos equipos de la Recomendación 802.22, además, toman elementos para la Capa de Enlace de la norma 802.3, de amplia difusión y conocimiento.

Palabras Claves: Espacios Blancos, 802.22, Espectro de Frecuencias, Comunicaciones Rurales, *Whitespace Alliance*.

1 Introducción

El Proyecto Redes Privadas Comunitarias [1] financiado por el FONCyT - ANPCyT apuntó a investigar, testear y comentar diferentes tecnologías para proveer enlaces de comunicaciones a comunidades aisladas con baja densidad poblacional, y por lo tanto sin interés comercial para las empresas que prestan este servicio.

A través de ese trabajo realizado [2], [3], [4], [5], [6], se encontró que en la Argentina hay localidades de baja densidad poblacional (100 habitantes o aún menos en algunos casos) que sufren la falta del servicio de energía eléctrica durante las 24 horas y no son alcanzadas por la red telefónica tanto fija, como la de equipos celulares, con la consecuente falta de acceso a redes de datos e Internet.

Muchos de estos parajes están ubicados en las cercanías de la red ferroviaria, hoy fuera de servicio. Estas circunstancias dificultan que proveedores de servicios inviertan en enlaces de comunicaciones que solucionen estos problemas. La idea es estudiar enfoques alternativos que representen una solución a este tipo de problemas, a costos razonables.

Por consideraciones prácticas y económicas se ha pensado y experimentado en soluciones digitales inalámbricas. Las razones que llevaron a su consideración están relacionadas con la rapidez y facilidad con que se pueden instalar estos enlaces, y en que, al utilizar frecuencias que no requieren de una autorización previa de la autoridad de aplicación, es mucho más simple su uso. Adicionalmente, se busca evitar la dependencia de un operador de telecomunicaciones local.

El trabajo realizado presentó las experiencias realizadas en las que se analizaron casos para distancias de cobertura importantes.

En este último caso, las mismas difieren de las utilizadas en las soluciones requeridas en las grandes ciudades, por estar éstas pensadas para cortas distancias, como fue el caso originalmente de la tecnología 802.11, [7].

Paralelamente, y dado el alto grado de saturación que presenta el espectro de frecuencias se está actualmente analizando a nivel nacional e internacional, la forma de reutilizarlo de una mejor manera, para lo cual se han ido desarrollando nuevas tecnologías más acordes al estado actual del arte, como es el caso de los equipos ya existentes en el mercado que cumplen con la Recomendación 802.22, de la IEEE [8].

Las sociedades modernas están confiando y utilizando cada vez más el espectro radioeléctrico. La omnipresencia de los servicios inalámbricos y dispositivos de comunicación, tales como teléfonos móviles, comunicaciones de la policía, Wi-Fi y recientemente los de radiodifusión de Televisión Digital Abierta de Alta Definición (entre otros) son ejemplos de esta dependencia. El espectro de frecuencias se ha convertido en uno de los recursos más necesarios y escasos de los tiempos modernos.

El crecimiento de la demanda mundial para el tráfico de datos móviles se ha incrementado en los últimos cuatro años a tasas que en muchos casos superan el 100%; y la tasa de crecimiento esperado es aún mayor [9]. Actualmente se estima en el orden de 134 Exabyte¹ anuales, lo que supone un incremento interanual del 66 por ciento para el periodo (2012-2017) [10].

El intenso uso del espectro de hasta 10 GHz y a futuro a frecuencias aún más altas ha dado lugar a una revisión de las políticas de regulación del mismo. A causa del congestionamiento en esas frecuencias, se ha intensificado el estudio de los llamados Espacios Blancos - White Space.

La Conferencia Europea de Administraciones de Correos y Telecomunicaciones - CEPT, ha definido a un Espacio Blanco como: *“Una porción del espectro que está disponible para ser utilizado por una aplicación de radiocomunicaciones en un momento dado, para una zona geográfica determinada, en forma simultánea con otra, pero con la característica de que la misma no produzca interferencias con respecto a los servicios que poseen una prioridad más alta a nivel nacional en esas mismas frecuencias”*[11].

En la actualidad, son intensos los esfuerzos de investigación realizados a nivel de organizaciones no gubernamentales, países y empresas de Telecomunicaciones para darle utilidad a esta parte del espectro que es desaprovechado. La importancia del tema, ha dado lugar a la creación de la *“Whitespace Alliance”*² cuya misión es *“Promover el desarrollo, despliegue y uso de estándares basados en productos y*

¹ EB (Exabyte) = 1.000 PB (Petabyte) = 1.000.000 TB (Terabyte).es promover el desarrollo, despliegue y uso de estándares basados en productos y servicios como un medio para proporcionar capacidades de banda ancha a través de espectro de espacios en blanco.

²<https://www.whitespacealliance.org/>

servicios como un medio para proporcionar capacidades de banda ancha a través de los espacios blancos existentes en el espectro de frecuencias”.

Dada la inversión y el fuerte despliegue que la República Argentina ha efectuado al crear e instalar el sistema de Televisión Digital Abierta - TDA, el mismo (infraestructura existente que) ofrece una inmejorable oportunidad para solucionar el problema de las comunicaciones rurales o de poblaciones aisladas o con baja densidad poblacional utilizando los espacios blancos que ese sistema permite reutilizar.

2 Estado de la situación actual

A nivel mundial dos de cada tres personas no tienen acceso a la Red Internet y más de la mitad de la población del mundo vive en zonas rurales sin acceso a banda ancha³ o con muchas dificultades para obtener este tipo de servicios. Las razones son variadas. Por un lado es costoso cablear con cobre o fibra zonas rurales, aisladas o remotas, especialmente con baja densidad de poblacional. Las soluciones satelitales si bien son posibles, los costos de instalación, servicio y mantenimiento no son aun razonables para los posibles usuarios de estos servicios. Luego, las soluciones radioeléctricas terrestres aparecen como las más viables.

En muchos países, los proveedores de servicios inalámbricos tradicionales se han centrado en las zonas urbanas con alta densidad de población que proporcionan un rápido retorno de las inversiones necesarias para prestar el servicio. Pero aun en casos en que se realicen las mismas a pérdida, aparece un problema adicional que es determinar y asignar una porción del espectro de frecuencias para su explotación [12]; y ese no es un problema menor, dado que el espectro siempre se presenta como un bien escaso.

Por otra parte, las tecnologías existentes hasta hace muy poco tiempo, no han podido proporcionar una cobertura usando medios radioeléctricos que tengan un alcance importante, y máxime, cuando no existe una línea de visión directa. La experiencia de Corral de Lorca [5] [6] utilizando la Recomendación 802.11, si bien resultó interesante, demostró que la existencia de barreras verdes formadas por montes de cierta altura y las distancias requeridas conspiraba con la posibilidad de obtener niveles de señales interesantes para prestar un servicio continuo.

Una de las razones por la que se ha formado la WhiteSpace Alliance ha sido precisamente promover y buscar soluciones para convertir esta brecha digital en una oportunidad mediante el aprovechamiento no utilizado o subutilizado del espectro de frecuencias mediante nuevas normas de tecnologías de banda ancha que permitan proporcionar conectividad a un costo razonable, facilitando el uso de los espacios blancos y ayudando a poner en marcha la interoperabilidad de los sistemas.

Por otra parte, en muchos países como la Argentina, la transición de la televisión analógica a la digital puede proporcionar la oportunidad para salvar la brecha señalada. Al digitalizar cada canal de televisión analógica a digital se pueden obtener, según la norma utilizada, hasta 5 señales de definición estándar de televisión digital.

El exceso de espectro es a menudo llamado el "*dividendo digital*" y puede ser utilizado para proporcionar acceso de banda ancha, en tanto y en cuanto, no se

³<http://www.internetworldstats.com/stats.htm>

produzcan interferencias a los usuarios con bandas asignadas por la autoridad regulatoria.

A su vez, los canales utilizados por las estaciones de televisión en las bandas de VHF / UHF tienen características de propagación altamente favorables para el alcance de larga distancia.

Las autoridades regulatorias de los países están actualmente estableciendo normas que permiten el uso sin licencia del espectro generado por los espacios blancos, en tanto y cuanto, su uso no interfiera con los receptores de televisión.

Los equipamientos que pueden cumplir con tales requerimientos utilizan técnicas de radio cognitiva, y al utilizar los espacios en blanco que permiten los canales de televisión obtienen del orden de diez veces mayor alcance, que las soluciones de acceso inalámbrico en bandas que están por encima de 1 GHz.

3 Nuevas tecnologías para el uso de los espacios blancos

3.1 Introducción

El problema de contar con equipamientos que puedan ser utilizados en los espacios blancos ha movilizó a distintas empresas de telecomunicaciones y a grupos de investigación a buscar soluciones que den respuesta a estos desafíos.

El uso de este tipo de equipamientos permitirá dar comunicaciones a zonas muy especiales como por ejemplo las rurales. Éstas en muchos países revisten significación por cuanto proporcionan cantidades significativas de productos alimenticios en sus diferentes etapas de fabricación y constituyen una trascendente fuente de productos básicos de exportación e ingresos de divisas. Para muchos países participan generando un porcentaje significativo del producto bruto interno de ellos.

La serie de Recomendaciones 802.XX incluye un conjunto de normas que regulan el funcionamiento de las comunicaciones inalámbricas. Algunas de ellas fueron estudiadas y evaluadas en la experiencia de Corral de Lorca ya señalada.

Posteriormente a la norma 802.11 fueron apareciendo distintas tecnologías que permitieron ampliar esta Serie merced al trabajo de distintos grupos de investigación.

Estos trabajos culminaron el 1 de julio de 2011 cuando finalizó el proceso de aprobación, del estándar “IEEE 802.22 - “IEEE 802.22: Cognitive Wireless Regional Area Network - Medium Access Control (MAC) and Physical Layer (PHY). Specifications: Policies and Procedures for Operation in the TV Band⁴”. La misma fue aprobada con el apoyo del Comité LAN/MAN⁵ de la IEEE [8].

Este nuevo estándar proporciona una opción que permite establecer enlaces inalámbricos full dúplex a distancias de entre 30 a 70 km entre antenas, utilizando frecuencias no restringidas por las regulaciones gubernamentales.

⁴“IEEE 802.22 - Red de Área Regional Cognitiva Inalámbrica de Control de Acceso al Medio MAC) y la Capa Física (PHY). Especificaciones, Políticas y Procedimientos para la Operación en las Bandas de Televisión”.

⁵ LAN: Local Area Network; MAN: Metropolitan Area Network.

La norma que pertenece a la serie 802.XX⁷⁸ tiene por objeto establecer los criterios para el despliegue de múltiples productos interoperables de la misma, ofreciendo acceso a la banda ancha fija en diversas áreas geográficas, incluyendo especialmente los de baja densidad de población en las zonas rurales, y evitar la interferencia a los servicios que trabajan en la televisión bandas de radiodifusión.

La misma es conocida actualmente como Red Inalámbrica de Área Regional (WRAN) y está pensada para operar principalmente como una forma de poder acceder a servicios de banda ancha a redes privadas de datos ubicadas en Zonas Rurales.

3.2 Características Generales de la Norma 802.22

La recomendación, además de proporcionar una solución al problema que nos ocupa, permite resolver dos problemas que afectan seriamente el uso del espectro de frecuencias. Uno es el problema de los llamados Espacios Blancos, y otro, el de las interferencias entre canales contiguos, ambos cada día más frecuentes debido al uso intensivo de las comunicaciones inalámbricas para todo tipo de servicios de comunicaciones.

Dichas características son desde el punto de vista de los sistemas de información muy interesantes, pues combinan un problema de comunicaciones, como es el uso adecuado del espectro, con el desarrollo de software especial que permite emplear las técnicas denominadas radio cognitiva

Las mismas moderan la interferencia que pueden ocasionar otros operadores existentes que trabajan en las mismas frecuencias y otorgan capacidad de geolocalización. Esto se ejecuta a través del acceso a una base de datos de los servicios establecidos, y así para detectar la presencia de otros servicios a través de la tecnología de espectro de detección. Para ello la norma señalada, se utiliza en combinación con otra conocida como WRAN o por su denominación IEEE 802.22.1.

Los sistemas WRAN se sirven de canales que van desde 54 a 862 MHz en las bandas de VHF y UHF. El uso de las tecnologías de radio cognitiva permite utilizar los espacios ubicados entre dos canales de TV abierta evitando la interferencia de estos servicios con las estaciones de televisión. Ambos operan en las mismas bandas.

La idea es utilizar las frecuencias que se han asignado para la transmisión del Sistema Nacional de Televisión Abierta integrándolo a un sistema basado en estas normas. Esto permitiría su uso en las comunicaciones rurales y en otros casos similares.

Estos esfuerzos de investigación realizados a nivel de distintas organizaciones para darle utilidad a esta parte del espectro están permitiendo posibles soluciones para el creciente tráfico.

Sin duda, requerirán la revisión y rediseño del marco regulatorio vinculado con la radiodifusión y del servicio celular, la mejora de los estándares de compresión, la sustitución de varios servicios inalámbricos -para que sus señales se transporten por satélite o por cable-, la solución a este problema será sin duda el acceso dinámico

⁷ Redes Inalámbricas.

utilizando las tecnología conocida como radio cognitiva, tecnología ésta que utiliza la norma 802.22.

La Tecnología de Radio Cognitiva (CRT) se considera actualmente una de las mayores posibilidades existentes para hacer frente a la creciente escasez de espectro y está orientada a aprovechar las frecuencias subutilizadas, es decir los huecos temporales de las señales primarias, y los diferentes tipos de espacios en blanco.

Si bien se encuentra ya desarrollada para variadas aplicaciones, a medida que el mismo avance cada vez más en su desarrollo podrá proporcionar tecnologías para una variedad de aplicaciones tales como: banda ancha rural, de seguridad pública y de respuesta de emergencia, uso de frecuencias urbanas, etc.

Esta tecnología también tendrá consecuencias significativas para la detección dinámica y gestión del espectro.

3.3 Software Definido por Radio

El crecimiento exponencial de las formas y los medios por los que la gente necesita comunicarse utilizando dispositivos inalámbricos ha llevado a que estos sean diseñados teniendo en cuenta dos aspectos centrales: *facilidad de operación*⁹ y una relación costo/prestaciones conveniente. Ambos, se han transformado en factores críticos para el éxito de un producto.

Una nueva tecnología conocida como “*Software Definido para Radio - SDR*”¹⁰ proporciona flexibilidad y rentabilidad tanto a usuarios finales, como beneficios a los proveedores de servicios y a los desarrolladores de productos [14]. Un foro especial que se ha formado, denominado “Wireless Innovation Forum”¹¹, integrado por investigadores, fabricantes de equipos de telecomunicaciones y operadores de distintos servicios, se ocupa de impulsar estas nuevas tecnologías.

Este foro ha definido el concepto de Software Defined Radio como “*equipos de radiocomunicaciones en los que una parte o la totalidad de las funciones de la capa física son ejecutadas por programas de software*”.

La radio es un dispositivo que transmite o recibe señales inalámbricas utilizando una porción del espectro de radio. Dispositivos de radio tradicionales basados exclusivamente en hardware (por ejemplo: mezcladores, filtros, amplificadores, moduladores/demoduladores y detectores) poseen la limitación de que, debido a que sus características, se pueden modificar solamente por intervención física.

Por el contrario, un equipo que utilice la tecnología Radio Definida por Software - SDR podrá ejecutar muchas de sus funciones por medio de un software específico ya sea sobre un computador o a través de un sistema embebido.

El concepto no es nuevo, pero las capacidades generadas a causa de la rápida evolución que ha tenido electrónica digital ha permitido que muchos procesos que antes eran sólo teóricamente factible hoy día sean ejecutados en forma práctica utilizando estos recursos [15].

Bajo esta tecnología, el software ha demostrado ser eficiente, y a un costo relativamente barato. Por otra parte, se pueden realizar mejoras continuas en los

⁹También describirse como “*manejo intuitivo*”.

¹⁰Software Defined Radio.

¹¹<http://www.wirelessinnovation.org/>

equipos a través de actualizaciones del software. En muchos casos, el software ya gestiona todas o la mayoría de las funciones que se requieren para operar el equipo (incluyendo, como se expresó) el procesamiento de la capa física.

3.4 Radio Cognitiva¹².

A finales de la década de los años 90, Joseph Mitola¹³ y Gerald Maguire, investigadores del Instituto Real de Tecnología de Estocolmo desarrollaron una tecnología que llamaron *Cognitive Radio*. La misma es una mejora de un trabajo anterior que habían presentado sobre trabajos relacionados con *Software Defined Radio* [15] [16].

Si bien Radio Definida por Software ofrece un gran potencial, es cierto que también se necesita una importante capacidad de procesamiento que puede limitar su flexibilidad, además de requerir una adecuada respuesta de la red.

La tecnología de Radio Cognitiva consiste en la introducción en el funcionamiento de los equipos de software de comunicaciones embebido utilizando el lenguaje “*Radio Knowledge Representation Language – RKRL*”.

Este puede considerarse un sistema inteligente y eficaz para las comunicaciones de radio y el funcionamiento de los protocolos. Consiste en proporcionar mecanismos basados en el uso de tecnología inteligente que permiten optimizar el espectro de frecuencias.

Como se mencionó, el uso y la asignación de frecuencias en un espectro saturado no es óptimo, originándose espacios blancos, en especial en las bandas que utilizan los operadores de la TV Digital Abierta.

Entre otras, esas fueron las razones que llevaron a desarrollar la tecnología cognitiva de radio para las comunicaciones inalámbricas, la que busca detectar -y luego utilizar- las partes del espectro de radiofrecuencia empleados ineficientemente, permitiendo su reutilización sin causar interferencias con los servicios que se les asignen.

Este procedimiento logra, mediante la asignación de frecuencia variable, que otros servicios se aprovechen de las partes no utilizadas del espectro, como trabaja por ejemplo la Recomendación 802.22 para ocupar los espacios blancos.

El uso de software inteligente Cognitive Radio analiza periódicamente el espectro en busca de *espacios blancos*, detecta el uso dado a cada uno de ellos, y luego determina si son reutilizables. Si lo son el sistema funciona cambiando los parámetros del transmisor basados en la interacción con el medio ambiente.

Tiene la capacidad y la tecnología para capturar o detectar la información de otros equipos de radio que están trabajando en la misma frecuencia y mediante sistemas de programación dinámica reconfiguran las frecuencia de transmisión permitiendo que se pueda transmitir y recibir en una variedad de frecuencias y de utilizar diferentes tecnologías de acceso de transmisión soportados por su diseño de hardware.

¹² Mitola define cognitive como “*la combinación de conocimiento declarativo y de procedimiento en un sistema que aprende de su propia experiencia*”.

¹³ Joseph Mitola III recibió su doctorado en dicho Instituto con su tesis: *Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio*.

Desde la primera versión 802.3 hasta el presente, se han abordado diferentes requisitos en las comunicaciones inalámbricas.

Figura 1 ilustra los estándares inalámbricos más significativas, y la posición relativa de estándar 802.22.

Se puede observar que los distintos estándares cubren radios que pueden ser pequeños, entendido por tales los que se extienden hasta 10 m y de mayor tamaño como el que está analizando que llega a cubrir hasta 100 km.

5 Pruebas de Campo

El Ente Nacional de Comunicaciones - ENACOM ha recientemente aprobado el “*Reglamento General del Servicio Universal*” por medio de la Resolución N° 2642/2016 de fecha 17 de mayo de 2016. Dicho Reglamento en su articulado, “*Título III, Artículo 19° Programas, apartado c. Conectividad en zonas Rurales y zonas en geográficas desfavorables para el desarrollo de servicios TIC*” prevé desarrollar aplicaciones, utilizando las modernas tecnologías radioeléctricas disponibles con el fin de permitir la distribución de señales digitales que puedan transportar voz, datos e Internet en zonas rurales, aprovechando parcialmente la infraestructura del Sistema Argentino de Televisión Digital Terrestre (SATVD-T).

Dicho Ente, luego de haber tomado conocimiento del trabajo que viene realizando este Grupo de Trabajo, lo ha convocado a los efectos de realizar pruebas de campo que permitan verificar si estas tecnologías permitirían proveer el Servicio Universal de Comunicaciones previsto en la mencionada disposición tanto para ámbitos rurales como para las zonas de baja densidad poblacional que no cuentan con el servicio universal.

A tal efecto actualmente se está trabajando en analizar los equipamientos disponibles en el mercado, sus costos y disponibilidades para comenzar a realizar las mencionadas pruebas de campo. Esta tarea esta ya avanzada y una vez finalizada se adquirirían los equipos disponibles para realizar las pruebas correspondientes.

Similares pruebas se están en estos momentos realizando con los mismos fines en Canadá, Estados Unidos, Reino Unido, Singapur, Uruguay, Sud África, Kenia, India, Filipinas, entre otros países.

6 Conclusiones

Por sus características y prestaciones parecería que la Norma IEEE 802.22 puede resultar adecuada para organizar un sistema de comunicaciones rurales. Esta norma pensada para distancias de hasta 100 km cumpliría largamente las exigencias que impone la dispersión de la población rural alejada de los centros poblados.

Tiene el respaldo de un conjunto de normas exitosas que la preceden y de la cual ella toma numerosas ideas, sobre todo lo que hace al funcionamiento de la capa de enlace, por cuanto toma elementos de la norma 802.3, de amplia difusión y conocimiento.

En consecuencia la utilización de la Norma IEEE 802.22 posibilitará solucionar el problema de comunicación que presentan las pequeñas y olvidadas localidades pequeñas utilizando la moderna infraestructura brindada por la Televisión Digital Terrestre (TDT).

Como se mencionó en el presente trabajo entre las técnicas posibles para optimizar el empleo del ancho de banda utilizado en transmisiones de TDT citaremos la directamente vinculadas a los sistemas de información conocidas como Software Defined Radio (SDR). Esta técnica consiste en que una parte o la totalidad de las funciones de los equipos de comunicaciones radioeléctricas son manejadas por programas informáticos. Este tipo de tecnología abre un importante camino a los Especialistas en Sistemas de Información por la trascendencia que tendrá en el desarrollo de equipamientos de comunicaciones.

A finales de la década de los 90, apareció un nuevo sistema denominado Cognitive Radio, que resulta una variante del sistema antes mencionado y que consiste en un verdadero sistema inteligente que posibilita, entre otros aspectos, manejar comunicaciones radioeléctricas logrando optimizar el espectro de frecuencias utilizado en los sistemas de TV Digital de Radiodifusión, entre otras aplicaciones.

Cabe recordar que en la red de Televisión Digital Terrestre existen estaciones que permiten que las antenas digitales envíen la señal digital a los equipos receptores de los usuarios. Estas estaciones convierten la señal digital en imágenes y sonidos que pueden ser mostradas en cualquier tipo de pantalla, este aspecto es de suma importancia para la implementación de un sistema de comunicaciones rurales.

Paralelamente, la existencia de un sistema de televisión por radiodifusión ya instalado en un gran porcentaje evitaría tener que usar una porción adicional del espectro de frecuencias, que como se ha señalado en este trabajo es cada vez más escaso y congestionado

7 Futuros Trabajos

Se estima que se deberá continuar con el estudio de esta recomendación en sus aspectos técnicos, para determinar fundamentalmente sus limitaciones, si ellas existieran.

Se requerirá un relevamiento del equipamiento que el mercado está ofreciendo sobre esta norma y un análisis de las capacidades del mismo. De esa manera se podrá tener una idea de los costos requeridos para cubrir distintas zonas del territorio nacional.

De la misma manera que se hizo con el Proyecto Corral de Lorca se deberá efectuar una prueba de campo para verificar el verdadero rendimiento del equipamiento y la dificultad que requerirá su despliegue.

Las instalaciones de las estaciones base del sistema de televisión digital terrestre, instaladas sobre shelters, sin duda pueden ser útiles para el despliegue de parte de los equipos requeridos por la 802.22.

8 Referencias

1. Antonio Castro Lechtaler (Director). PICTO 11-18621. Redes Privadas Comunitarias. Proyecto FONCyT, ANPCyT. Working Paper.
2. J. García Guibout, C. García Garino, A. Castro Lechtaler, R. Fusario and Guillermo Sevilla. Physical and Link Layer in Power Line Communications Technologies. Proceedings of 13th of Argentine Congress on Computer Science. ISBN 978 - 950 - 656 - 109 - 3. pp. 56 a 67. Corrientes. October 2007.
3. J. García Guibout, C. García Garino, A. Castro Lechtaler, R. Fusario and Guillermo Sevilla. Power Line Communications in the Electric Network. Proceedings of 13th of Argentine Congress on Computer Science ISBN 978 - 950 - 656 - 109 - 3. pp. 68 a 79. Corrientes. October 2007.
4. J. García Guibout, C. García Garino, A. Castro Lechtaler and R. Fusario. Transmission voice over 802.11. Proceedings of 14th of Argentine Congress on Computer Science. ISBN 978 - 987 - 24611 - 0 - 2. pp. 307 a 318. Chilecito. October 2008.
5. A. Castro Lechtaler, A. Foti, R. Fusario, C. García Garino and J. García Guibout. Communication Access to Small and Remote Communities: The Corral de Lorca Project. Proceedings of 15th of Argentine Congress on Computer Science. ISBN 978 - 897 - 24068 - 4 - 1. pp. 1.117 a 1.126. Jujuy. October 2009.
6. A. Castro Lechtaler, A. Foti, C. García Garino, J. García Guibout, R. Fusario and A. Arroyo Arzubi. Proyecto Corral de Lorca: Una solución de conectividad a grupos poblacionales pequeños, aislados y distantes de centros urbanos. Proceedings de la Novena Conferencia Iberoamericana en Sistemas, Cibernética e Informática: CИСCI 2010. - Volume III - ISBN - 13: 978 - 1 - 934272 - 96 - 1. PP. 121a 127. Orlando, USA. June 2010.
7. <http://www.cplus.org/rmw/index.html> (Radio mobile software).
8. IEEE 802.22 - Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Policies and Procedures for Operation in the TV Bands.
9. Carlos Cordeiro, Kiran Challapali, and Dagnachew Birru, Sai Shankar N. IEEE 802.22: An Introduction to the First Wireless Standard based on Cognitive Radios Journal of Communications, Vol. 1, N° 1, april 2006.
10. http://www.cisco.com/c/es_es/about/press-2013/2013-02-05-trafico-global-de-datos-moviles-se-multiplicara-por-trece-en-2017.html.
11. CEPT Report 24. A preliminary assessment of the feasibility of fitting new/future applications/services into non-harmonized spectrum of the digital dividend (namely the so-called "*white spaces*" between allotments. Report C from CEPT to the European Commission in response to the Mandate on: Technical considerations regarding harmonization options for the Digital Dividend. 1 July 2008.
12. http://www.cse.wustl.edu/~jain/cse574-14/ftp/j_09wsp.pdf
13. http://www.wirelessinnovation.org/introduction_to_sdr
14. Dillinger, M; Madani, K; Alonistioti, N. Software Defined Radio: Architectures, Systems and Functions. Ed. Wiley & Sons, 2003.
15. J. Mitola, G. Maguire. Cognitive radio: making software radios more personal. IEEE Personal Communications Magazine, vol. 6, Nr. 4, pp. 13–18, Aug. 1999.
16. J. Mitola. Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio. Dissertation Submitted in Partial Fulfillment of the Degree of Doctor of Technology. Royal Institute of Technology - KTH Teleinformatics. ISSN 1403 - 5386. Sweden. May 8. 2000.



VIII Workshop Innovación en Sistemas de Software (WISS)

Comité Científico:

Bursztyn Andrés (Argentina)
Capobianco Marcela (Argentina)
Estayno Marcelo (Argentina)
Filottrani Pablo (Argentina)
Flores Andrés (Argentina)
Meles Judith (Argentina)
Montejano Germán (Argentina)
Riesco Daniel (Argentina)
Sposito Osvaldo (Argentina)
Vitturini Mercedes (Argentina)
Vivas Luis (Argentina)
Zanarini Dante (Argentina)



VIII Workshop Innovación en Sistemas de Software (WISS)

ID	Trabajo	Autores	
9111	S.A.A.C., Sistema Automático de Asistencia al Cultivo	Federico Di Luca (UCALP), Luis Di Iasi (UCALP), Javier Marengo (UCALP), Matías Monguillot (UCALP), Juan Negri (UCALP), Rodolfo Bertone (UCALP)	917
9131	InfoUNLP3D: An interactive experience for freshman students	Federico Cristina (UNLP), Sebastián Dapoto (UNLP), Pablo Thomas (UNLP), Patricia Pesado (UNLP)	926
9142	Adquisición Automatizada de Hechos para Sistemas Basados en Conocimiento Empleando Dispositivos Móviles	Matías Pirovano Varela (UM), Gustavo Pereira (UM), Iris Sattolo (UM), Jorge Ierache (UBA)	934
9144	Sistema de Tarjeta Aumentada Basada en Conocimiento para la Asistencia en Emergencia Médica con el empleo de dispositivos móviles	Cristian Montalvo (UNLaM), Facundo Petrolo (UNLaM), Diego Rubén Sanz (UNLaM), Nahuel Adiel Mangiarua (UNLaM), Nicolás Nazareno Verdicchio (UNLaM), Santiago Igarza (UNLaM), Jorge Ierache (UNLaM)	945
9175	Platform of Citizen Interaction: the contribution to the governability of New TIC's and Citizen Relational Management	Juan Doberti (UNLaM), María Laura Pepe (UNLaM), Mónica I. Larrosa (UNLaM), Carla Crocco (UNLaM), Gabriela Gabay (UNLaM), Guillermo Hindi (UNLaM), Carlos Molina (UNLaM), Gonzalo Paradela (UNLaM)	955
9180	Sistema autónomo de video conferencia motorizado, Implementación sobre Raspberry Pi	Daniel Giulianelli (UNLaM), Rocío Andrea Rodríguez (UNLaM), Gabriela Yanina Valles (UNLaM), Mariano Dogliotti (UNLaM)	966
9182	Preliminary results using Snrocket to detect errors in the post-coordination of SNOMED CT.	Pilar Avila (HIBA), Ma. Laura Gambarte (HIBA), Hernan Berinsky (HIBA), Sonia Benítez (HIBA), Hee Park (HIBA), Daniel Luna (HIBA), David Perez (HIBA), Fernando Campos (HIBA), Sofia Zanetti (HIBA), Soledad Diaz (HIBA)	976



ID	Trabajo	Autores		
9196	An Expert-Driven Approach for Collaborative Knowledge Acquisition	Alejandro Leo Lopez Osornio(UNPSJB)(TermMed),	Ordinez Rodrigo Santos (CONICET),	986
		Damián Pablo Barry (UNPSJB),	Sergio Firmenichi (UNLP),	
		Alejandro Sanchez (UNSL)		
9200	Development of a datalogger with open source hardware and software for the study of desert animals	Daniel Díaz (UNSJ),	Sandra Oviedo (UNSJ),	996
		Ramiro Madriaga (UNSJ),	Francisco Ibañez (UNSJ)	
9218	Librería Multiplataforma para Videojuegos en Dispositivos Móviles	Cristian Costa (UNCPBA),	Federico Bricker (UNCPBA),	1006
		Claudio Aciti (UNCPBA)		
9235	Experimentando un uso no tradicional de PhoneGap	Agustina M. Zimbello (UNLP),	Cecilia Challiol (UNLP)	1 016
9264	Procesamiento de sentencias NMEA-0183 para el análisis de la geometría satelital utilizando receptores GPS de bajo costo	Alberto Eduardo Riba (UNdeC),	Jorge Damián Tejada (UNdeC),	1025
		Nelson Acosta (UNCPBA),	Juan Toloza (UNCPBA)	
9298	Reparación eficiente de estructuras de datos en tiempo de ejecución basada en SAT	Marcelo Uva (UNRC),	Pablo Ponzio (UNRC),	1032
		Germán Regis (UNRC),	Nazareno Aguirre (UNRC)	
9321	Principios, Roles y Métricas en alineamiento estratégico de nuevos requerimientos utilizando DevOps	Alberto Belalcázar (UNLP),	F. Javier Díaz (UNLP),	1042
		Lía Molinari (UNLP),	Christian Rodríguez (UNLP)	

S.A.A.C. - Sistema Automático de Asistencia al Cultivo

Di Luca Federico¹, Di Iasi Luis¹, Marengo Javier¹, Monguillot Matías¹,
Negri Juan¹, Bertone Rodolfo¹
fedecthd@hotmail.com,
javimarengo.89@gmail.com,
luisdias@gmail.com,
matiasmonguillot@gmail.com
rodolfo.bertone@ucalp.edu.ar

¹Facultad de Ciencias Exactas e Ingeniería
Universidad Católica de La Plata, UCALP Diagonal 73 esq. 47
1900 La Plata, Buenos Aires, Argentina

Resumen. Este artículo presenta una experiencia realizada en el marco de un proyecto de transferencia desarrollado dentro de la cátedra Proyecto de Sistemas de la carrera Ingeniería en Sistemas de Información de la Universidad Católica Argentina de La Plata.

El proyecto, denominado SAAC, Sistema Automático de Asistencia al Cultivo, describe un modelo experimental con una solución integral capaz de controlar de manera automática diferentes tipos de cultivos desde un dispositivo móvil Android. El resultado obtenido es una combinación eficaz de Hardware, Software e ideas innovadoras, que debido a una parametrización sencilla permite controlar y monitorear la temperatura, humedad, nivel de fertilizante en tierra e iluminación del cultivo, pudiendo modificar los parámetros censados de acuerdo a las necesidades propias definidas para cada tipo de cultivo y/o plantación y la situación climatológica imperante en el entorno.

Keywords: Control Automático, Android, Arduino, Micro Controlador, asistencia al cultivo

Introducción

En la actualidad, en la región del partido de La Plata, se puede percibir un constante incremento y desarrollo de la explotación agrícola y de floricultura en toda su extensión [1]. Este panorama trae arraigado nuevas necesidades, desafíos y profesionalización de la actividad en cuestión. Estas mejoras pueden analizarse desde varios puntos de vista. En particular, desde la perspectiva de la introducción de la Informática o del Control automático puede incorporar desde sistemas de riego automatizados, hasta nuevas tecnologías para la fertilización o adecuación del clima de invernáculo a fin de maximizar los resultados finales.

Para los agricultores o floricultores sería una situación ideal poder controlar su cultivo desde una computadora. Así, el tiempo de riego, la cantidad de agua, la forma en que el agua llega al cultivo, la cantidad de fertilizante inyectado o hasta la temperatura ideal puede controlarse de manera remota y a distancia a fin de maximizar la producción [2]. Esta optimización puede ir desde un pequeño invernáculo instalado en el patio de su hogar hasta una serie de invernáculos para la producción masiva como el control de cientos de hectáreas de cultivo.

El partido de La Plata en particular, en conjunto con los partidos de Ensenada y Berisso, todos en la provincia de Buenos Aires, han visto un incremento de la actividad agrícola y de floricultura desde la década del 80 del siglo pasado. En particular, desde comienzo del siglo XXI, el sector dedicados a las hortalizas [3] ha crecido en demanda enviando los productos generados a centros como Rosario, Córdoba, Mar del Plata y Bahía Blanca. Esto ha llevado al crecimiento del sistema productivo, mediante la instalación de nuevas granjas con mayor cantidad de invernáculos.

La zona platense no sólo está dedicada a la producción hortícola. El mercado de las flores [4] tiene un aumento de la producción notorio, tanto en la cantidad de nuevos emprendimientos, como en la ampliación de los existentes.

Sin embargo este crecimiento del mercado y de la producción no tiene un correlato evidente en la incorporación de la tecnología para monitorear y mejorar el comportamiento de los cultivos. Los nuevos invernáculos cuentan con riego integral, pero no en todos los casos el mismo está automatizado. La mayor automatización impuesta tiene que ver con el control de riego por tiempo mediante temporizadores. De esta forma, es posible lograr un control mínimo de humedad de las plantas, pero asociado a parámetros muy generales que no analizan de una forma la temperatura exterior ni las características del tipo cultivo.

La automatización en el control de cultivos es una actividad simple y económica de desarrollar. Se cuentan con dispositivos que permiten monitorear el medio ambiente y, teniendo asociado las necesidades de cada cultivo es posible controlar mejor los recursos como agua, fertilizante o calor.

Motivación

Teniendo en cuenta este contexto, desde la asignatura Proyecto de Sistemas de quinto año de la Ingeniería en Sistemas de Información, y como parte del trabajo final de la materia, se puso en práctica la iniciativa de generar un prototipo experimental con una solución sistémica-automatizada que proporcione una serie de respuestas controladas a los factores como humedad, calor y fertilización que puede necesitar productor frutihortícola en un contexto de uno o varios invernáculos.

De esta forma, se propuso el desarrollo una herramienta que se denominó SAAC (Sistema Automático de Asistencia al Cultivo). Este producto tuvo por objetivo generar en forma experimental, un entorno que permitiera controlar de forma segura y confiable parámetros tales como temperatura ambiente, humedad del suelo, humedad ambiente, luminosidad y fertilizantes de una vasta gama de cultivos. Además tenía como requerimiento base, permitir una configuración y parametrización amplia para responder a las necesidades específicas de cada tipo de cultivo.

El concepto a i mplantar con el desarrollo de SAAC es la gestión de plantaciones (hortícolas o de flores). Cada una de ellas con características y necesidades propias y parametrizables

Para desarrollar este modelo experimental se utilizó una plataforma de hardware libre denominada Arduino, con la cual se han controlado todos los módulos (Relé, DHT-22, HL-69, HC-06), y se desarrolló de una aplicación Android, desde donde se controla al hardware de acuerdo a los parámetros establecidos.

Arduino es una plataforma de código abierto basado en hardware y software de utilización simple. Una placa Arduino permite leer entradas, como por ejemplo la luz de un sensor, la activación de un botón o un estado cualquiera y convertirlo en una salida que produzca un efecto [5]. Para programar el microcontrolador de la placa, se utiliza el lenguaje de programación de Arduino (Wiring [6]), y el software de Arduino (IDE), basado en Processing [7] .

Arduino presenta una serie de características que lo hacen viable no solo para este proyecto experimental, sino que además pueden ser implantados en proyectos reales de mediana o gran envergadura. Algunas de estas características son [8] [9]:

- Costo. Las placas Arduino son relativamente económicas en comparación con otras plataformas de microcontroladores. Es posible ensamblar manualmente una versión económica por menos de us\$50.
- multiplataforma - El software de Arduino (IDE) se ejecuta en Windows, Macintosh OS X, y Linux. La mayoría de los sistemas de microcontrolador se limitan a Windows.

- Simple con entorno de programación clara - El software de Arduino (IDE) es fácil de usar, pero lo suficientemente flexible para los usuarios avanzados puedan generar aplicaciones complejas.
- Código abierto y el software extensible - El software de Arduino se publica como herramientas de código abierto, disponible para la extensión por los programadores experimentados.
- Código abierto y hardware ampliable - Los diseñadores de circuitos pueden hacer su propia versión del módulo, ampliándolo y mejorándolo, con más hardware.

Desarrollo realizado

Se confeccionó una maqueta para la simulación del proyecto que consiste en dos partes: una de hardware y otra de software.

Para la parte de hardware se armó una maqueta con acrílico dentro de la cual se colocaron a modo experimental macetas con plantas. Las macetas fueron rodeadas con mangueras que transportan agua y fertilizantes. En uno de los frentes de la maqueta se adhirió una caja con la placa controladora, y luego por cables y de manera externa otras dos cajas donde se realizó la conexión del Protoboard para conectar el módulo bluetooth, el módulo micro SD, la fuente de alimentación y los rele. La figura 1. presenta una foto del hardware.



Figura 1. Maqueta que simula el invernáculo

Se describen a continuación los componentes de hardware utilizados para el control automático de la maqueta de invernáculo construida:

- Arduino mega 2560, se seleccionó este componente debido a que puede considerarse el microcontrolador más capaz de la familia Arduino, además

posee una conexión USB que sirve como alimentación y como entrada para la programación, sumado a que tiene un bajo costo.

- Fuente de alimentación, fue diseñada y ensamblada con la finalidad de entregar 3 voltajes distintos: 5V, 9V y 12V, alimentando la placa de Arduino Mega 2560. El desarrollo de esta fuente de alimentación fue parte de otro trabajo de cátedra. De esta forma pudo ser construida reduciendo los costos de la misma
- Módulos Relé optoacoplador, fueron utilizados para activar/desactivar mediante una señal eléctrica los diferentes componentes del sistema, en este caso se utilizaron para las bombas de agua y fertilizante y para el encendido/apagado de la lámpara y los coolers refrigeradores.
- Módulo DHT-22, el cual tiene como finalidad obtener e informar la temperatura y humedad ambiente. Dicho elemento, está especialmente diseñado para interactuar con cualquier placa de micro-control de la familia Arduino. El módulo está compuesto por un integrado llamado AM2302, que es el encargado de llevar a cabo la tarea. Si bien este módulo puede sensar cada dos segundos como mínimo, en el caso del modelo realizado no generó problema. El rango de sensado tanto para temperatura como para humedad fue aceptable para el problema analizado.
- Módulo HL-69, tiene como finalidad obtener e informar la humedad del suelo. Nuevamente, este módulo interactúa con cualquier placa de microcontrol de la familia Arduino. Está compuesto por 2 partes: el sensor de humedad propiamente dicho (IL-69) y una placa electrónica de acondicionamiento de señal (IL-38).
- Módulo bluetooth HC-05, un módulo muy popular para aplicaciones con microcontroladores Arduino. Es un dispositivo de bajo costo y que se pueden adosar a cualquier microcontrolador. Es un módulo Maestro-Esclavo, quiere decir que además de recibir conexiones desde una PC o tablet, también es capaz de generar conexiones hacia otros dispositivos bluetooth, en particular teléfonos smart.
- Módulo adaptador micro SD card XX. Debido a que el arduino por sí solo no puede persistir los datos censados, se incorporó este módulo para que cada instantes de tiempo predeterminados se almacene información obtenida. Cuando el dispositivo móvil se conecta con el arduino, debe leer de la tarjeta de memoria para actualizar la base de datos.
- Timer para controlar el tiempo de encendido y apagado de la lámpara, desde el dispositivo móvil se parametriza el tiempo en segundos de cuanto debe estar prendida y/o apagada la lámpara.
- Por último tres elementos accesorios que fueron parte de invernáculo experimental
 - Lámpara halógena, para asegurar el aumento de la temperatura en caso de ser necesario, y en cierta forma emular los efectos de la radiación solar.

- Coolers de PC, para bajar la temperatura en caso de ser necesario.
- Bombas de pecera son controladas por el módulo Relé para el encendido y apagado, que permiten activar el riego y la fertilización.

Para la parte de software se trabajó sobre el IDE proporcionado por el arduino para la programación de la lectura de los sensores. En tanto que para la programación de la interfaz de usuario se utilizó el IDE android studio 1.4. Esta herramienta permite generar, probar y firmar aplicaciones, las cuales se pueden publicar para su descarga y utilización. Esta versión permite realizar un monitoreo más exhaustivo de recursos, ya sea CPU, GPU Y de red y, además, incorpora facilidades para interactuar con bases de datos.

La versión de Android utilizada para realizar el software y los modelos de simulación consecuentes fue Jelly Bean 4.1 o posterior, debido a que se contaba con esas versiones en los dispositivos móviles disponibles. La comunicación entre el dispositivo móvil y el Arduino se estableció vía bluetooth mediante el protocolo I2C proporcionado por el Arduino.

A partir de la maqueta se generó la simulación del sistema. Para ello se parametrizó desde el dispositivo móvil, los valores mínimos y máximos de cada uno de los sensores. Posteriormente, el sistema lee cada un Δ de tiempo (medido en segundos y parametrizable también), los valores arrojados por los sensores. Estos se comparan con los valores de referencia definidos en la Base de Datos para el cultivo (en el caso del experimento valores adecuados para las flores) y reaccionar según corresponda.

Se diseñó la aplicación para que en el dispositivo móvil se actualice en tiempo real la información obtenida desde los sensores. Asimismo, la información sensada es almacenada en una BD asociada para tener la trazabilidad de los valores obtenidos a lo largo del tiempo..

La Aplicación SACC.

La figura 2 presenta otra visión de la maqueta desarrollada. Se pueden observar los diferentes módulos Arduino y su interconexión con los dispositivos de riego y fertilización en la parte superior de la maqueta.

La aplicación desarrollada tiene varios módulos o componentes: seguridad, parametrización y monitoreo y toma de decisiones. La figura 3 muestra algunas capturas de la interfaz de usuario de la aplicación realizada.

El módulo de seguridad permite definir usuarios y el comportamiento que cada uno de ellos tendrá con el sistema y/o cultivo. De acuerdo a su perfil un usuario puede solo ver el historial del cultivo, modificar algún parámetro específico, ingresar nuevos tipos de cultivos o asociar (en un futuro) un cultivo a un área censada (actualmente solo es experimental y se dispone de una sola maqueta).

El módulo de parametrización que permite definir nuevos cultivos y sus características. El monitoreo de los sensores se realiza, como se indicó anteriormente, cada un Δ de tiempo preestablecido. A partir de los valores obtenidos, y luego de persistirlos en la Base de Datos, los mismos son comparados contra los parámetros del cultivo para tomar las decisiones adecuadas.

Una de las pruebas realizadas consistió en generar un cultivo de Pepinillos en las macetas de la maqueta. Alguno de los parámetros de simulación realizados fueron:

Cantidad de Luz: 5 segundos mínimo - 10 segundos de máximo

Temperatura: 22 grados mínimo - 29 grados máximo

Humedad Ambiente: 22 mínimo - 40 máximo

Fertilizante: 5 segundos mínimo - 10 segundos máximo

Humedad de suelo: 400 mínimo - 700 máximo

Se pudo comprobar que la aplicación controló y mantuvo dentro de esos valores cada uno de los parámetros durante una simulación realizada durante 96 hs.



Figura 2. Vista Trasera de la maqueta.

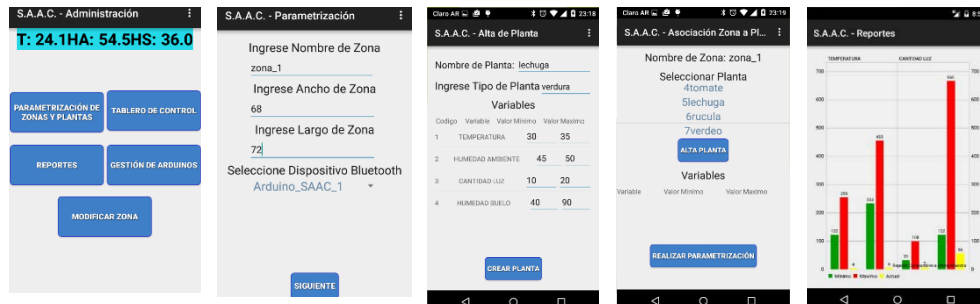


Figura 3. Vistas de SACC.

Conclusiones

S.A.A.C proporciona a los usuarios una herramienta capaz de obtener una mejor cosecha del cultivo. Para lograrlo administra de forma más eficiente los recursos relacionados con el problema. Estos recursos son: Agua, fertilizante y energía, dado que se controla la temperatura y humedad ambiente mediante lámparas y ventiladores.

Se diseñó una herramienta que independiza al usuario del mantenimiento del cultivo, ya que con sus sensores se puede abastecer al suelo y al ambiente de los componentes necesarios.

Desde el punto de vista de la aplicación, S.A.A.C le proporciona al usuario una vista general de todas las variables, siendo capaz de mostrar en tiempo real el valor de las mismas. Así también, permite obtener un historial de resultados para un parámetro determinado por el rango de tiempo deseado.

Las posibilidades de implantación en un entorno real son concretas. Se está trabajando con una cooperativa de la zona para implantar el prototipo dentro de algunos de sus invernáculos.

La ventaja más notoria que presenta el producto desarrollado es su economía. Cada componente Arduino tiene un costo bajo y, si bien en un invernáculo real es necesario plantar más cantidad de sensores, esto no encarece en demasía el problema dado que se logra aumentar la producción y reducir el costo en agua y fertilizante. Ahora, se agrega un factor importante como el uso más racional de la energía.

El proyecto fue desarrollado en el ámbito de una asignatura de la Carrera de Ingeniería en Sistemas de Información. Se alcanzaron todas las metas previstas pero además se contempló que en cohortes posteriores de alumnos, se generan mejoras en el producto. Se

detallan a continuación algunas de las mejoras previstas y que están siendo analizadas por la cohorte 2016 de la materia.

Implementación del módulo Wi-fi: Se puede implementar un módulo wi-fi en lugar del módulo de bluetooth para el envío y recepción de datos. De esta manera, el usuario podrá consultar el estado de las variables en cualquier momento.

Medición de altura de las plantas: Luego de una investigación realizada se llegó a la conclusión que para medir la altura de las plantas lo más recomendable es usar el sensor de distancia HC-SR04 que funciona mediante ultrasonido. La altura de la planta agrega el crecimiento como otro factor más para determinar la utilización de fertilizantes y agua.

- Cámara de fotos: Otra forma de medir las plantas, sería utilizando una cámara de fotos. Para un buen uso de este método se debe colocar una regla que quede de manera visible dentro de la maqueta. La cámara de fotos tomaría capturas cada determinado tiempo. De esta manera el usuario puede ir monitoreando el crecimiento del cultivo y su estado.
- Advertencia sobre comportamiento inusual de las variables: otra mejora posible es la implementación de un módulo en la aplicación que le avise al usuario sobre el comportamiento inusual de alguna/s de las variables. Ya sea porque el arduino se quedó sin batería o bien debido a que alguno de los sensores haya dejado de funcionar.

Bibliografía

1. <http://www.senasa.gov.ar/senasa-comunica/noticias/gran-la-plata-impulso-los-productores-en-la-inocuidad-de-los-cultivos-hortícolas>
2. [http://inta.gob.ar/sites/default/files/script-tmp-manual de cultivos para la huerta organica familiar -.pdf](http://inta.gob.ar/sites/default/files/script-tmp-manual_de_cultivos_para_la_huerta_organica_familiar_.pdf)
3. http://www.agro.unlp.edu.ar/sites/default/files/paginas/2008._manual_cinturon_horticola_la_plata_directora_mabel_vazquez.pdf
4. http://aulavirtual.agro.unlp.edu.ar/pluginfile.php/16537/mod_resource/content/4/GUIA%20FLORICULTURA%20ARGENTINA%202013.pdf
5. <http://www.arduino.cc>
6. <http://wiring.org.co>
7. <https://processing.org>
8. [Arduino: a comprehensive starting up guide for complete beginners Kindle Edition](#)
9. [Exploring Arduino: Tools and Techniques for Engineering Wizardry 1st Edition ISBN-13: 978-1118549360](#)

InfoUNLP3D: An interactive experience for freshman students

Federico Cristina¹, Sebastián Dapoto¹, Pablo Thomas¹, Patricia Pesado^{1,2}

¹ Instituto de Investigación en Informática LIDI
Universidad Nacional de La Plata – Argentina

² Comisión de Investigaciones Científicas de la Provincia de Buenos Aires - Argentina

{fcristina, sdapoto, pthomas, ppesado}@lidi.info.unlp.edu.ar

Abstract. The use of technology in educational environments is becoming a common standard. Current high-end mobile devices provide the necessary hardware to allow the execution of complex and resource-demanding 3D applications, at least in their simplest forms under low-end devices. Under this context, a multiplatform 3D mobile application is proposed, presenting a complete virtual scenario of the Faculty of Informatics (Universidad Nacional de La Plata, Argentina), with the main purpose of being an interactive guide for students. The application provides a navigational model of the building, several playing modes in order to get to know the facilities, and updated information about classrooms and courses.

Keywords: virtual buildings, 3D navigation, mobile devices

1 Introduction

Nowadays, the use of technology for helping purposes in everyday situations is something increasingly common. In this context, mobile devices such as cell phones, smart phones or tablets increase the possibilities of creating new technology to assist people under certain circumstances. This technology has the necessary potential to be also used in learning activities [1, 2, 3].

Additionally, the computational capacity of these devices have increased considerably in the past few years, allowing the execution of 3D applications with high levels of hardware requirements with ease. By doing this, it is now possible to create tridimensional experiences in which the user can even achieve a complete immersion within a virtual environment.

Thus, these two concepts (learning software and virtual 3D experiences) can be related in order to create a vast number of applications that aid users in their lives through easy-to-use and visually attractive solutions. One of these applications is the one proposed in the present paper. Its main goal is to visually help students with their first steps at the faculty to locate classrooms and place themselves within the building.

The rest of the paper is organized as follows. Section 2 presents the motivations for the proposed work. Sections 3 to 5 describe the main stages in the creation of the interactive experience. Section 6 presents the results achieved after its publication and use by the students. Section 7 presents a list of possible related work.

2 Motivation

For freshman students, the initial days within the faculty can be a challenging and stressful experience. For the most part, a new and more independent way of life begins; all this changes might overwhelm them [4, 5].

In view of this situation, and despite the fact that the building provides all corresponding signaling and information offices, this is sometimes not enough for freshman students, who, in some cases, present interaction difficulties in their first days at the university.

Nowadays, most of the students own mobile devices. This is even more noticeable in computer science careers. They are familiarized with this kind of technology and easily master a large variety of mobile applications, which are a fundamental part of their daily life.

Therefore, a mobile application that assists students in their first stages at the university turns out to be very useful, helping them to become familiar with the building, its main areas and offices.

The proposed application consists of a mobile interactive 3D virtual model of the Faculty of Informatics, Universidad Nacional de La Plata, Argentina. Users can navigate the building or take virtual tours all around its facilities, in which several signs with the most relevant points of interest are displayed in an *augmented information style*. Additionally, information about classrooms and courses is shown while navigating the virtual scenario.

3 From blueprints to 3D building model

The first stage in the development of the interactive solution was the creation of a 3D model of the whole building. This task implied constructing the mesh based on the original faculty blueprints.

The faculty is a 3 stories building, containing dozens of classrooms, administrative offices, bathrooms, elevators, a library, a cafeteria, a green area and a parking lot; all of which had to be reproduced in the mesh.

In addition to blueprints, web mapping services like Google Maps [6] and Bing Maps [7] were used as reference in order to consider additional visual information not included in the blueprints such as roof color, trees placed inside the green area, building orientation, etc.

During the design, a special consideration was given to possible performance issues if the application would be run on slow mobile devices. Key aspects like the number of objects in the scene, number of polygons per object, or the textures used

for the materials were taken into account with the goal of a correct balance between quality and performance.

Figure 1 shows an early stage of the modeling process. The blueprint is used as a reference in order to create the walls, floor, etc. The corresponding constructed 3D model is shown at figure 2.

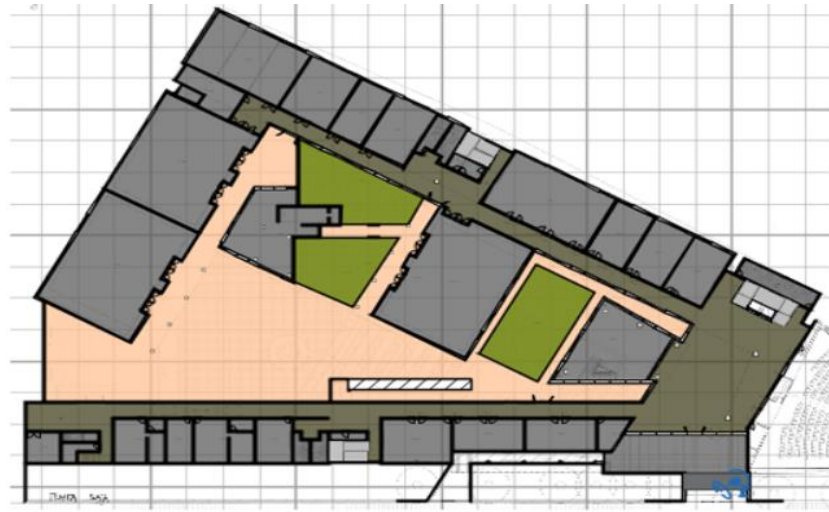


Fig. 1. Blueprint used as reference.



Fig. 2. The constructed 3D model.

4 Interactive experience

The main goal of the experience is to allow students to become acquainted with the different areas of the faculty, and how to get to them.

This is achieved in several ways or playing modes available in the application:

- *General tour*: this option automatically navigates the whole building; starting from the main entrance, the tour presents a complete walkthrough through the hallways for each story, ending in a general bird's-eye view of the complex.
- *How to get to...:* in order to easily locate a bathroom, the cafeteria, etc. several preloaded navigations were included so that the student simply selects the location of interest, and the tour will explain how to get there from the main entrance.
- *Free roam*: the student will be able to navigate any area of the faculty in a first person perspective, virtually *walking* through the building hallways, stairs, etc.

These modes - in more extent the free roam one - requires an application with full user interaction support.

To achieve this goal, the application was created using Unity [8], a 3D development framework for creating interactive solutions, games, etc. Being a 3D solution, Unity supports the use of 3D meshes [9,10, 11, 12]. In particular, the model previously created in Sweet Home 3D [13] was exported and imported in Unity.

Once imported, a series of activities were necessary to adapt the 3D object for its correct use in Unity. The most important ones were the scale correction of the 3D model for proper handling and a mesh collider application on its members in order to run the simulation as expected in the free roam mode.

Together with the model, visual text hints were created as quick references for each place of interest, such as the library, bathrooms, etc. These hints are always displayed on foreground so that students can recognize these places along the use of the application, wherever they are placed.

The free roam mode required a special user interface, which allows the user to control the virtual character to move throughout the building. The mobile versions of the application present two virtual controls in order to move and change its orientation, while the desktop versions control the character by using keyboard and mouse.

Figure 3 shows the model already imported and adapted for Unity. Individual animations were created for each tour, setting camera position and rotation for each keyframe along time, as shown at the bottom of the image.

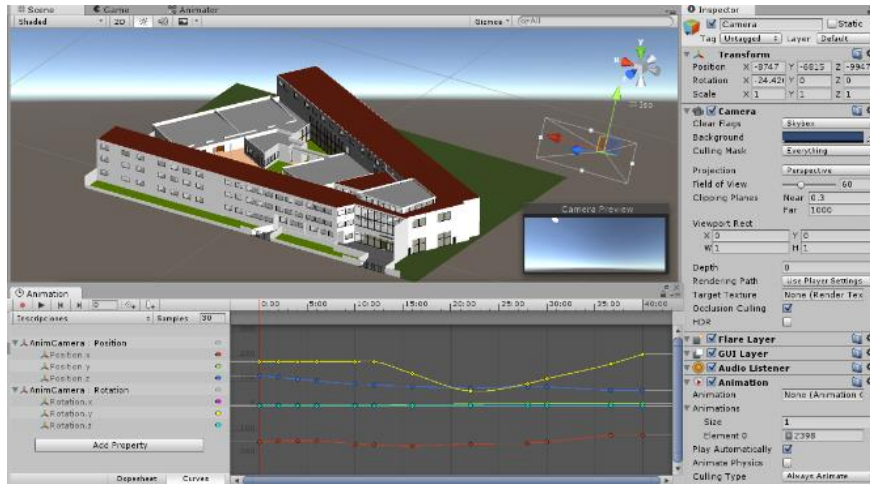


Fig. 3. The 3D model in Unity.

5 Integration with the classrooms information API

The *free roam* mode allows a useful additional feature. Every time the user is near a classroom, the daily schedule of courses for that classroom is displayed with information such as course name, professor in charge, type of course, hour of the day, duration, etc. The user can also browse the schedule through different days of the week, or even search courses for another quarter.

To achieve this functionality, the application interacts with an external system, also developed at the faculty: *Teachers Management System* [14], in particular with the *Classrooms Information API*. This API allows to query for a particular classroom information through a REST call [15].

The complete schedule for every classroom is prefetched and processed at the application startup time, so that all the information is already loaded by the time the user starts using the application. This eradicates any possibility of display lag when the user approaches a classroom.

Figure 4 shows an example of this feature. Every time the user moves towards a different classroom, the information will change accordingly.

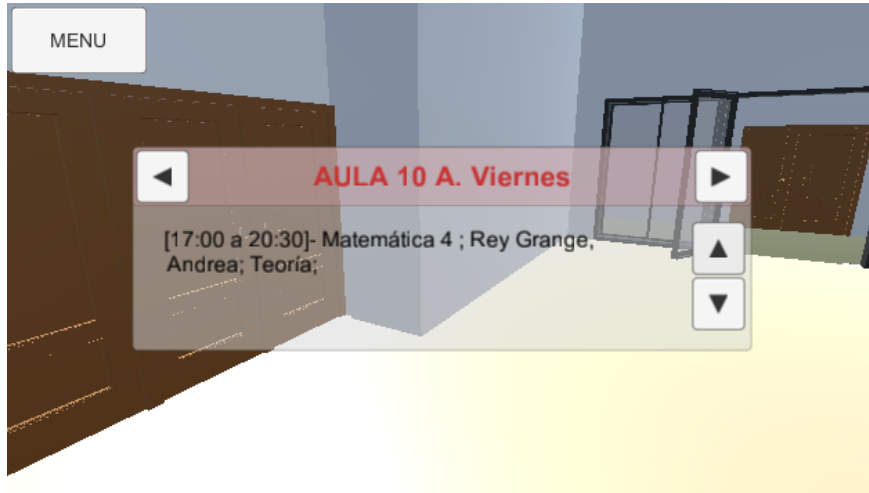


Fig. 4. Detailed information about the courses schedule is displayed while the user walks near the classroom.

6 Results

Once the development stage was completed, the resulting application was released for distribution. The binaries are available in the *institutional* section of the faculty's official website [16].

The application was released under different builds for several platforms:

- An Android version for mobile devices such as cell phones or tablets.
- A Windows version for traditional desktop/notebook devices
- A Web version for any other type of device/operating system.

The main difference between the native versions (Android/Window) in contrast to the Web one is that the achieved performance is better in the former ones. New builds are planned to be created, such as distributions for iOS and Linux in order to include the whole range of users using native versions.

Students were notified about the application through the use of social networks like the faculty page at Facebook (around 3000 followers). A positive feedback was received by users, which *shared* and *liked* the publication [17].

The application's *General Tour* mode was even used at the inauguration of a new area of the faculty building [18, 19].

Figure 5 depicts the final application running on an LG device with Android OS. The different *playing modes* can be selected from the menu on the left, or any of the predefined destinations in the *How to get to...* mode. Virtual controls are displayed along the screen for moving and rotating the camera in the *free roam* mode.

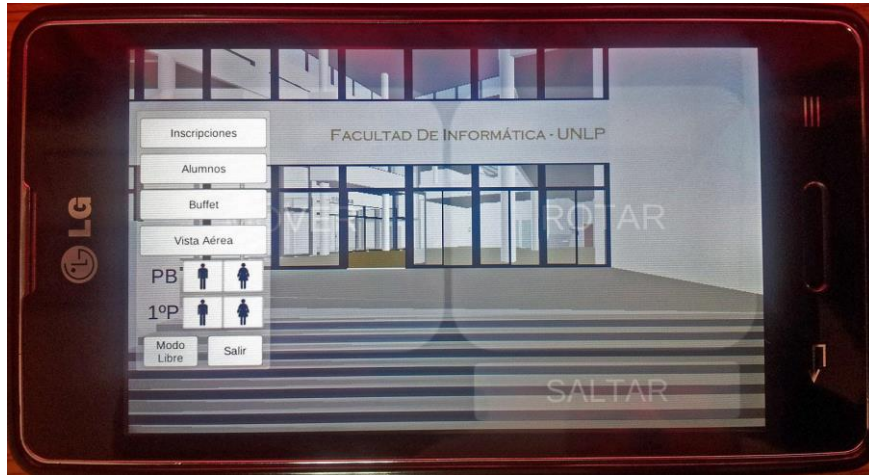


Fig. 5. The final application running on an LG device with Android OS.

7 Conclusions and future work

This paper presented a novel application that aids students to have a better knowledge of the faculty where they are starting their studies, through a virtual experience which shows the highlights of the building.

Several playing modes are available in order to enrich the experience, including a free-roam mode that allows a completely free navigation of the installations.

The application is available for several platforms, spanning a wide range of desktop and mobiles devices. There is also a web-based version in order to extend its use to any type of device.

The application was well received by students and general users, who gave positive feedback about it and even suggested new features, which now are the main objectives planned for future work.

First, a mid-term goal is to expand the interactive experience increasing the augmented information displayed in the application, which will be useful not only for students but also for teachers. For instance, when a user is positioned in front of a room, the system will provide related information such as:

- Room capacity
- Room equipment (computer, projector, etc.)
- Number of students

Second, a long-term goal is to extend the virtual scenario with new models from other faculties, so that the whole university will be included in the experience. In this way, the application will not be targeted only for informatics, but for a wider range of students from any other area.

References

1. Cristina, F.; Dapoto, S.; Thomas, P.; Pesado, P. "A simplified multiplatform communication framework for mobile applications". IEEE International Conference on Computer Engineering & Systems (ICCES). December 2014. ISBN 978-1-4799-6593-9.
2. Cristina, F.; Dapoto, S.; Thomas, P.; Pesado, P. "Prototipo móvil 3D para el aprendizaje de algoritmos básicos". XXI Congreso Argentino de Ciencias de la Computación CACIC. October 2015.
3. Goldin C.; Katz L. "The Race between Education and Technology". 2010. ISBN-13: 978-0674035300
4. Caballero Z.; Gómez M.; Borgobello A.; Ciarla D. "Una experiencia de taller para alumnos pre-ingresantes". 2004. Facultad de Psicología. UNR.
5. Romero H. "Características de los ingresantes a la universidad". V Encuentro Nacional y II Latinoamericano La Universidad como objeto de investigación. 2007. ISBN 978-950-658-187-9.
6. Google Maps. <http://google.com/maps>
7. Bing Maps. <http://bing.com/maps>
8. Unity online manual. <http://docs.unity3d.com/Manual/index.html>
9. Nystrom R. "Game Programming Patterns". 2014. ISBN-13: 987-0990582908
10. Hocking J. "Unity in Action: Multiplatform Game Development in C# with Unity 5. 1st Edition". 2015. ISBN-13: 978-1617292323
11. Smith M. "Unity 5.x Cookbook". 2015. ISBN-13: 978-1784391362
12. Linowes J. "Unity Virtual Reality Projects". 2015. ISBN-13: 978-1783988556
13. Sweet Home 3D user's guide. <http://www.sweethome3d.com/userGuide.jsp>
14. Sistema de Gestión Docente. Facultad de Informática. Universidad Nacional de La Plata. <http://gestiondocente.info.unlp.edu.ar/>
15. Leonard Richardson, Sam Ruby. "RESTful Web Services". O'Reilly. 2007. ISBN: 978-0-596-52926-0
16. Virtual tour access links at UNLP official site. November 2015. <http://www.info.unlp.edu.ar/index.php/institucional/recorrido-virtual>
17. UNLP official Facebook Page. Virtual Tour news publication. November 2015. <https://www.facebook.com/InfoUNLP/posts/946207192138632>
18. "Avanzan las obras de ampliación del edificio de Informática". YouTube video. December 2015. <https://www.youtube.com/watch?v=9DSdWxuswJk>
19. "Noticias UNLP - Informática empezará 2016 estrenando instalaciones". YouTube video. December 2015. <https://www.youtube.com/watch?v=m7esG8Z63sk>

Adquisición Automatizada de Hechos para Sistemas Basados en Conocimiento Empleando Dispositivos Móviles

Matias Pirovano Varela, Gustavo Pereira, Iris Sattolo, Jorge Salvador Ierache

Instituto de Sistemas Inteligentes y Enseñanza Experimental de la Robótica (ISIER)
Facultad de Informática, Ciencias de la Comunicación y Técnicas Especiales
Universidad de Morón, Cabildo 134 (B1708JPD), Morón, Buenos Aires, Argentina
jierache@unimoron.edu.ar

Abstract. Los sistemas basados en conocimiento existen desde hace varias décadas, y han sido utilizados durante ese periodo extensivamente en el desarrollo de sistemas expertos, pero recién con la aparición de los dispositivos móviles modernos se ha conseguido una renovada apreciación de ellos. Los dispositivos móviles permiten tener un experto en el tema que se necesite, cuando y donde se lo requiera. Este trabajo propone una arquitectura para la integración de sistemas basados en conocimiento integrados en dispositivos móviles que permita la adquisición de los datos que representan los hechos desde la nube minimizando la carga de datos por parte del usuario.

Keywords: Sistema Basado en Reglas, Dispositivo Móvil, Android, Sistemas Inteligentes.

1 Introducción

Este trabajo propone una arquitectura para una plataforma móvil que permita la ejecución de un Sistema Basado en Conocimiento (SBC) con la mínima intervención del usuario. Si bien actualmente existen SBC que corren en forma nativa sobre dispositivos móviles, estos requieren generalmente que el usuario ingrese datos para continuar con la ejecución del sistema. Estos no cuentan con una ejecución automatizada, con la mínima intervención del usuario en el proceso de carga de datos, es decir, que el sistema se encargue de buscar los datos sin solicitárselos al usuario, tanto en la nube u otros dispositivos asociados, como sensores o artefactos en el contexto del Internet de las cosas. En este orden se desarrolló un marco de trabajo que funcione sobre plataformas móviles, con el cual se pueda configurar la ejecución de un SBC, en lugar de requerir que un usuario ingrese la mayoría o la totalidad de los datos en la nube, el proceso de ejecución debe ser capaz de buscar la información necesaria desde otras fuentes de datos, reduciendo la carga de atención del usuario para el ingreso de los mismos.

1.1 Sistemas Basados en Conocimiento

Un SBC facilita la toma de decisiones a usuarios no expertos, utilizando el conocimiento de un experto o especialista humano [1]. Los SBC están diseñados para resolver problemas complejos utilizando reglas generadas de acuerdo al conocimiento extraído y educido del experto [2].

La arquitectura de un SBC puede dividirse en dos partes: a) una dependiente de la problemática a resolver, y b) otra independiente, formada por el Motor de Inferencia (MI). La parte dependiente es la Base de Conocimiento (BC) definida como la unión del conjunto de aserciones y reglas en los que se plasmó el conocimiento del experto. La BC está conformada por la Base de Hechos (BH) y la Base de Reglas (BR).

La BH tiene como función proveer al MI los datos sobre el problema, lo que le permite activar las reglas declaradas en la BR, mostrar los resultados obtenidos y generar una explicación de los pasos tomados.

1.2 Dispositivos Móviles

Los dispositivos móviles (celulares inteligentes, tablets, y similares) que han surgido en los últimos años tienen en su mayoría muy buena capacidad de procesamiento [3], lo que facilita la implantación de SBC en estos dispositivos.

La ventaja principal de poder ejecutar un SBC sobre un dispositivo móvil es justamente la movilidad del dispositivo. Esto genera la posibilidad de tener un experto siempre al alcance de la mano.

Para el desarrollo de SBC sobre dispositivos móviles, se debe tener en cuenta el sistema operativo, actualmente el mercado está concentrado en dos sistemas operativos, Android con el 80% e iOS con el 17% [4]. Programar para Android requiere un menor costo operativo, se posee una mayor penetración del mercado y permite una mayor variedad de desarrollos e integraciones entre aplicaciones.

De los varios SBC que existen, hay uno que se destaca sobre el resto, e2gDroid para Android [5]. e2gDroid es el único que posee un MI independiente y está preparado para ejecutar cualquier BC que el usuario le provea.

1.3 Trabajos Previos

La integración entre los dispositivos móviles y los SBC comenzaron hace años con un sistema funcionando en un servidor externo y utilizando el dispositivo únicamente como pantalla para mostrar resultados [6], luego, al incrementar la capacidad de procesamiento de los dispositivos móviles, fue posible crear aplicaciones nativas.

Algunos trabajos han tratado la implementación de SBC en el área pedagógica, con el fin de expandir la actual educación a distancia [7]; la integración con la medicina y el procesamiento móvil [8], [9]; o creando un sistema de asistencia al conductor de vehículos (Artemisa) [10]. Otros trabajos han dejado abiertas líneas de investigación referidas a la integración de SBC con el modelado basado en agentes, combinándolos mediante una arquitectura llamada ARTIMIS [11]; mientras que otros definieron un marco de trabajo genérico que plantea una solución para la comunicación entre los

agentes y sus usuarios humanos [12]. Otro modelo de agentes presentado es ANDROMEDA, una plataforma orientada a embeber agentes en el sistema operativo Android [13]. También se deben tener en cuenta a Drools, librería que permite implementar un MI [14], [15]; y d3web, un SBC web [16].

2 Comparación de Sistemas Basados en Conocimiento Existentes

Se investigaron las distintas soluciones existentes para dispositivos móviles presentándose en la tabla 1 una síntesis comparativa de las características deseadas, enunciadas en la misma, que muestra la situación actual con respecto al desarrollo de un SBC en dispositivos móviles.

Tabla 1. Comparación de soluciones de SBC en dispositivos móviles.

Características de aplicaciones	e2gDroid	Artemisa	Drools	d3web
Compatibilidad de Sistema Operativo	Android	Android	Android	Android / iOS
Corre nativo al SO	Sí	Sí	Sí	No
Requiere desarrollo para ser utilizado	No	No	Sí ¹	Sí ²
Requiere conexión activa con Internet	No	Sí	No	Sí
Aplicación independiente de la base de conocimiento	Sí	No	No	Sí
Permite captura de datos automatizada	No	Sí	No	No

Del análisis presentado se desprende como síntesis que ninguna opción es una solución completa a la problemática que se investiga, en términos de alcanzar una solución que brinde una ejecución de la adquisición de datos automatizada, siendo las detalladas sólo soluciones parciales.

e2gDroid es el único que demuestra la posibilidad de tener un sistema basado en conocimiento funcionando en un dispositivo móvil y a la vez ser independiente de la base de conocimiento que se ejecuta, pero no permite la captura de datos automatizada.

3 Solución Propuesta

El primer punto planteado en el desarrollo, fue la selección de Android como sistema operativo. Si bien iOS posee un ambiente más controlado, una menor cantidad de errores en el sistema operativo y una menor fragmentación de versiones, Android es una solución más económica, posee un mayor mercado local y permite un mayor número de desarrollos posibles.

Se tomó como punto de inicio para el desarrollo la flexibilidad y madurez de e2gDroid. La arquitectura del SBC funciona de forma local en el dispositivo móvil,

¹ Dado que Drools es una librería, es necesario programar una aplicación que las emplee.

² d3web requiere la instalación y configuración del servidor web donde va a estar alojado.

pero también puede actualizar las BC desde un repositorio central remoto, simplificando el mantenimiento de las mismas. También es capaz adquirir las respuestas por sí mismo, es decir, además de esperar que el usuario ingrese los valores para continuar con la ejecución, la aplicación obtiene los datos consultando fuentes externas. Estas fuentes pueden ser direcciones web o dispositivos a los cuales se comunicará.

3.1 Arquitectura

Luego de relevar las arquitecturas existentes [17], se optó por utilizar una arquitectura híbrida. En esta arquitectura el proceso se ejecuta sobre el dispositivo móvil, conservando conexiones a servidores y dispositivos externos. Estas conexiones permiten la actualización y adquisición de la BC y de la Base de Automatización de captura de datos, o simplemente Base de Automatización, (BA). Esta nueva base contiene la descripción de a dónde ir a encontrar el dato que alimenta a la BH de la BC. En la figura 1 se detalla la arquitectura propuesta.

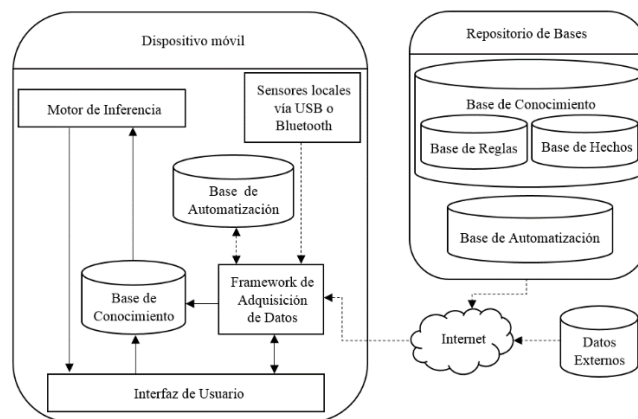


Fig. 1. Arquitectura conceptual del SBC para dispositivos móviles.

El Motor de Inferencia (MI) de la aplicación corre localmente en el dispositivo, actuando sobre la copia local de la BC. El framework de adquisición de datos cumple un doble rol, mantiene actualizadas las BC y BA, y la automatización de la captura de datos del exterior. La ventaja principal de la arquitectura radica en la centralización del mantenimiento de las BC, permitiendo que los cambios impacten en simultáneo en cada dispositivo móvil que utilice las mismas. La adquisición de datos desde sensores locales vía bluetooth o usb se encuentra prevista en futuras líneas de trabajo.

3.2 Base de Conocimiento

Para la configuración de la adquisición de datos automatizada se desarrolló un marco de trabajo o framework que facilite la configuración de la captura de cada dato. Estos

datos representan los hechos que componen la BH. Se define así la BA donde se detallan los datos para la búsqueda automática en la nube y la actualización de los valores. Para las BC, se adoptó el formato de bases definido por e2gDroid [18], para las BA propuestas se utilizó el lenguaje XML, estándar para la transferencia de datos [19].

Ejemplo de una BR de un SBC especializado en predecir si va a llover y recomendar el uso de paraguas. Se detallan tres preguntas al usuario, una regla de inferencia y el atributo objetivo.

```
REM Usar o no paraguas

RULE [Va a llover próximamente?]
If [nublado] = True and
[presión_actual] < [presión_3horas]
Then [recomendación] = "Llevar paraguas"

GOAL [recomendación]

PROMPT [nublado] YesNo
"Está nublado?"

PROMPT [presión_actual] Numeric
"Cuál es la presión atmosférica actual?"

PROMPT [presión_3horas] Numeric
"Cuál fue la presión atmosférica hace 3 horas?"
```

Ejemplo de una configuración de automatización de captura de datos para la BH del SBC, donde se detalla las rutas URL para la búsqueda automática de los valores de los atributos presión_actual y presión_3horas, y el valor por defecto para el atributo nublado.

```
<atributo>
<nombre>nublado</nombre>
<valor>True</valor>
</atributo>
<atributo>
<nombre>presión_actual</nombre>
<url>http://infinnitem.dyndns.pro/data/presion_actual.html
</url>
</atributo>
<atributo>
<nombre>presión_3horas</nombre>
<url>http://infinnitem.dyndns.pro/data/presion_3horas.html
</url>
</atributo>
```

3.3 Desarrollo de la Aplicación

El desarrollo efectuado para la construcción de un SBC se realizó en el lenguaje de programación Android [20], que funciona de forma local en un dispositivo móvil con Android (desarrollado para versión del S.O. 2.3.3 en adelante, probado en un Samsung I9300 con Android 4.1.2). La aplicación desarrollada permite: a) la explotación independiente de BC mediante el MI de la aplicación; b) la obtención de hechos a través de los datos provistos por el usuario como del exterior, en este último caso en forma automática y transparente al usuario al ser capturados desde la nube; c) el mantenimiento de las bases (BC y BA) se realice contra un repositorio central; d) revisar la trazabilidad de las reglas utilizadas.

Para correr el SBC se debe seleccionar y cargar la BC a utilizar. El sistema luego intentará adquirir los hechos a través de lo configurado en la BA, y de ser necesario generará preguntas al usuario hasta alcanzar un resultado. Para esto se realizan sucesivas búsquedas hacia atrás revisando cada regla, hasta encontrar un camino lógico que puede completar. Al finalizar todo el proceso, se mostrará en pantalla el resultado obtenido. En la figura 2 se compilan distintas capturas de pantallas de la aplicación corriendo las bases de ejemplo anteriormente detalladas. La primera captura muestra la pantalla de selección de BC; la segunda ejemplifica una pregunta realizada al usuario; y en la tercera se observa el resultado obtenido, con el agregado de una búsqueda relacionada realizada en Internet.



Fig. 2. Capturas de pantalla del nuevo SBC en ejecución.

Respecto al mantenimiento de las bases, el mismo se realiza desde una aplicación web con acceso al repositorio central, desde la cual se pueden crear, actualizar y borrar BC, como así también BA. La figura 3 muestra una imagen de la aplicación desarrollada para la gestión de la BC vía web. En el lateral izquierdo se pueden ver las BC publicadas para su selección, mientras que en sector central se observa el cuadro de edición de la BC seleccionada. En la figura 4 se captura el mantenimiento de la BA mediante la aplicación web. La aplicación posee una lista de BC en la izquierda y en la

parte central se ingresan las declaraciones, como ser una URL a un atributo en la BC seleccionada, en este caso presión atmosférica correspondiente a la BC Usar Paraguas.

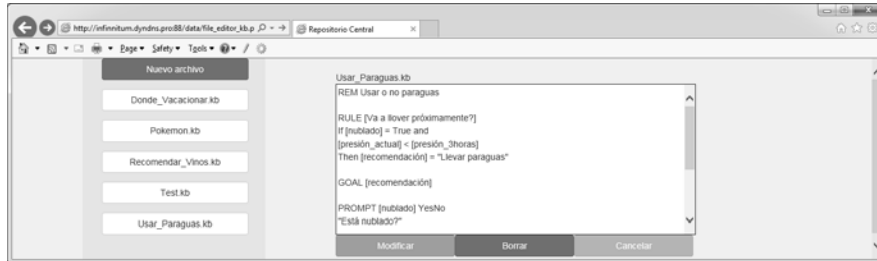


Fig. 3. Captura de pantalla de la aplicación web del repositorio central – BC.

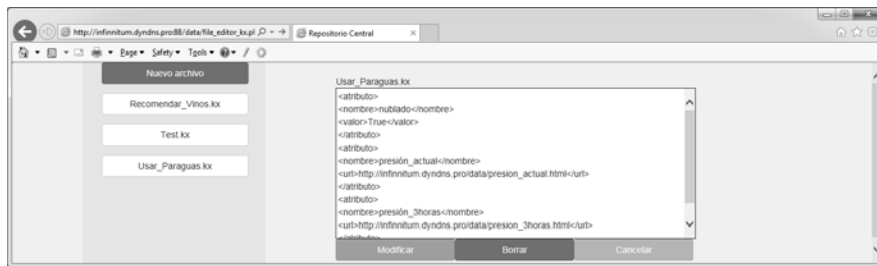


Fig. 4. Captura de pantalla de la aplicación web del repositorio central – BA.

Se presenta en la figura 5 el diagrama de componentes, en el cual se visualizan las interacciones entre las distintas interfaces del sistema. Mediante la aplicación móvil se puede interactuar con el MI y con el módulo de actualización de las bases (BC y BA). Tanto el MI como el módulo de actualización requieren para su funcionamiento interactuar con las BC residentes en el móvil. El módulo de actualización realiza la actualización extrayendo la última versión de las bases desde el repositorio remoto. Por el otro lado, utilizando el sistema web el usuario le da mantenimiento al repositorio de las BC y al de las BA.

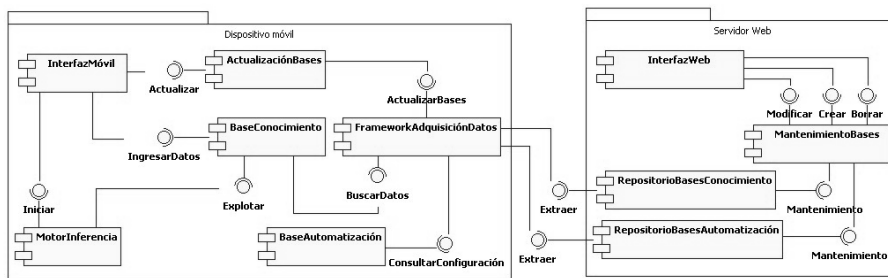


Fig. 5. Diagrama de componentes de la solución propuesta.

4 Pruebas de la Solución

Se efectuaron distintas pruebas con el fin de comprobar la arquitectura propuesta y su integración con la Base de Conocimiento (BC) y la Base de Automatización (BA). Estas pruebas se efectuaron en dominios particulares: predicción de clima, recomendación de lugares turísticos, diagnóstico de fallas de PC y maridaje de vinos. El desarrollo de pruebas se realizó con un set de siete casos para cada BC, con distintas configuraciones y objetivos puntuales. En la tabla 2 se detalla el set empleado para cada uno de los casos mencionados.

Tabla 2. Pruebas efectuadas sobre la solución propuesta.

Prueba	Objetivo
Comparación entre Motores de Inferencia (MI)	Probar el MI desarrollado contra uno existente. Se utilizó el mismo caso en el nuevo SBC y en e2gDroid.
Automatización de un solo atributo	Lograr una ejecución con una BA configurada con un valor fijo para un solo atributo.
Automatización de dos atributos	Obtener una ejecución con una BA que tenga dos atributos y sus respectivos valores fijos.
Automatización de la totalidad de los atributos	Ejecutar de forma totalmente automatizada, agregando a la configuración del caso de prueba anterior un valor fijo para el tercer y último atributo de la BC.
Automatización con URL	Conseguir el mismo resultado anterior pero que un atributo, en vez de tener un valor fijo, dependa de la búsqueda en una URL.
Automatización con URL no existente	Evaluar la tolerancia a fallos sin la URL configurada no existe. Se ejecutó la prueba anterior modificando el entorno para que la URL sea un recurso inválido.
Automatización con URL y borrado	Comprobar la capacidad del SBC de explorar otras rutas cambiando las respuestas obtenidas.

Continuando con el ejemplo presentado en la sección 3.2, se detalla en la tabla 3 un caso de prueba utilizando una BC que recomienda el uso de un paraguas dependiendo de la situación climática, logrando una ejecución totalmente automatizada, de tal forma que el usuario no ingrese ningún valor de forma manual. La Base de Hechos (BH) se confecciona con un valor fijo y con búsquedas de valores en dos direcciones web declaradas en la BA. La ejecución de la prueba se puede apreciar en la figura 7³. Se observa que se logra el resultado deseado, sin la necesidad de ninguna pantalla intermedia que pregunte al usuario el valor de ningún atributo.

³ Para ver la ejecución completa referirse a <https://goo.gl/z5EqxK> (<https://www.youtube.com/playlist?list=PL3WKnmIEYB91sJ5XvqbGRQxaDIXSsJEsH>)

Tabla 3. Caso de prueba detallado.

Pregunta	Respuesta	Origen	Configuración
¿Está nublado?	True	Base de hechos	Valor fijo: True
¿Cuál es la presión atmosférica actual?	1050	Base de automatización	Dirección URL: http://infinitum.dyndns.pr o/data/presion_actual.html
¿Cuál fue la presión atmosférica hace 3 horas?	1100	Base de automatización	Dirección URL: http://infinitum.dyndns.pr o/data/presion_3horas.html
Recomendación	Llevar paraguas		

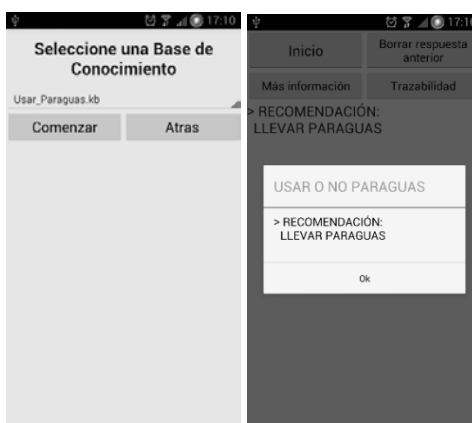


Fig. 7. Capturas de pantalla del caso de prueba en ejecución.

5 Conclusión y Futuras Líneas de Trabajo

La arquitectura propuesta contribuye a la implementación de aplicaciones en dispositivos móviles que brinden servicios útiles, facilitando la integración de sistemas basados en conocimiento y la actualización de datos, como así también el mantenimiento de la base de conocimiento. En el marco de las futuras líneas de trabajo se prevé la integración y experimentación con sensores y artefactos en el contexto del Internet de las cosas.

6 Referencias

1. Jackson, Peter (1998). *Introduction To Expert Systems* (3 ed.), Addison Wesley, ISBN 9780201876864.

2. Cornelius T. Leondes (2002). *Expert systems: the technology of knowledge management and decision making for the 21st century*, ISBN 9780124438804.
3. Smart Insights (2016). *Mobile marketing statistics 2016*, <<http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>>.
4. Statista (2016). *Smartphone OS global market share 2009-2015*, <<http://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems>>.
5. eXpertise2Go (2016), <<http://www.expertise2go.com>>.
6. Sandeep Kumar and Ravi Bhushan Mishra (2010). *Web-based expert systems and services. The Knowledge Engineering Review*. doi:10.1017/S0269888910000020.
7. Asabere, Enguah (2012). *Integration of Expert Systems in Mobile Learning*, University of Cape Coast, Cape Coast, Ghana, <http://esjournals.org/journaloftechnology/archive/vol2no1/vol2no1_8.pdf>.
8. Lopes, Abreu, Batista (2008). *Innovation in Telemedicine: en Expert Medical Information System Based on SOA, Expert Systems and Mobile Computing*, Federal University of Maranhão, Brasil, <http://link.springer.com/chapter/10.1007/978-1-4020-8741-7_22>.
9. Mitchell, Sharma, Modi, Simpson, Thomas, Hill, Goyal (2011). *A Smartphone Client-Server Teleradiology System for Primary Diagnosis of Acute Stroke*, Estados Unidos de América, <<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3221380>>.
10. Corcoba Magaña, Muñoz-Organero (2011). *Artemisa: using and android device as an eco-driving assistant*, Universidad Carlos III de Madrid, Madrid, España, <http://e-archivo.uc3m.es:8080/bitstream/10016/13091/1/android_JMTC_2011.pdf>.
11. Mestizo Gutierrez, Guerra Hernandez, Parra Loera (2008). *Desarrollo de un Centro de Ayuda Inteligente mediante el uso de Tecnologías de Internet*, Veracruz, México, <<http://cdigital.uv.mx/bitstream/123456789/32358/1/mestizogutierrezsonia.pdf>>.
12. France Telecom — CNET (1997). *ARTIMIS: Natural Dialogue Meets Rational Agency*, Francia, <<http://www.ijcai.org/Past%20Proceedings/IJCAI-97-VOL2/PDF/035.pdf>>.
13. Aguero, Rebollo, Carrascosa, Julian (2009). *Developing intelligent agents on the Android platform*, Universidad Politécnica de Valencia, Valencia, España, <http://www.gt-ia.upv.es/sma/tools/Andromeda/archivos/downloads/Description_of_Andromeda.pdf>.
14. StackOverflow (2011). *Integration of Drools (Expert System) with Android Projects*, <<http://stackoverflow.com/questions/8646934>>.
15. Verlaenen, Kris (2011). *jBPM5 lightweight? Running on Android!*, <<http://kverlaen.blogspot.com.ar/2011/03/jbpm5-lightweight-running-on-android.html>>.
16. d3web (2016). *Main*, <<http://www.d3web.de>>.
17. Albin, Stephen (2003). *The Art of Software Architecture: Design Methods and Techniques*, ISBN 9780471228868.
18. eXpertise2Go (2016), *e2gRuleWriter decision table software: e2gRuleEngine knowledge base generator*, <<http://www.expertise2go.com/e2g3g/e2g3gdoc/e2gRuleWriterRef.htm>>.
19. W3C (2016). *Extensible Markup Language (XML) 1.1 (Second Edition)*, <<https://www.w3.org/TR/2006/REC-xml11-20060816>>.
20. Android (2016). *Android Developers*, <<https://developer.android.com>>.

Sistema de Tarjeta Aumentada Basada en Conocimiento para la Asistencia en Emergencia Médica con el empleo de dispositivos móviles

Cristian Montalvo, Facundo Petrolo, Diego Sanz, Nahuel Mangiarua, Nicolas Verdicchio, Santiago Igarza, Jorge Ierache

¹ Grupo de Investigación en Realidad Aumentada Aplicada, Departamento de Ingeniería e Investigaciones Tecnológicas UNLaM. Universidad Nacional de La Matanza, Av. Florencio Varela 1903 (B1754JEC) San Justo, Provincia de Buenos Aires, República Argentina.
jierache@unlam.edu.ar

Abstract. Este trabajo describe la integración de realidad aumentada y sistemas basados en conocimientos aplicados a la innovación de las Tarjetas clásicas de identificación de un individuo en el contexto de los sistemas de Salud. Se muestran resultados iniciales de la aumentación de los elementos físicos de identificación de una persona (como tarjetas, credenciales, colgantes de chapa, etc.) con información virtual y la categorización del individuo en función de sus antecedentes de salud, efectuada a través de un sistema basado en conocimiento con el fin de contribuir a la asistencia médica temprana con información relevante frente a una situación de emergencia o descompensación.

Keywords: informática médica, realidad aumentada, asistencia, primeros auxilios, salud, emergentología.

1 Introducción

El Sistema de Tarjeta Aumentada Basada en Conocimiento para la Asistencia en Emergencia Médica, (al que denominaremos de ahora en adelante como Sistema) integra tecnologías de Realidad Aumentada, de Sistemas Basados en Conocimientos, Dispositivos Móviles (tales como smartphones & tablets) y de Elementos Físicos de identificación de individuos (como tarjetas, credenciales, colgantes de chapa, etc.) con el fin de contribuir a la asistencia y primeros auxilios de personas que sufren un accidente o descompensación en la vía pública o donde sea que nos encontremos.

Este sistema puede ser utilizado: internamente dentro de un hospital o centro de salud por su personal médico, al cual se le brindará información acerca de los antecedentes del paciente; como por el equipo de ambulancias quién poseerá dentro del vehículo una instancia del sistema con los registros de salud de los pacientes registrados, y también por el ciudadano que tenga la aplicación al momento de ayudar a una persona que necesita asistencia en la vía pública, desplegando información significativa relacionada con los antecedentes de la persona que requiere atención.

En la actualidad en el contexto de los sistemas y servicios médicos existen diferentes medios físicos por los cuales se asocia a un individuo con cierta información. Los medios físicos clásicos, como las tarjetas, tienen por objetivo

identificar a una persona de forma unívoca. El Sistema propuesta hace uso de Realidad Aumentada (RA), y tecnologías de Inteligencia Artificial (IA) en orden al empleo de Sistemas Expertos como herramientas de apoyo para la realización de tareas complejas [1]. En cuanto a la RA, esta tecnología generalmente es comparada con la Realidad Virtual, sin embargo no se refieren a lo mismo. Esta última introduce al usuario en un ambiente completamente virtual sustituyendo a la realidad física, mientras que RA complementa (aumenta) a la realidad física, manteniendo al usuario en contacto con la misma, a la vez que interactúa con objetos o información virtual superpuestas sobre ella [2]. La Realidad Aumentada es un conjunto de dispositivos que añaden información virtual a la información física ya existente, es decir, añadir una parte sintética virtual a lo real. Esta es la principal diferencia con la realidad virtual, puesto que no sustituye la realidad física, sino que sobreimprime los datos informáticos al mundo real. [3] La combinación de la información obtenida desde distintas fuentes, mas aquella que podemos obtener del contexto gracias al uso de los sensores y componentes que están presentes en las nuevas generaciones de dispositivos móviles, proporcionan la piedra angular para la construcción de aplicaciones que hagan uso de la Realidad Aumentada [4]. Esta característica especial brindada por la RA, es explotada por el sistema facilitando rápidamente datos personales e información médica relevante para la toma de decisiones (como estado de salud, enfermedades, alergias, operaciones, grupo sanguíneo, etc.) al aumentar el identificador físico del individuo, con la finalidad de contribuir a la asistencia y primeros auxilios de personas que sufren un accidente o descompensación.

Para cumplir con su objetivo el sistema de tarjeta aumentada basada en conocimientos se conforma de: a) Identificador Físico como tarjetas, credenciales, chapas colgantes de empleo común que sirve para la identificación unívoca de una persona con sus datos personales b) Marcador, una imagen impresa que se superpone al identificador físico (tarjeta) que actúa como elemento activador del contenido virtual de información; aumentado de esta forma con el uso de tecnologías de Realidad Aumentada al identificador físico con datos Personales, información médica relevante en relación a los antecedentes de salud o datos de contactos del individuo. c) Dispositivo Móvil encargado de enfocar y reconocer al marcador del identificador físico a través de su cámara incorporada, y permitir la visualización de la información virtual aumentando al identificador físico (tarjeta) de la persona sobre la pantalla del mismo, d) Sistema de Categorización: suma información a la tarjeta aumentada en relación a la importancia de los antecedentes de salud del individuo, esta información se determina con el empleo previo de un sistema basado en conocimiento, que permite la clasificación en función de tres grupos que se detallan en la próxima sección. En la figura 1 se muestra conceptualmente cómo se encuentra conformado el Sistema

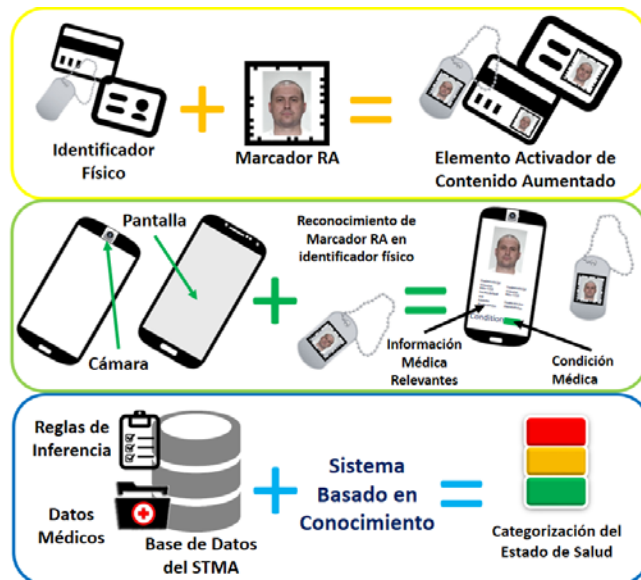


Fig. 1. Conformación del Sistema.

2 Sistema de Tarjeta Aumentada Basada en Conocimiento para la Asistencia en Emergencia Médica

El demostrador tecnológico del sistema se encuentra implementado en tres módulos funcionales que interactúan entre sí para cumplir con los objetivos del sistema. En la figura 2 se puede contemplar un diagrama conceptual de la estructura modular del sistema conformado por el: Módulo Web, el Módulo Móvil y el Módulo Principal. El Módulo Principal actúa como servidor entre el Módulo Web y el Módulo Móvil e incluye el sistema basado en conocimiento. Expone una API REST [5] desarrollada sobre Spring IO [6] y Apache Tomcat [7]. Aquí, se implementa el soporte de almacenamiento a través de una base de datos relacional con JDBC [8] con un ambiente de desarrollo MySQL [9]. El Módulo Web tiene como objetivo la carga de los datos médicos utilizando un formulario y generando un FrameMarker con la foto del individuo. Está desarrollado utilizando HTML en conjunto con JavaScript y librerías públicas como JQUERY [10] y Bootstrap [11], todo está soportado por un backend Java [12] a través de Apache Tomcat y Spring IO para conseguir flexibilidad. Finalmente, el Módulo Móvil manipula contenido tanto estático como los enriquecidos con realidad aumentada mediante una aplicación para dispositivos con sistema operativo Android, integrando funcionalidades del motor Unity3D [13] y el sistema de reconocimiento de Vuforia [14].

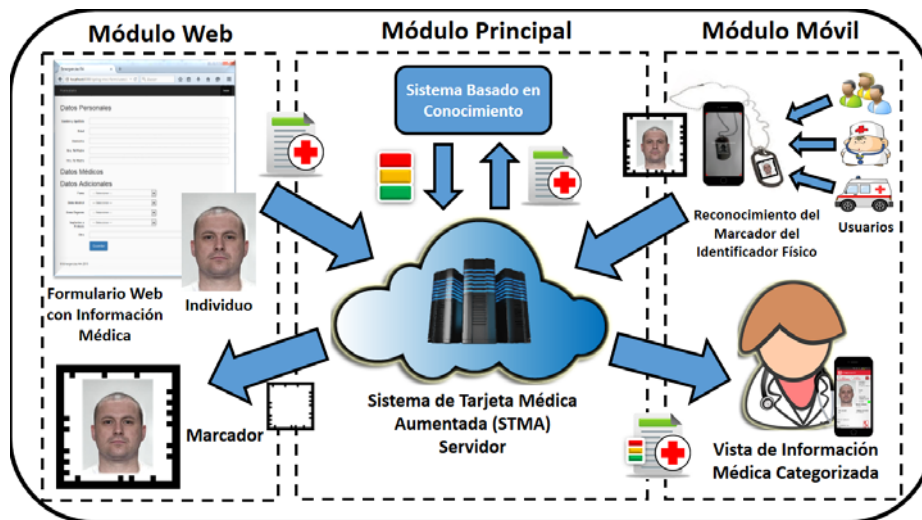


Fig. 2. Diagrama conceptual del Sistema.

En las siguientes subsecciones se detallarán cada uno de los módulos que conforman al Sistema de Tarjeta Aumentada Basada en Conocimiento para la Asistencia en Emergencia Médica.

2.1 Módulo Principal - Aplicación Servidor y Sistema Basado en Conocimiento

Este módulo está comprendido por una RESTful API que actúa como backend brindando los servicios de creación, edición, registro, almacenamiento y descarga de formularios web con datos personales e información médica relevante de los diferentes individuos. Está integrado al sistema basado en conocimiento para la categorización del estado del individuo en función de sus antecedentes de salud, ya que actualmente los centros hospitalarios/servicios de asistencias en emergencias en la vía pública presentan la falencia de tener que trabajar “a ciegas” con los individuos que sufren un accidente o descompensación. A partir de esta problemática se presenta la necesidad de incorporar a las funcionalidades iniciales de aumentación de tarjetas un sistema basado en conocimientos que logre categorizar al individuo en función de sus antecedentes como enfermedades, operaciones realizadas, alergias, etc. y de esta forma contribuir a definir un estado de salud del mismo a partir de sus antecedentes significativos ya precargado. El estado de salud del paciente brindará una información de forma rápida a la persona que lo esté asistiendo, el cual será identificado en la aplicación móvil como el estado de salud. En la figura 3 se puede apreciar que el Sistema Basado en Conocimiento (SBC) toma como entrada el registro médico de un individuo, basado en una serie de reglas que permiten al motor de inferencia del SBC lograr categorizar al individuo, generando como resultado un estado y grado de salud identificado por colores (rojo, amarillo, verde) y niveles de criticidad en función de

los antecedentes vitales. Como resultado de la conceptualización de conocimientos se presenta el Mapa de conocimiento, ver figura 4, el que identifica los atributos que caracterizan los distintos factores a tener en cuenta para determinar la situación meta, representada por el estado de salud del individuo. En este orden los atributos que contribuyen para el modelo basal en desarrollo son: medicamentos de consumo, alergias, antecedentes médicos, discapacidades, operaciones o datos sobre adicciones y edad del paciente. Inicialmente se aplicó a los atributos que representan alergias, antecedentes médicos, discapacidades y cirugías una ponderación, representando el valor el grado de influencia del atributo, que será determinado por otra componente específica del sistema basado en conocimiento.



Fig. 3. Sistema Basado en Conocimientos

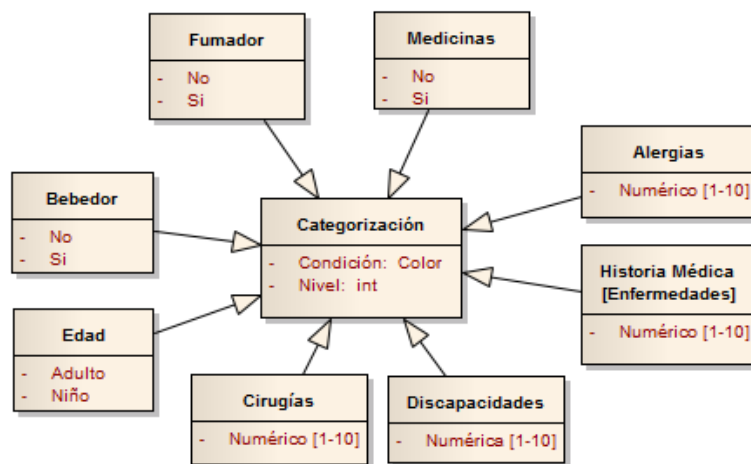


Fig. 4. Mapa de conocimientos de Categorización

2.2 Módulo Web - Aplicación de Carga de Datos

Este módulo le proporciona al usuario/médico de cabecera el servicio de ingreso de datos personales e información de antecedentes de salud relevante al sistema a través

de formularios web. Una vez completado este formulario es enviado al servidor del sistema quien lo registra y almacena en la base de datos. Posteriormente al registro se creará un FrameMarker (imagen utilizada como marcador para aplicaciones de Realidad Aumentada) en respuesta al usuario, el que podrá descargar y utilizar en el identificador físico (tarjeta, credencial, colgante de chapa) que actuará como disparador de la aplicación móvil, desplegando así la información que aumenta al identificador físico. En la Figura 2 lateral izquierdo se puede ver cómo está conformada la secuencia básica del ciclo de carga de datos médicos relevantes.

En la figura 5 puede verse un diagrama de despliegue que ilustra los componentes que están involucrados en la interacción entre la aplicación móvil y el servidor del Sistema. Esta aplicación se encuentra integrada por los componentes de creación de marcadores y contenidos (Marker Factory, Catalog Factory y Content Factory), el sistema de reconocimiento de marcadores (Marker Recognition System) y el componente Network Manager que proporciona un protocolo de comunicación entre la Aplicación Móvil y el Servidor STMA. Mientras que el servidor que aloja al Sistema de Tarjetas Médicas Aumentadas, está conformado por el Core quién interactúa con el Servidor de Base de Datos y los Controllers, que hacen nexo con el Navegador Web que sirve de medio para la creación y carga del contenido de los antecedentes de salud relevantes.

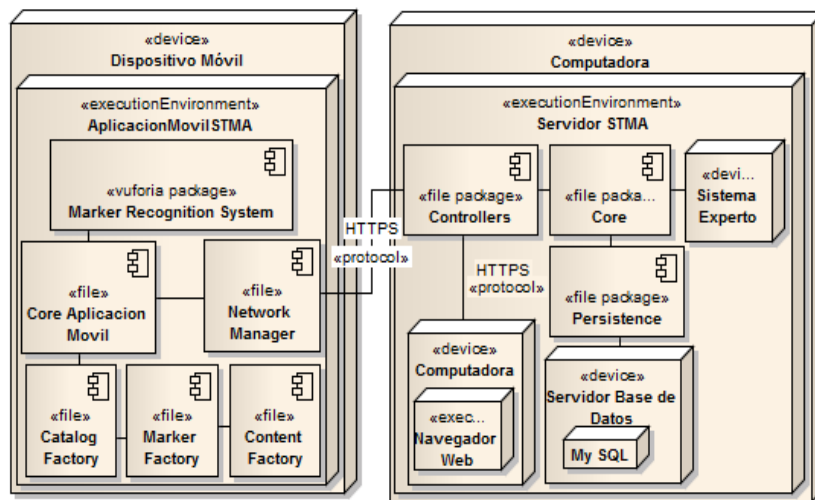


Fig. 5. Diagrama de despliegue del Sistema

2.3 Módulo Móvil - Aplicación para la Explotación del Contenido Aumentado

Este módulo está encargado del reconocimiento del marcador que posee el identificador físico del individuo al que se está auxiliando, representando el contenido aumentado de la aplicación en el mundo real. El funcionamiento de este módulo comprende la siguiente secuencia: 1) Si el usuario se encuentra registrado

previamente entre los historiales del servidor y posee marcador sobre su elemento de identificación físico (tarjeta) se procede a reconocer el mismo a través de la aplicación móvil. 2) Una vez reconocido el marcador, se envía una petición de descarga de contenidos al servidor del sistema. 3) Una vez completada la descarga de contenidos, la aplicación se encargará de desplegar datos personales útiles, información médica relevante y categoría del estado y nivel de salud del individuo, sobre la pantalla del dispositivo móvil de la persona que esté atendiendo al individuo. En la Figura 2 lateral derecho se observa el ciclo básico de explotación del contenido de información relevante de un determinado individuo.

La Figura 6 muestra a la aplicación móvil trabajando, donde se está enfocando el marcador sobre el elemento físico (chapa ID colgante) de un individuo. Se puede observar las opciones del contenido aumentado siendo desplegado en la pantalla del dispositivo.

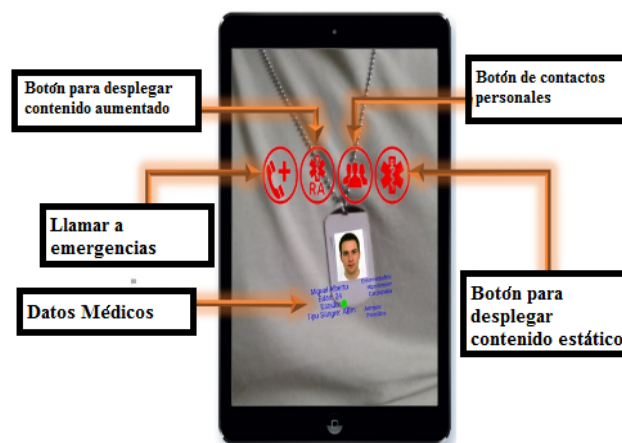


Fig. 6. Vista de realidad aumentada en el Módulo Móvil.

Se puede observar en la figura anterior que esta aplicación utiliza realidad aumentada, las opciones de botones que se presentan sobre la pantalla del dispositivo son para: a) llamar al servicio de atención de emergencias, b) llamar a contactos del individuo, c) mostrar datos médicos relevantes del individuo, d) activar contenidos en forma estática (sin la necesidad de enfocar posteriormente al marcador), se presenta la información del individuo distribuida en tres secciones dentro de la aplicación, que se muestran en la figura 7, (de izquierda a derecha) la primera sección proporciona información acerca de datos personales del individuo (nombre, edad, tipo de sangre, alergias, condición de salud, enfermedades anteriores) y el resultado de clasificación del estado de salud realizado por el sistema basado en conocimientos que se indica con un círculo en este caso en rojo (antecedentes considerables). La segunda sección muestra antecedentes de salud del individuo, donde podemos ver los detalles acerca de enfermedades, cirugías, alergias, operaciones, etc. Finalmente, la tercera sección

contiene datos de carácter complementario, por ejemplo: Si una persona es un donador de órganos, si es fumador si posee alguna fobia, entre otros.

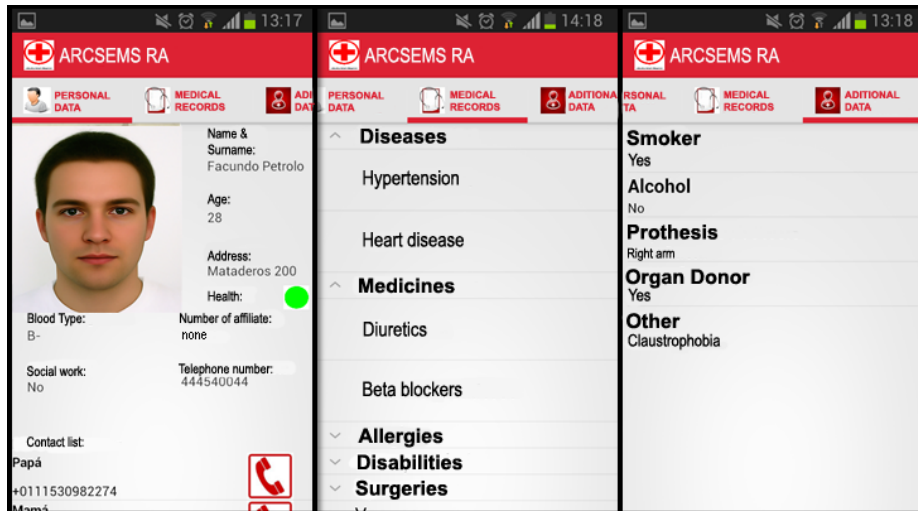


Fig. 7. Vista de las tres secciones de información del Módulo Móvil.

3 Resultados y pruebas preliminares

Se enuncia en esta sección las pruebas preliminares realizadas. En el mes de diciembre del año 2015 se convocó a un grupo de estudiantes de la carrera de medicina de la Universidad Nacional de La Matanza quienes se encargaron de reconocer marcadores y visualizar información médica relevante de un conjunto de individuos precargados en el sistema web a través de una versión inicial de la aplicación móvil. Luego se realizó una encuesta para evaluar el grado de usabilidad, tiempo de respuesta, innovación e interfaz intuitiva que brindaba la aplicación. Se obtuvieron resultados satisfactorios los cuales fueron publicados en [15]; los que se pueden visualizar en la Figura 8. Una segunda instancia de pruebas preliminares del demostrador tecnológico del sistema en su versión inicial fue llevada a cabo en WICC 2016 [16], presentando la misma encuesta a los participantes del evento, en la Figura 9 se puede apreciar los resultados de la misma.

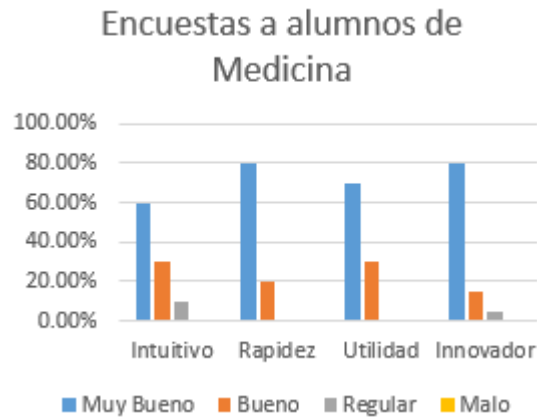


Fig. 8. Encuesta a estudiantes de Medicina

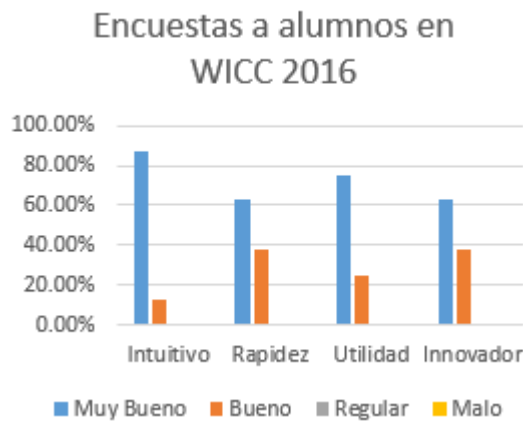


Fig. 9. Encuesta a estudiantes en WICC 2016

4 CONCLUSIÓN

El Sistema contribuye a la asistencia de personas que se encuentra en situaciones de emergencia, proporcionando información de sus antecedentes significativos y una categorización primaria de la persona en función de estos. Su empleo futuro contribuirá mejorar la eficiencia y calidad, de la atención a personas ante situaciones de emergencia o descompensación, proporcionando información de carácter relevante para que esta pueda ser consultada con rapidez a través del uso de la tecnología de Realidad Aumentada, permitiendo ayudar al personal médico a tener una primer impresión del paciente. Los desarrollos futuros están destinados a expandir la funcionalidad del sistema incluyendo capacidades futuras al integrar sensores que reporten opcionalmente parámetros del individuo (presión, pulsaciones, sudor, etc) , integrados a través de su dispositivo móvil el que reportará frente a una variación

significativa o tendencia de interés no solo una alerta, sino también actualizará el estado y nivel de salud en el sistema.

5 Agradecimientos

Este trabajo es financiado en el marco PROINCE del Departamento de Ingeniería e Investigaciones Tecnológicas (DIIT) de la Universidad Nacional de La Matanza (UNLaM), en función del requerimiento inicial planteado por Prof. Dr. Daniel Eduardo Martínez.

6 References

1. Gisselle Rey Salazar, Alex García Araya: Sistema Experto para determinar tipo de diabetes. pp.290-294. (2007);
2. Manresa Yee, M. Abásolo, R Más Sansó and M Véneré.: Realidad virtual y realidad aumentada. Interfaces avanzadas. (2011)
3. Jorge Mario Gaviria Hincapiel, Guillermo Alonso Castaño Perez, Byron PortillaRosero, Jose León Sierra Ospina: SLD203 Realidad Aumentada En el Tratamiento de las Enfermedades Mentales y las Adicciones. XV edición de la Convención y Feria Internacional Informática (2013)
4. Pablo J. Iuliano, Claudia A. Queiruga, Francisco J. Diaz: UNLP Aumentada: Desafíos y Retos. In: Biennial Congress of Argentina (ARGENCON), 2014 IEEE pp. 43-18. (2014)
5. APIRest, <http://www.restapitutorial.com/lessons/whatisrest.html>.
6. Spring IO, <https://spring.io/>
7. Apache Tomcat, <http://www.tomcat.apache.org/>
- 8.JDBC, <http://www.oracle.com/technetwork/java/javase/jdbc/index.html>
9. MySQL, <https://www.mysql.com/>
10. JQUERY, <https://jquery.com/>
11. Bootstrap, getbootstrap.com
12. Java, <https://www.java.com/>
13. Unity3D, <http://unity3d.com/es>
14. Vuforia, <https://developer.vuforia.com/>
15. Jorge Salvador Ierache, Santiago Igarza, Nicolas Verdicchio, Nicolás Duarte Duarte, Nahuel Mangiarua, Diego Sanz, Jonathan Barth, Cristian Montalvo and Facundo Petrolo: Augmented Reality Card System for Emergency Medical Services, International World-Conference on Bioinformatics and Biomedical Engineering (IWBBIO) (2016)
16. Verdicchio, Nicolás Nazareno, Sanz, Diego Rubén, Barth, Jonathan, Montalvo, Cristian, Petrolo, Facundo, Mangiarua, Nahuel Adiel, Igarza, Santiago, Ierache, Jorge Salvador: Líneas de investigación de realidad aumentada aplicada a la asistencia médica en el campo de la emergentología, Workshop de Investigadores en Ciencias de la Computación (2016) <http://sedici.unlp.edu.ar/handle/10915/53035>

Platform of Citizen Interaction: the contribution to the governability of New TIC's and Citizen Relational Management

Juan Doberti¹, María Laura Pepe¹, Mónica Larrosa¹, Carla Crocco¹,
Gabriela Gabay¹, Guillermo Hindi¹, Carlos Molina¹, Gonzalo Paradela¹,

¹ Engineering and Technological Researches Department; National University of La Matanza, Florencio Varela 1903, B1754JEC, San Justo, Argentina
{jdoberti, mpepe, mlarrosa, ccrocco, ggabay, ghindi, cmolina, cparadela}@unlam.edu.ar

Abstract. Understanding "governability" as an interaction process among government and society, the New Technologies of Information and Communications (NTICs) can contribute as a strategy of improvement and conversion in the type of communication between both actors. This article introduces concepts referred to Customer Relationship Management applications turned into Citizen Relationship Management tools. The plus of a modeling of the multiplatforms functionalities that this solution offers and a protocol of implementation and system migration is a step forward in the achievement of obtaining an electronic government with civil democratic interaction and transparency in the management. The idea of a Platform of Citizen Interaction that auspice as a repository of the communication between public and civil entities allows having a unique and centralized database capable of being segmented of different ways to attend to a great diversity of subject matters.

Keywords: E-Government, Citizen Relationship Management (CiRM), Platform of Citizen Interaction, Customer Relationship Management (CRM), New TICs (NTICs).

1 Introduction

New TICs are currently interceding as indispensable means between government and citizens, generating exclusive communication channels between them. In our so informed society, there is an increasingly need to be able to traverse the veil that conceals the actions of the public sphere and the access to the information that state organizations possess. The governability -understood as the interaction between the government and the society-, is demonstrated in the success of this process. "Governments worldwide are faced with the challenge of transformation and the need to reinvent government systems in order to deliver efficient and cost effective services, information and knowledge through information and communication technologies. Development of information and communication technologies catalyzed

and led up to E-government". [1] The current society demands major transparency on behalf of the governments, more decentralization, and more independence to express their issues, desires, prerogatives and priorities. The civil participation is one of the most important components of the democratic governments, especially municipal ones, with functions directly related to the territorial aspects and to the satisfaction of the needs of their local community. The citizenship and the social organized movements have increased their participation in the process of governmental decisions making, and in many cases it is possible to glimpse an effort to reach public decisions in cooperation with the population, and the role of the E-government has been highly significant to such an end.

The use of the New TICs (NTICs) has helped in the last years to improve governability and participation. It can be proved for example, in the use of the social networks for political mobilizations and campaigns of obligatory alphabetization. The increasing accessibility to these technologies, which include -in addition to computers-, the Internet, mobile communication and other mass media, has allowed, among other things, a major access to knowledge and the expansion of cultural activity; but additionally it can be potentially a powerful tool for the development of a new critical conscience and of civil participation.

The great summit that is having nowadays the E-government has been a clear consequence of the exponential development of these tools. "Electronic government refers to government's use of technology, particularly web-based Internet applications to enhance the access to and delivery of government information and service to citizens, business partners, employees, other agencies, and government entities." [2]

Its influence in governability is outlined because of its application in justice, security, human rights and education, as well as for the best accessibility to information of public interest and of consumer protection organisms. These tools also contribute in the interaction with the different government organisms on providing information on its actions, and there is an increasing trend to expand its use on the services given to the citizenship, such as obtaining identity documents and death certificates or to facilitate the payment of taxes and obligations.

According to Barak Obama, in his Memorandum "Transparency and Opened government", he indicates that a government opened to the citizenship "will strengthen our democracy and will promote the efficiency and the effectiveness in the government". For this commitment, in the same document he clarifies:

- The government must be transparent. The transparency promotes the account surrender and provides information to the citizens on what it is doing its government.
- The government must be participative. The public commitment improves the efficiency of the government and improves the quality of its decisions.
- The government must be collaborative. The collaboration involves actively the citizens in the performance of its government." [3]

The integration of the diverse alternatives that offer the new technologies allows improving these processes in a bidirectional way, allowing obtaining and processing

information to the pace of the "modern times", contributing this way with the governability understood as an “overarching concept, as the quality of a social-political entity as a whole, not an static quality; to the contrary, always changing”[4].

One of the fields in which these technologies can tender major contributions is basically to obtain public information from the governmental organisms. This should not be understood merely as the official online publication of accountability information and data that support transparency, only interpreted by professionals. It should be taken into account the rest of the population, with differentiated texts for specific target groups, guaranteeing the satisfaction of the social demands as for the access to the information, the empowerment and the creation of citizenship. Additionally, these tools can contribute so that governments could obtain information from the community and consequently elaborate more effective policies and action strategies, in topics as, for example, security and prevention of crime.

Understanding this increasingly need of incorporating the NTICs into the relation between citizens and the different governmental scales, exist nowadays multiple channels of virtual communication among the diverse actors, motorized in the last years principally for the so called "social networks". And in this society in which we live it is not possible to ignore the importance that is having this trend, which, as never before in the history of the humanity, is going through so many different ages and socioeconomic levels, and even overcoming the limits of distance.

In general terms, it might be said that the subject matter shows the following situation: [5]

<p>Strengths:</p> <ul style="list-style-type: none"> -Is a State policy -It is fitted inside a strategic governmental plan -The government has the necessary aptitude to have success in the implementation of the plan -There have been successful experiences in similar actions 	<p>Weaknesses:</p> <ul style="list-style-type: none"> -Lack of suitable communications infrastructure -Absence of necessary information systems -Lack of technological resources -Dependence of the financial availability of the State -Lack of qualified Human Resources
<p>Opportunities:</p> <ul style="list-style-type: none"> -Acceptance of the society of solutions to their demands of services -Increase of penetration of the TICs -Availability of new schemes and solutions of telecommunications 	<p>Threats:</p> <ul style="list-style-type: none"> -Lack of commitment of the governments -Absence of budget for financing -Inability to have the physical necessary spaces -Politicization or appropriation of the results or spaces

2 Implementation of a CiRM: preponderant aspects

Any organization, so much of public as private sphere, has as aim to cover the needs of a specific group of people. The difference is that in some cases the purpose is to obtain economic profitability and in others, the final aims are of different nature. The application of technologies such as CRM systems is of usefulness in both cases. "CRM is a philosophy and a business strategy supported by a system and a technology designed to improve human interactions in a business environment." [6] It is already demonstrated the efficiency of these solutions of software in the private sector, but recently it has been verified also that, in the governmental scale, the use of CRM contributes to a more efficient attention of the citizen, enabling the interaction by means of multichannel contacts where the NTICs possess a role of relevancy.

However, the focus should be diametrically opposed, because while in the private sector CRM is a very important tool for segmenting customers according to their economic potential, when talking of citizenship the criterion to segment involves identifying the most vulnerable, so that these ones are the beneficiaries of a "preferential treatment" according to their needs. In definitive, while in private management the use of CRM is customer oriented, in the public sector evolves to specific features, acquiring the name of Citizen Relationship Management (CiRM), with objectives such as measure citizens approval (in order to act according to their satisfaction degree) and know their concerns. In general, "government has not yet arrived at a strategic understanding of how to apply CiRM effectively and maximize citizen participation. Therefore an examination of CRM is appropriate and necessary to developing a more complete understanding of the power, potential, and pitfalls of CiRM", [7] basically by means of a plan to focus on people necessities using different types of management methodologies to succeed.

Deeper into the implementation of a CiRM, essentially, this application represents part of the front office from the point of view of the citizen. Subsequently, the process done with this interaction will be handled by the back office, which normally consists of a system called ERP (Enterprise Resource Planning).

In the Figure 1 are represented the features that the use of an application of this nature has, helping not only the citizens so they can accede to the services that the agencies provide (orders, claims, status of their transactions, services payment, etc.), but also, for the own organization, it works as an element of control and transparency, as well as an opportunity to approximate to the citizens, who feedback through their requirements the need for a continuous improvement in service.

It would be important to remark, then, that there is a "need to manage citizen relationships and open new models of citizen participation. However, as with eGovernment in general, CiRM increasingly demands that agencies act across jurisdictional boundaries and find new forms of governance." [8]

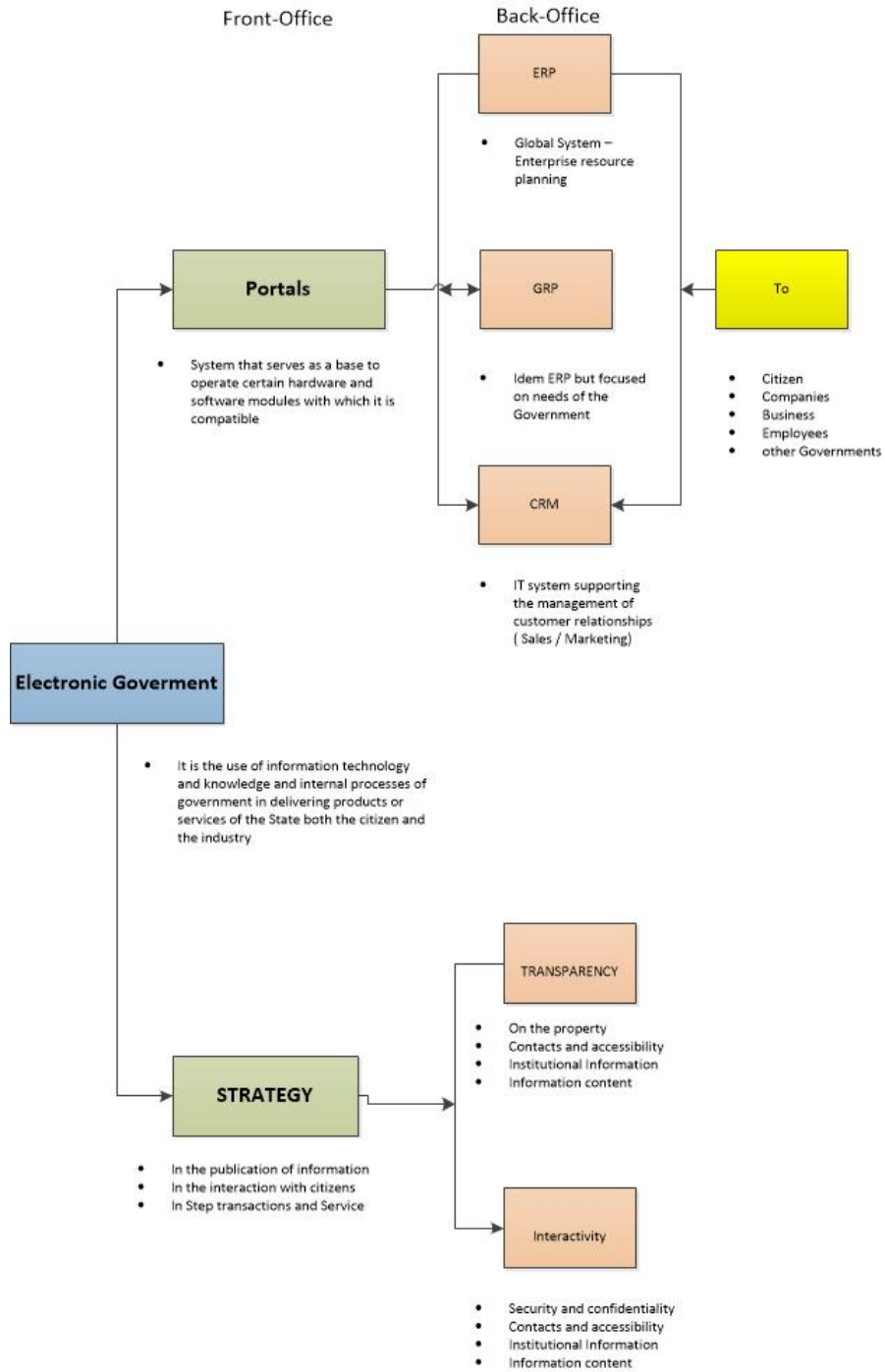


Fig. 1. Functionalities

Nowadays, as it happens in another type of technologies, there is a big offer of free software available or paid systems (that can also be stiff tools or solutions of software made-to-measure). It is important to mention that any deployment option must interact with office automation tools in use by any citizen, in order to do the interaction as smooth as possible. Obviously, to adopt one or another depends on the structural requirements of the area in which the application will be installed. In general, if the number of personnel is relatively small, the best option is to use a packaged CiRM. Anyway, the difficulty with this type of packages of software is that the adjustment of the standard functionalities can be difficult, because for a successful implementation there is needed of understanding a different conception from this type of software. If the quantity of users that they will use this tool is major, it will be better to use, of base, a CRM to the measure of the need. "There is a lot to making CRM decisions and there are a lot of ways to implement CRM programs." [9] This information is important according to the governmental scale in which is going to be applied the CiRM, since exist either very small areas of application or spaces of very significant dimensions. Needless to say that the costs of tools in both cases are very different, which is not a minor detail, and that will be another point to consider.

It is important to understand that the installation of a CiRM will be creating a single repository of relations with citizens that will store all interactions regardless the communication channel that has been used (e-mail, text message, web page or other). It is here where the role of the NTICs acquires an absolutely preponderant character, since they facilitate a massive access of citizens of very diverse characteristics. Besides these considerations, it is necessary to notice that there must be identified exhaustively all the areas affected by the implementation of this software, such as the IT area and the one of attention to the citizen in the first instance, but also all those areas that have active responsibility on the needs and requests of the citizenship or that take charge of processing information and generating policies and strategies of government.

Any CRM system consists of three parts: the operational one (in case of the CiRM they will be those typical functions related to the service to the citizen), the collaborative one (conductive thread between the citizen and the public entity,) and the analytical one (which carries out the capture, storage, extraction, processing, interpretation and generation of reports of information of the citizen, for a further use of the application). "CRM success requires the seamless integration of every aspect of business that touches de customer –including people, process, and technology- revolutionized by the Internet. Each component presents significant challenges, but it is the ability to integrate all three that makes or breaks a C RM system". [10] According to the governmental scale that is implementing this application, the role of the analytical component acquires dimensions of high sensibility and importance, since it is the one that contributes with information of interest from the political point of view for the decisions making.

The implementation of a CiRM solution, in general lines, can be summarized in the following stages:

1. *Analysis*: determination of the needs of the citizen.

2. *Diagnosis*: design in function of the necessities of the citizen and of the entity of government where it will be implemented, and determination of the requirements of technological infrastructure. It is important to set objectives and strategies between technical teams and managers.
3. *Performance*: system implementation, complying with tracking software and project stages. The necessary technology is installed and also the necessary trainings are done. The key indicators of the processes are ongoing to be able to have a feedback of the functioning of the system.

Another aspect to consider is the diffusion of this channel of participation for the citizens supported in the NTICs, it should also be an aspect to bear in mind for the success of the implementation, because it will need a pertinent diffusion strategy. Besides, it will be necessary to take into account the organizational structure of the organism, because furthermore will probably be needed some changes, since doubtless starting this solution of software should lead to a reclassification that will have to be carefully foreseen in order to not affect in a negative way all the involved actors. The most important thing is to recognize that some organizational changes are likely to be generated. Another probable consequence of the implementation of a CiRM platform is related to the level of delegation allowed to the employees responsible for the civil attention. Provided that the requirements of the citizenship will have to be structured by high degree of detail, to achieve an equal follow-up will be necessary a major capacity of response, and consistently, it will be necessary to delegate major authority for the already standardized questions. It will be necessary to pay attention to the existent barriers in order so that an entity of government can adopt a system of these characteristics. Public agents are not always ready to go through the cultural change -and of behavior- that is necessary for a satisfactory functioning of a CiRM. The success or failure depends entirely on the commitment of people and their flexibility to change.

After the implementation and the beginning of the utilization of the system there will probably be needed to be instrumented new functionalities to this tool, so, at the moment of choosing the best alternative, this factor will have to be taken into account. In an implementation of a made-to-measure tool, the updates and new functionalities will have to be executed by IT personnel specialized in the application, whereas a packed software will probably allow automatic updates through the Internet. This is the reason why a balance between maintainability and customization must be achieved.

In synthesis, implementing a CiRM system has to do with seeing some of the benefits related basically with information managing, which can be obtained in first instance and re-used; even the preferences of the citizens can be capitalized by means of the analysis of already done interactions. In addition, the information could be shared between different government entities in order to accelerate some processes. Of course services might be personalized referring to a geographical zone, social level, and other functional aspects and, for example, there might be sent automatic notifications of settlements or other obligations and even indicate to the citizen where they can do certain steps.

3 Platform of Citizen Interaction

Currently, the different levels of government have mostly implemented some websites with some degree of interactivity, where they offer their citizens more than news about the government, guidelines or "useful information". In this way, it is very important to possess a tool software that should allow these interactions, but it is also indispensable to think about modeling all the possibilities that this format offers, as well as a protocol of transfer of management (in person or through web pages but used only as a support tool), to an on-line management with permanent interaction with the citizenship, which often faces with the difficulty of "not being able to locate the institutional persons in charge of the decisions making. The solution of this problem has to do with connecting with the population on a bidirectional way and that provides to the informed person a channel of response that guarantees the return of the information or of the demand to the correct person in charge of area". [11] For this reason, the implementation of a Platform of Citizen Interaction based on the utilization of a CRM tool turned into a CiRM application, with the attaché of a modeling of the multiplatforms functionalities that offers this solution and a protocol of implementation and migration system, is a step forward in the achievement of obtaining an electronic government with civil democratic interaction and transparency in the management.

There exist diverse ways of generating this platform sustained by all those road links between the citizens and their authorities: in person, telephonic (call center), text message (msj), WhatsApp (wpp), Facebook, e-mail, Web route, etc. This plurality generates the following advantages to the user and to the government:

- The citizen is not tied to the only way of communicating with a public organism, since in all the channels that they should use, their communication record stays registered.
- This record frees the citizen of the person who attends him, since whoever is the agent who takes again the communication, has access to a record or "conversation" with the citizen (claims, steps, denunciations, etc.).
- The record of every citizen generates knowledge of the person by the public organization, which might use this information to produce profiles that could be used for offering services segmented by the interests of the contributors.
- Generation of real and constant statistics based on the reality of the citizens, in order to take the pertinent initiatives for the solution of detected problems.
- Civil empathy among the possibility of being able "to talk" with their authorities, feeling that their voices are being listened.
- Major democratic quality for the different areas of government, with major participation and clarity in the management, with the consequent political benefit.

- Rational utilization of the NTICs and of the social networks, which makes simpler the life and the work of the citizens.
- Inclusion of citizens in decision-making, with binding or not surveys, about decisions in the district.
- Knowledge by the taxpayers of the authorities, their history, heritage, activities, etc.

All these reasons and many more emanate since succeeding in establishing a protocol and implementing this Platform of Citizen Interaction, generating openness of government to its citizens: "Open Government is not a one-way street from the state to the citizens but a dialectical and multidirectional process that nourishes the actions of the government, citizens, companies and intermediary institutions". [12]

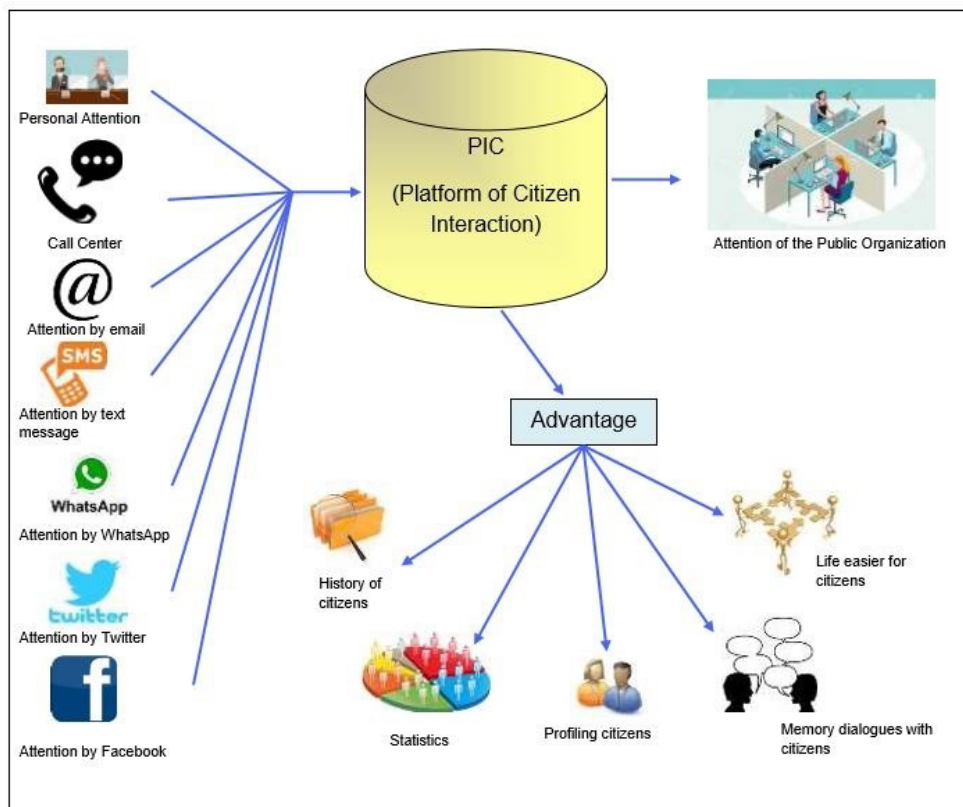


Fig. 2. Platform of Citizen Interaction

4 Conclusions

In current times, "the diffusion to planetary scale of the new technologies of the information and communication and of the IT networks, has started a process of connectivity without precedents in the human history, which has allowed the multiplication of the possibilities of generating and socializing information and knowledge, eliminating progressively the time and space barriers. Probably the most symptomatic phenomenon of these changes is the Internet, network of networks that has had a substantive impact in the economic processes, in the culture and in the generation of new modalities of interaction, communication and exchange of experiences between the different actors, institutions and social movements ". [13] The combination of these tools, used for the civil management together with CRM systems, allows offering a more efficient attention, facilitating the interaction through multichannel contacts. As well as in the private sector is observed that the management is orientated towards the client, over all as an economic agent, in the public sector, this principle applied to the citizenship, might be understood as the necessity to know their interests and to manage agilely the interaction with them, and besides, offering services to the population beyond the channel of participation, dealing and acting on the matters that do not satisfy the population, and being able to measure the degree in which their interests are being interpreted.

These topics have been studied for decades by theorists who have developed various philosophies that explain how citizens, primarily through the vote, expressed their will, concerning the provision of public goods, merit goods and other goods and services provided by the state. Today the NTICs and other IT tools like the CRM have converted themselves in instrument of communication and generation of opportunities in order that the citizens make hear their "voice" to the different governmental scales. And in the same way, as different disciplines like economy are today developing models that explain how these instruments transform the economic relations between private agents, it should be important to go deeper also in understanding how it concerns the public sector and - in specially - its relation with the citizenship, focusing in the resolution of problems of asymmetry of information inside the state.

The challenges that produce the integration of these tools in a Platform of Citizen Interaction turn about the question of how to allow the advance of different segments of the society towards the innovation and adjustment of new technologies -or new applications of already installed technologies-, to improve the processes of government and to reduce the social and economic differences. It is necessary to define public policies that are capable of promoting the use of this technology, contributing to the social integration, to the expansion of the human freedom and to the reduction of the digital gap. The problem is not so much of technical or economic nature, but of political will and of the desire of equality and justice. The governments must develop inclusive policies with social participation in order to carry it out. In synthesis, previously, these policies must already exist then the NTICs can significantly promote the participation and partnership.

References

1. Fang, Z.: Government in Digital Era: Concept, Practice, and Development. International Journal of The Computer, The Internet and Management. Vol. 10, No.2 (2002)
2. Maric, M.: End user oriented e-Government model. In MIPRO 2011 Proceedings of the 34th International Convention. IEEE. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5967318&isnumber=5967009>
3. Obama, B.: Transparency and Open Government: Memorandum for the Heads of Executive Departments and Agencies. https://www.whitehouse.gov/the_press_office/TransparencyandOpenGovernment
4. Kooiman, J.: Governing and Governance. SAGE. 2003
5. Province of Misiones Government, Argentinean Republic. Technological Center of Citizen Attention Project. <http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/ProyectoVersionFinal.pdf>
6. Greenberg, P.: A Commonwealth of Self-Interest. CRM Magazine. 2003
7. Huebner, J.: E-Participation and Citizen Relationship Management in Urban Governance: Tools and Methods. In Citizen E-Participation in Urban Governance: Crowdsourcing and Collaborative Creativity. IGI Global. 2013
8. Schellong, A.: Citizen Relationship Management. European University Studies. Peter Lang. 2008
9. Greenberg, P.: CRM at the Speed of Light, Fourth Edition: Social CRM 2.0 Strategies, Tools, and Techniques for Engaging Your Customers. McGraw Hill. 2010
10. Goldenberg, B.: CRM automation. Prentice Hall. 2003
11. Alcántara García, L., Santin Moreno, C.: Platform for civil participation for mobile. Tecnim@p. Sevilla. 2006
12. Golosetti, L.: Open Government: a management model citizen-centered. Initiatory Foundation. Santa Fe. 2016
13. CEPAL, Proyect ITA/04/002. RISALC, Network on line of Social Institutions of Latin America and the Carib: The new information and communication technologies and the social institutionalism. Towards a management based on the knowledge. 2005

Sistema autónomo de video conferencia motorizado - Implementación sobre Raspberry PI

Daniel Giulianelli, Pablo Vera, Rocío Rodríguez, Gabriela Vallés, Mariano Dogliotti

Universidad Nacional de La Matanza
Departamento de Ingeniería e Investigaciones Tecnológicas
San Justo, Buenos Aires, Argentina
{dgiulian, pvera, rocio.rodriguez, gvalles, mdogliotti}@unlam.edu.ar

Resumen. En el presente artículo se plantea la importancia de desarrollar una solución para Video Conferencia que ofrezca un sistema integral con un hardware dedicado de bajo costo, el cual a través de distintos componentes permita la motorización de una cámara que seguirá al orador cuando este se desplaza por su entorno. Permitiendo de este modo poder realizar video conferencias interactivas con manipulación de objetos en donde siempre el foco de atención esté dirigido al orador. Se presenta la estrategia para la construcción del sistema y las selecciones de hardware que resultan necesarias para llevar a cabo el mismo.

Palabras Clave: Video Conferencia, Hardware Dedicado, Raspberry Pi, Sensores, Motorización

1 Introducción

Los sistemas de videoconferencia son cada vez más populares, hoy existen herramientas de uso masivo como por ejemplo Skype de Microsoft, Hangout de Google, que permiten entablar videoconferencias rápidamente incluso con varias personas en simultáneo. Sin embargo, muchas veces estos sistemas se ven afectados por una pobre performance y falta de calidad al momento de realizar una videoconferencia, principalmente cuando la misma es de una duración considerable. Existen empresas como Cisco [1] o Polycom [2] que se dedican a realizar software y hardware específico brindando soluciones de gran calidad pero con el inconveniente de los altos precios para poder acceder a estos equipos. Es por eso que se busca crear un equipo de videoconferencia dedicado que tenga características profesionales y que además pueda ser implementado con hardware de bajo costo.

Los requerimientos del sistema son: (a) Solución autónoma que permita conectarla directamente a un televisor o monitor para uso; (b) Inicio automático de la aplicación con solo encender el equipo; (c) Control por medio de hardware sencillo como un teclado para smart tv o un control remoto; (d) Fácil configuración inicial; (e) Cámara que permita mediante motorización cambiar el ángulo de visión o seguir al orador automáticamente mientras este se mueve; (f) Conexión directa por IP sin necesidad de un servidor central; (g) Posibilidad de utilización de IPv6 para mejor

performance; (h) Posibilidad de realización de videoconferencia entre varias personas al mismo tiempo; (i) Video fluido con alta calidad de imagen.

Esto trae aparejado el desafío de planificar la utilización de hardware dedicado que permita implementar el sistema autónomo de video-conferencia (VC) motorizado, considerando Cámara, Sensores, Servomotores y Hardware específico donde poder ejecutar la solución.

El desarrollo del sistema de video conferencia en un dispositivo dedicado de tamaño reducido permite crear una solución de fácil implementación ya que con solo conectarla a un televisor o monitor ya podrá utilizarse sin necesidad de poseer conocimientos adicionales.

Al tener un sistema dedicado en el hardware el software debe contemplar los siguientes puntos: (a) El sistema debe iniciar automáticamente al iniciar el equipo; (b) No debe permitir salir del sistema; (c) En caso que por alguna razón como un error desconocido el sistema se cerrará, se deberá disponer de un proceso automático que intente volver a iniciarlo y si no es posible reiniciar el equipo; (d) Se debe diseñar una interfaz amigable para cualquier tipo de usuario tomando en cuenta que puede tener pocos o nulos conocimientos en el manejo de tecnología; (e) La interfaz debe estar diseñada para ser manejada por dispositivos sencillos (inicialmente se desarrolla una interfaz para los teclados de smart-tv y una segunda versión incorporará el manejo desde un control remoto tradicional); (f) Al desconocer en qué tipo de pantalla se utilizará, se debe diseñar una interfaz de usuario que se visualice correctamente en cualquier resolución y tamaño; (g) La configuración inicial para el uso de la aplicación debe realizarse de la forma más sencilla posible, guiando al usuario en cada paso (por ejemplo al configurar la red wifi).

2 Selección del Hardware

Se analizaron distintas plataformas en las cuales podría desarrollarse la solución planteada y se opta por Raspberry Pi debido a su mayor poder de cómputo sumado a la posibilidad de utilizar una cámara integrada de alta resolución. La raspberry además cuenta con puertos especializados para la comunicación directa con hardware externo lo que facilita las conexiones necesarias para poder realizar la automatización de la cámara.

La Raspberry Pi 2 Model B [3] tiene las características en cuanto a hardware que se presentan en la tabla 1.

Tabla 1. Características Raspberry PI 2 Model B

RAM	1 GB
Almacenamiento	Micro SD
Procesador	ARM A7
Velocidad	900 Mhz
Ethernet	Sí
Salida HDMI (audio y video)	Sí
Nº puertos USB	4

Soporta instalación de S.O.	Sí (Linux)
Cantidad de GPIO (puertos útiles)	40
Lenguajes de programación soportados	Python, C, C++, Java, Perl, Ruby, etc
Alimentación	5v
Tamaño	85 x 56 x 17mm

En una memoria SD es posible instalar un sistema operativo como Linux y luego un programa que permita realizar algo en particular, en el caso del presente proyecto una aplicación de video-conferencia, desarrollada específicamente para este hardware. Cabe destacar que actualmente ya existe una versión superior en el mercado de la raspberry pi que incluye mejoras en hardware y conectividad wifi incorporada, dicha versión fue lanzada con posterioridad a la adquisición del hardware para realizar el proyecto. De todos modos la versión adquirida tiene las prestaciones necesarias para poder implementar la solución.

2.1 Cámara

Cada uno de los nodos está implementado con una solución de hardware que incluye una Raspberry Pi 2 Model B. Cada una de ellas se conectará por HDMI a un monitor o televisor, se le agregó a la solución una cámara interna usando el puerto SCI de la Raspberry Pi (ver figura 1).

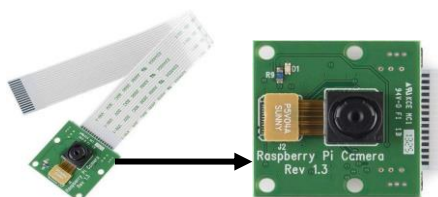


Fig. 1. Módulo de Cámara – Puerto SCI

La ventaja de la cámara integrada mediante el puerto SCI es la velocidad y la calidad de captura de imagen y video. El gabinete de la Raspberry trae un zócalo en donde se inserta la placa correspondiente al módulo de la cámara (ver figura 2). Los modos de videos soportados por la cámara integrada son: 1080p 30fps, 720p 60fps y 640x480p 60/90fps



Fig. 2. Módulo de Cámara conectado a la Raspberry Pi 2 Model B

2.2 Sensores

Se incorporan diversos módulos y elementos adicionales a la Raspberry Pi: Módulo de WIFI por el cual se establecerá la conexión a internet; Sensores (sonido y

ultrasonido – ver figura 3), los cuales se conectan a un puerto denominado GPIO (General Purpose input/out) especialmente diseñado para operaciones de entrada/salida. Este puerto tiene distintos pines que pueden utilizarse para controlar fácilmente tanto sensores como actuadores.

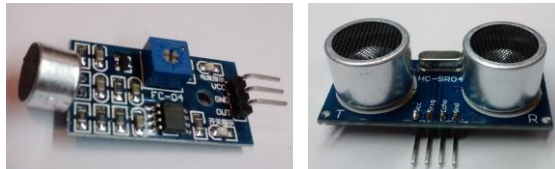


Fig. 3. Sensor sonido y ultrasonido

El sensor ultrasónico HC-SR04 para Raspberry Pi es un sensor que permite medir la distancia entre un objeto y el sensor. Tiene dos cilindros uno de ellos transmite un pulso de ultrasonido (de alta frecuencia, no audible por el oído humano), el cual rebotará contra el objeto que está frente al sensor y será recibido (dicho rebote) por el otro cilindro. Cuanto más cerca esté el objeto, el tiempo de recepción de dicho pulso será menor y por medio de ese tiempo se puede calcular la distancia con el objeto en cuestión, en un área de 3cm a 3m (con una precisión de 3 mm).

2.3 Motorización

Los sensores permitirán detectar la posición del orador dentro de una sala y motorizar la cámara para que pueda moverse siguiendo al orador. Para mover la cámara se agregan dos servomotores (vertical y horizontal) los cuales moverán la cámara (ver figura 4).



Fig. 4. Base de montura para la Raspberry PI con la cámara integrada – Sensores en dos Ejes

Los servos son motores de corriente continua (CC), pero no pueden lograr un giro continuo de 360°, están preparados para moverse a un ángulo fijo en respuesta a una señal de control, y mantenerse fijos en dicha posición. “Un servo principalmente está formado por un conjunto reductor (engranajes), un motor de CC y por último por un circuito de control, aunque en la práctica se comporta como un bloque funcional que posiciona su eje en un ángulo preciso en función de la señal de control” [4].

Normalmente estos pequeños servos funcionan con una tensión de alimentación de 5V y a su vez la señal digital de control puede ser de un nivel de tensión entre 3V y 7,2V, pero el valor de voltaje recomendable para evitar errores de detección del “1 lógico” es de 5V. El control se realiza mediante una señal modulada por ancho de pulso, en la que el ancho el pulso indica el ángulo que se desea que adopte el eje. La modulación por ancho de pulso, PWM (Pulse Width Modulation), es uno de los sistemas más empleados para el control de motores. Este sistema consiste en generar una onda cuadrada en la que se varía el tiempo que el pulso está en el nivel alto, manteniendo el mismo período (normalmente), con el objetivo de modificar la posición del servo según se desee (ver figura 5).

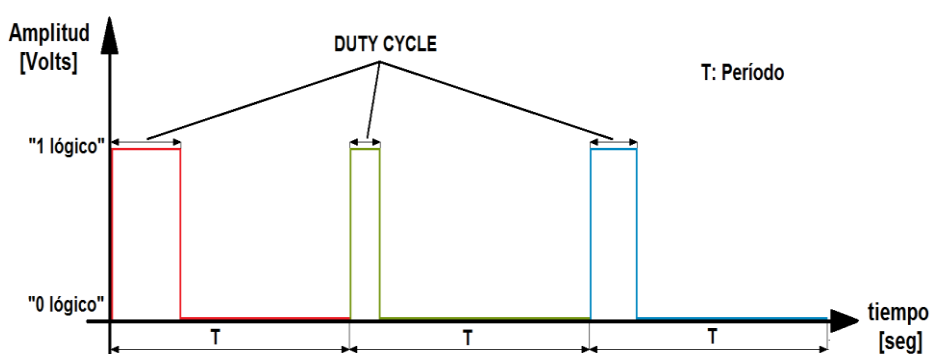


Fig. 5. Ejemplo de modulación por ancho de pulso (PWM)

Por lo tanto, se enviará un tren de pulsos por la entrada de control del servomotor tal que irá variando la duración (anchura) del pulso activo en función al ángulo de giro deseado del motor. Los pulsos donde la señal de PWM se mantiene en “1 lógico” se denomina “Duty Cycle”.

Se necesitan valores específicos de frecuencia en la señal de control. Los valores más generales se corresponden con períodos de la señal cada 20 ms, y pulsos en nivel alto de entre 1 ms y 2 ms de anchura. En otras palabras, se trataría de una señal de 50Hz de frecuencia con un duty cycle entre el 5% y el 10% del período.

Habitualmente cada servo tiene sus márgenes de operación, que se corresponden con el ancho del pulso máximo y mínimo con el que el mismo funciona. Mayormente pueden moverse entre un ángulo de 0° y 180°, aunque existen algunos modelos comerciales que permiten un giro de 360°. En el caso del modelo SG90 permite un ángulo de giro de 180°, con lo cual con un duty cycle de 1.5 ms indicaría la posición central o neutra (90°), con otro de 2 ms giraría 180°, con otro de 1 ms se movería a la posición de 0°, y mientras que otros valores del pulso lo dejarían en posiciones intermedias. En la figura 6, se demuestra gráficamente lo explicado.

Estos valores suelen ser los recomendados, sin embargo, es posible emplear pulsos menores de 1 ms o mayores de 2 ms, para poder conseguir ángulos mayores de 180°. Pero en la práctica esto no se corresponde con la teoría, ya que existen límites mecánicos propios del servomotor, que si se sobrepasan el servo comenzará a emitir un zumbido ya que estará ejerciendo una fuerza para intentar girar un ángulo mayor a 180° que el factor limitante mecánico no se lo permitirá. Es por ello que se debe evitar enviar señales de control de estas características para evitar problemas tales como:

pérdida de precisión de giro o desgaste mecánico producido más rápidamente de lo estimado.

La duración entre pulso y pulso en que la señal de PWM se encuentra en nivel bajo ("0 lógico") no es crítica, e incluso puede ser distinto entre uno y otro período de la señal. Generalmente se suelen emplear valores de período de la señal de aproximadamente 20 ms (entre 10 ms y 30 ms). Si el tiempo de duración del pulso en nivel bajo es inferior al mínimo, puede interferir con la temporización interna del servo, causando un zumbido, y la vibración del eje de salida. Si es mayor que el máximo, entonces el servo pasará a estado dormido entre pulsos. Esto provoca que se mueva con intervalos pequeños.

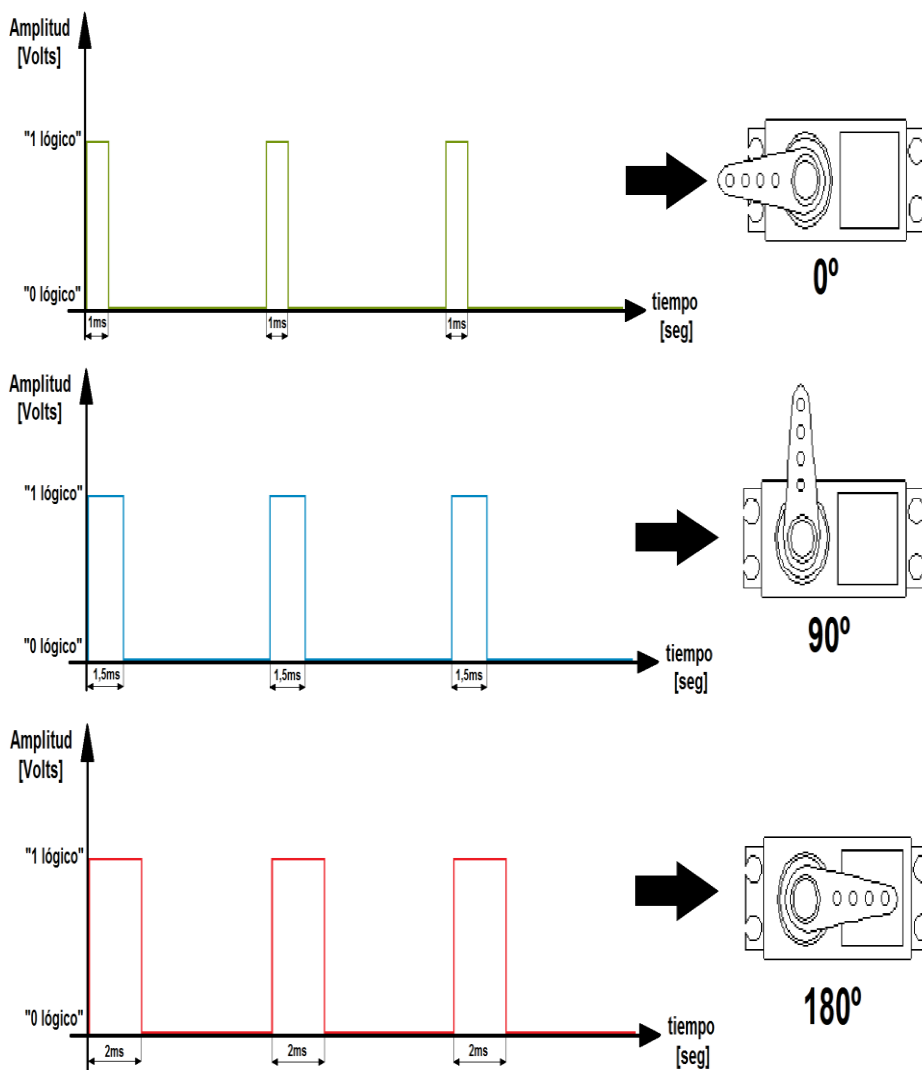


Fig. 6. Ángulos de Giro del ServoMotor

Es importante destacar que para que un servo se mantenga en la misma posición durante un cierto tiempo, es necesario enviarle continuamente el pulso correspondiente. De este modo, si existe alguna fuerza que le obligue a abandonar dicha posición, este intentará resistirse. De igual manera, si se deja de enviar pulsos (o el intervalo entre pulsos es mayor que el máximo) entonces el servo perderá fuerza y dejará de intentar mantener su posición, de modo que cualquier fuerza externa podría desplazarlo.

Un servo tiene un conector de 3 hilos, Alimentación de 5V (rojo), GND (negro o marrón) y el otro Control (amarillo o blanco). Además como se mencionó anteriormente funcionan con niveles de tensión de la señal de control entre 3V y 7,2V, por lo tanto esto nos permite utilizarlos fácilmente en placas de Arduino o Raspberry, ya que la primera trabaja con niveles de tensión de 5V y la segunda con 3,3V.

A continuación se presenta un fragmento de código en python que permite mover los servomotores cambiando la posición de la cámara.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(21, GPIO.OUT)
p = GPIO.PWM(21, 50)
p.start(7.5)
try:
    while True:

        p.ChangeDutyCycle(4.5)
        time.sleep(0.5)
        p.ChangeDutyCycle(10.5)
        time.sleep(0.5)
        p.ChangeDutyCycle(7.5)
        time.sleep(0.5)
except KeyboardInterrupt:
    p.stop()
GPIO.cleanup()
```

El servomotor utilizado es el modelo SG90, en la hoja de especificación de datos del fabricante puede observarse sus características, algunas de ellas se resumen en la tabla 2.

Tabla 2. Características del ServoMotor seleccionado (SG90)

Peso	9 g
Dimensiones	22.2 x 11.8 x 31 mm aprox.
Longitud del cable de Conector	24.5cm
Torque	1.8 kg / cm (a un voltaje de operación de 4.8V)
Velocidad de Operación	0.1 seg / 60° (a un voltaje de operación de 4.8V)
Voltaje de Operación	3.0V ~ 7.2V (valor recomendado 5V)
“Dead Band Width”	10 μs

Rango de Temperatura	-30°C ~ 60°C
Ángulo de Rotación	180°
Ancho de pulso	500-2400 μs

“Dead Band Width” significa Ancho de Banda Muerto y es el tiempo máximo hasta el cual el servomotor no detecta si hay una señal activa en su entrada de control. Por ejemplo, si se enviase una señal de control con un pulso activo de 1,5ms, entonces el servo se movería a su posición central (90°). Si ahora el pulso activo fuese de 1,5ms + 5μs, el servo seguiría en la misma posición que antes, es decir que no se moverá hasta que el pulso activo sea de 1,5ms ± 10μs.

Dead Band Width es un parámetro utilizado para evitar que el servo detecte señales de muy bajo duty cycle, ya que las mismas podrían tratarse de ruido y esto provocaría que el mismo se moviese sin haber enviado ninguna señal de control.

3 Desarrollo de Software de Video Conferencia

La raspberry pi tiene una aplicación de consola que permite capturar fácilmente el video de la cámara incorporada. Esta aplicación se denomina raspivid [5] y soporta numerosos parámetros de configuración entre los cuales se encuentra la calidad del video a capturar, los cuadros por segundo y la posibilidad de enviar el resultado a un archivo o a la salida estándar. Este último parámetro de configuración es justamente el que permite capturar el video desde una aplicación al enviarlo a la salida estándar.

Por ejemplo desde Java se puede utilizar la clase Process y asignarle al mismo la ejecución del comando raspivid con los parámetros deseados:

```
Process procesoRaspivid = Runtime.getRuntime().exec("raspivid -n -b 15000000 -w 1920 -h 1080 -fps 30 -t 0 -o -");
```

Donde se invoca al comando raspivid con los parámetros presentados en la tabla 3.

Tabla 3. Parámetros del comando raspivid

-w 1920	Para establecer el ancho del video en 1920 pixel
-h 1080	Para establecer el alto del video en 1080 pixel
-b 15000000	Para establecer el bit rate a 15Mbit/s
-n	Evita que muestre una ventana con el video capturado
-fps 30	Establece la captura a 30 cuadros por segundo
-t 0	Captura el video indefinidamente
-o -	Envía el resultado a la salida estándar

Luego se puede capturar el flujo de video generado por el proceso mediante el comando:

```
InputStream flujoVideo;
flujoVideo = procesoRaspivid.getInputStream();
```

Al flujo de video capturado se lo divide en paquetes de información para que puedan ser enviados por la red:

```
byte[] datosVideo = new byte[4096];
while (flujoVideo.read(datosVideo) != -1)
{
    //envío de los paquetes por la red
}
```

El tamaño de los paquetes a enviar va a depender de la resolución del video y la cantidad de cuadros por segundo. Existen numerosos autores que trabajan sobre ese tema, entre ellos podemos nombrar [6], [7], [8]. Además puede adaptarse el tamaño de los paquetes de acuerdo al tiempo que tardan los paquetes en llegar a destino según la conexión particular de cada caso.

Para el manejo de las comunicaciones pueden utilizarse las bibliotecas estándar de java o puede facilitarse el proceso incorporando bibliotecas externas por ejemplo Eneter Messaging Framework [9] que es un framework para comunicación entre procesos multiplataforma gratuito para uso no comercial.

En la figura 7, se puede ver la solución en funcionamiento conectada a un monitor con HDMI.



Fig. 7. Conectividad entre dos nodos en laboratorios de la Universidad

4 Conclusiones

Los beneficios de esta solución residen principalmente en la autonomía, que puede ser conectada por HDMI a una pantalla y manejada en forma remota con un teclado de smart tv. Crear esta solución requiere considerar diversos aspectos, que integran no sólo cuestiones de hardware o desarrollo de la aplicación, sino también considerar un

diseño de la interfaz general que le permita al usuario de forma amigable manejarse con el sistema. Actualmente se cuenta con tres nodos idénticos implementados, sobre los cuales en una memoria SD se ejecuta una versión puntual del sistema de video conferencias para IPv4 y se está trabajando en otra versión que incorporará soporte a IPv6. Aún queda por delante trabajar en poder manejar el sonido según lo lejos o cerca que se encuentra el orador, permitiendo automáticamente variar el volumen para obtener un volumen final adecuado, también se continúa trabajando en mejorar el seguimiento del orador por medio de los valores arrojados por el sensor de ultrasonido.

Referencias

1. Cisco, "Cisco TelePresence SX20 Quick Set"
<http://www.cisco.com/c/en/us/products/collaboration-endpoints/telepresence-sx20-quick-set/index.html>
2. Polycom "HD Video Conferencia"
<http://www.polycom.com/hd-video-conferencing.html>
3. Raspberry. (2016) "Raspberry Pi 2 Model B".
<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
4. del Campo García Mariano, "Programar posiciones en un Micro Servo Tower Pro SG90 9G". (2016)
<http://miarduinounotiencunblog.blogspot.com.ar/2016/01/programar-posiciones-en-un-micro-servo.html>
5. Raspberry. Raspivid.
<https://www.raspberrypi.org/documentation/usage/camera/raspicam/raspivid.md>
6. B. Q. Zhao, J. C. S. Lui and D. M. Chiu, "Exploring the optimal chunk selection policy for data-driven P2P streaming systems," 2009 IEEE Ninth International Conference on Peer-to-Peer Computing, Seattle, WA, (2009), pp. 271-280. doi: 10.1109/P2P.2009.5284548
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5284548&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D5284548
7. N. Ramzan *et al.*, "Peer-to-peer streaming of scalable video in future Internet applications," in *IEEE Communications Magazine*, vol. 49, no. 3, pp. 128-135, March (2011). doi: 10.1109/MCOM.2011.5723810
<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5723810&isnumber=5723785>
8. JOHN Samson Mwela, OYEKANLU Emmanuel Adebomi. Impact of Packet Loss on the Quality of Video Stream Transmission. (2010)
[http://www.bth.se/com/mscee.nsf/attachments/6182_Thesis_Report_pdf/\\$file/6182_Thesis_Report.pdf](http://www.bth.se/com/mscee.nsf/attachments/6182_Thesis_Report_pdf/$file/6182_Thesis_Report.pdf)
9. ENETER Messaging Framework. Cross-Plataform Framework for Interprocess Communication <http://www.eneter.net/>

Preliminary results using Snorocket to detect errors in the post-coordination of SNOMED CT

Pilar Avila¹, Laura Gambarte¹, Hernán Berinsky¹, Sonia Benítez¹,
Hee Park¹, Daniel Luna¹, David Pérez¹, Fernando Campos¹,
Sofia Zanetti¹, Soledad Díaz¹, and Alejandro López²,

¹ Departamento de Informática en Salud, Hospital Italiano de Buenos Aires
mariaad.avila@hospitalitaliano.org.ar

² TermMed, Buenos Aires

alejandro.lopezosornio@gmail.com

Abstract. We present preliminary results for the application of a procedure that detects and corrects errors in concept definitions of a local interface vocabulary with SNOMED CT as its reference vocabulary. Using the relations inferred by SNOROCKET we detected redundant fully defined concepts, but also we detected suspected patterns where concepts had redundant inferred relations. Our procedure detected errors in 1.63% of the whole vocabulary, the primary type of error was produced by duplications since these concepts did not exist when the knowledge modeler asserted them. Using these results, we implemented a GUI to track patterns and correct errors. Our procedure contributes to the quality assurance of our local interface vocabulary since errors in the hierarchies can compromise interoperability and meaningful use of the vocabulary. Our approach could be used by thesaurus implementers to detect suspected patterns, grouping them, and offer a centralized interface to correct them.

Keywords: Classification; Controlled Vocabulary; Systematized Nomenclature of Medicine; Quality Control; Knowledge representation; User-Computer Interface

1 Introduction

SNOMED CT is a clinical terminology that allows storage and retrieves healthcare information based on its meaning[1], information definitions are build using their semantic relations. When a single concept is not enough to define the information is possible to build a new one using post-coordination, understood as the representation of a clinical meaning using a combination of two or more SNOMED CT concept identifiers[2]. The Hospital Italiano de Buenos Aires (HIBA) has a local interface vocabulary where each term is mapped via a direct link or a post-coordinated expression to SNOMED CT as its reference vocabulary. The local interface vocabulary was implemented from 2002, and by 2016 already had 520,000 post-coordinated concepts in its terminology system, this system gives services to the

Hospital Italiano Healthcare facilities as well as other health care institutions in Argentina, Uruguay, and Chile.

A major benefit of the local interface vocabulary is its big size and coverage but also is the biggest obstacle for its use and maintenance. It's a requirement for a terminology system that the represented knowledge should be faithful to reality and its quality must be assured, especially in concepts that are post-coordinated for other concepts because the information is propagated whether is right or wrong, therefore a single relation could have a broad impact and unintentional effects[3]. In order to facilitate the inconsistency detection, classifiers are used to obtain a formal representation that provides the explicit semantics of the represented knowledge[4].

A description logic classifier is used here to process all SNOMED CT concepts and HIBA post-coordinated expressions, and based on their definitions organizes the concepts into hierarchies[1]. The results are used to detect redundant concepts and two types of suspicious error definitions. Between the available classifiers, SNOROCKET was selected, since it was specifically designed to be efficient with the logic descriptor used by SNOMED CT[5].

This article analyzes preliminary results for applying a procedure to detect errors in the definitions of the concepts through their inferred form, and to correct them in an iterative cycle of quality assurance. The large size of the local interface vocabulary require a combination of automated and manual checks and reviews, our contribution is a computational approach that helps the users(knowledge modelers) with the audit task in errors automatically found. This approach could be extended to identify more patterns and be presented to users in a consistent GUI, as well as to be applied in others thesaurus with formal definitions.

2 Background

SNOMED CT hierarchies are formulated using a subset of first-order logic known as “description logics” that specifies their semantics. SNOMED CT hierarchies are comprehensive and universal. All and only concepts satisfying the definition of a higher-level “ancestor”, concepts are classified under it as “descendants”, and all the definitions that apply to a concept also apply to all of its descendants.[3]

When new definitions are created, they must be formulated in a description logic in two steps[3]:

- Modelers assert manually each new concept using SNOMED CT as reference vocabulary, creating a “stated form” with defining relations to SNOMED CT concepts.
- A “classifier” software organizes the concepts logically into hierarchies based on their stated definitions; the result is an “inferred form” of the concepts.

Quality assurance is an important part of a terminology's lifecycle, SNOMED CT has been specially studied because of its importance. Many techniques have been proposed for identifying errors[6], these could be summarized in 3 groups: (a) evaluating the semantic completeness[7], (b) from an ontological and logical perspective[8-10], (c) and using Description Logic modeling and concept classification[1,3]. We will rely on the latter approach in this paper.

An important feature of description logics is they allow concepts to be either fully or partially defined. To be fully defined means to be defined by necessary and sufficient conditions, in the case of partially defined, called “primitive” by SNOMED CT, is only defined by necessary conditions[1]. Consequently, only with fully defined concepts is possible to find redundant concepts using a classifier, this redundancy occurs because the inferred form of the post-coordinated concepts is equivalent or because the concept already exists in the reference vocabulary and it should be pre-coordinated instead of post-coordinated.

Using the inferred form is possible to find another type of errors, SNOMED CT publishes descriptive statistics in a draft[11] and implements in its GitHub repository [12]the detection of 22 patterns that could be considered as suspected error definitions. In the beta version of this procedure is detected a general pattern as suspected definition that happens when the inferred form has at the same time two relations to concepts where one is subsumed by the other. In a correct classifier execution, the most general relation should be deleted, and only the most specific should appear. These errors were classified into two types:

- By group interchanges occurs between general and specific relations. For example, figure 1 shows a graphical comparison of the stated and inferred form for “Right main coronary artery thrombosis”. Here the finding site “Right coronary artery structure” is in the same group with the associated morphology “Thrombus”. There is also another group with a more general finding site “Coronary artery structure” but a more specific associated morphology “Occlusive thrombus”. This type of error is called crossover.
- When is inferred a more specific attribute relation. In the case of the example of “Allergy to antiseptic agent” that is shown in the figure 2, the inferred form has a causal agent more specific “Antibacterial agent” that the one asserted in the stated form “Anti-infective agent”. There is a suspected error pattern because the attribute relation was inherited from a concept that has the most specific definition.

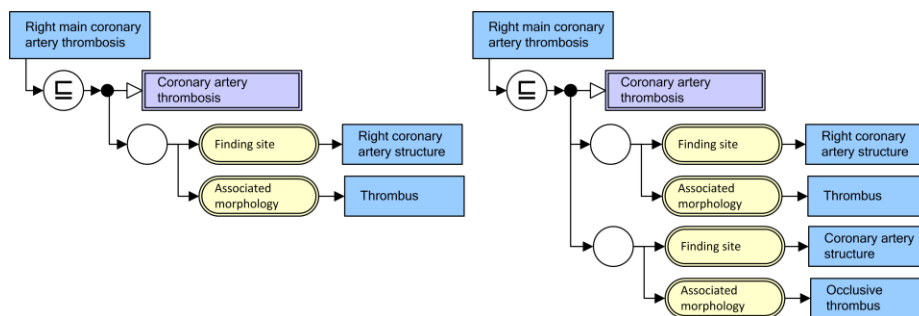


Fig. 1. Stated (left) and inferred (right) model comparison of the concept “Right main coronary artery thrombosis”.

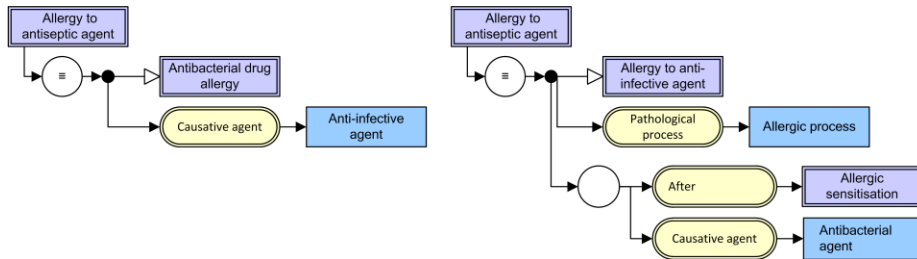


Fig. 2. Stated (left) and inferred (right) model comparison of the concept “Allergy to antiseptic agent”

3 Materials and methods

This study uses the SNOMED CT Spanish edition available on 31 October 2015. This edition has 310,000 concepts defined by 1530,000 relations, and the HIBA's local interface vocabulary that has 520,000 post-coordinated concepts defined by 830,000 relations.

The classification task is made using SNOROCKET, which is the SNOMED CT's preferred classifier[3,13] This classifier returns two outputs: (1) equivalent concepts, to help to deduct which concepts should be pre-coordinated and which ones should be merged into a single one; (2) inferred form, useful to detect suspected error patterns in the definitions.

The implemented method is schematized in the figure 3, and consisted of a cycle of detection and correction error for achieving quality assurance of the local interface vocabulary of HIBA:

1. Modelers define the concept's stated form.
2. SNOROCKET is executed. The output is used to detect suspected error definitions and redundant concepts.
3. Modelers identify which concepts should be merged and which ones should be redefined.
4. Modelers manually correct detected errors in the concept's stated form.

A platform built “in-house” with a GUI was developed to do the last two steps of the algorithm, here modelers can track down errors to the source and merge the concepts keeping the referential integrity in the terminology system.

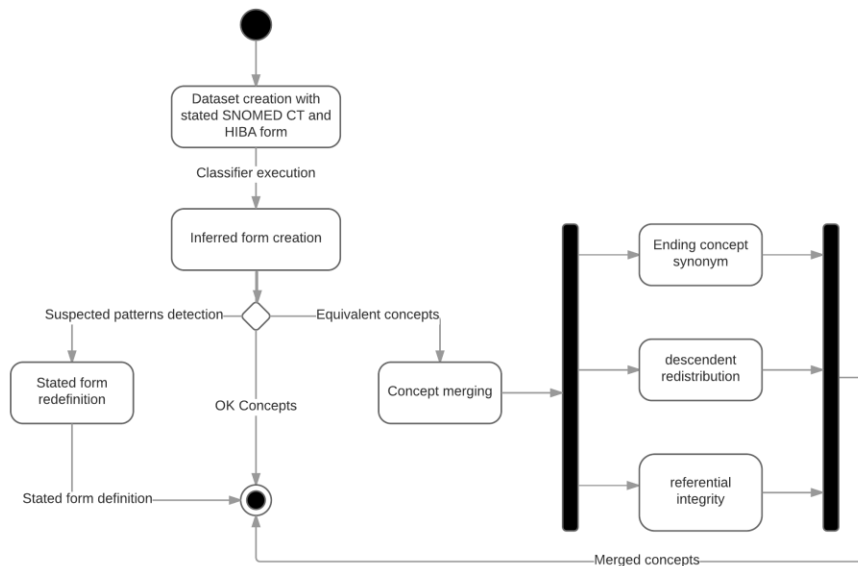


Fig. 3. Implementation procedure

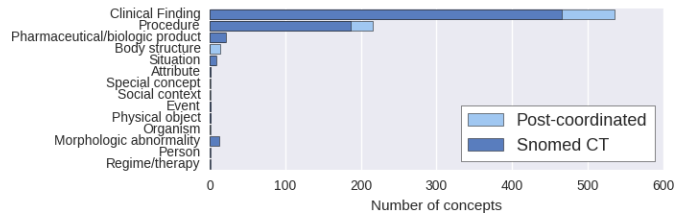
4 Result analysis

After the classifier was run, suspicious patterns were detected in 8,532 HIBA's concepts, representing 1.63% of the whole local interface vocabulary. Of these patterns, 85.62% were redundant concepts, 12.31% were crossover type, and the remaining 2.07% were patterns where the inferred form has relations of type attribute more specific than the stated form.

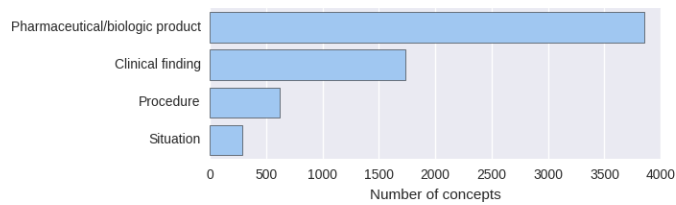
As it can be seen in the figure 4, in both cases the redundant concepts focus on clinical finding, procedure, and pharmaceutical/biologic product hierarchies. These three hierarchies are the 88.82% of the total of all the post-coordinated concepts. There were two types of redundant concepts:

1. Concepts that should be pre-coordinated since already exists a single SNOMED CT's concept that defines the HIBA's concept entirely.
2. Post-coordinated concepts that were equivalent and must be merged or redefined.

Only in the fully defined concepts is possible to detect redundancies, and must have been corrected manually by a knowledge modeler. In the case of primitive concepts, it was used two criteria to detect patterns with suspected errors. The pattern called crossover were focused on concepts from the clinical finding hierarchy; and the concepts of the pattern where the inferred form had relations of type attribute more specific than the stated form, were uniformly distributed in the three major hierarchies previously mentioned, as it is shown in the figure 5.

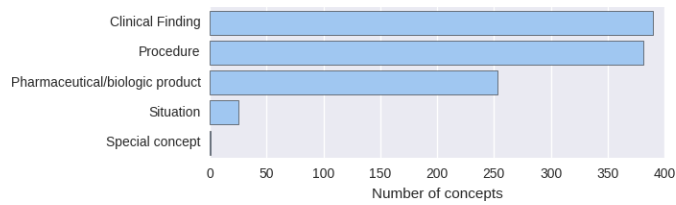


(a) Concepts that must be pre-coordinated

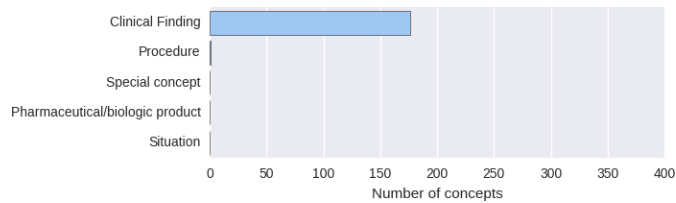


(b) Concepts that must be merged

Fig. 4. Redundant concepts



(a) Concept with inferred form that had relations of type attribute more specific than the stated form



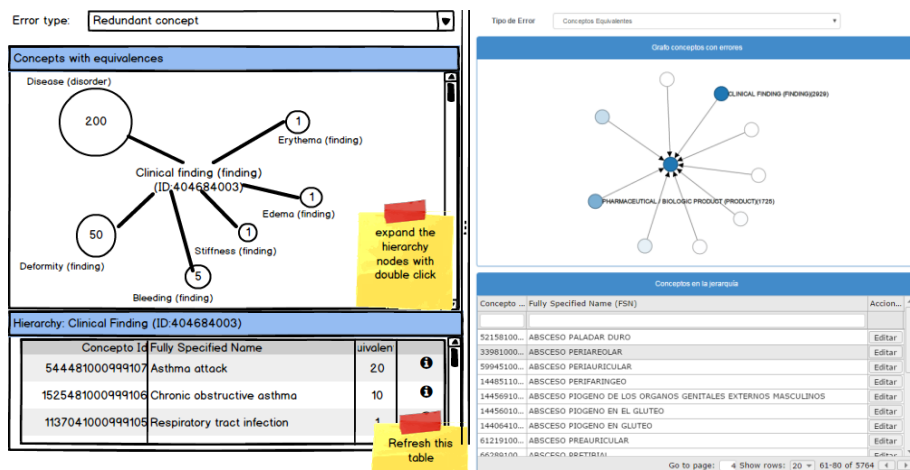
(b) Concepts with crossover

Fig. 5. Concepts with suspected error patterns

From the results obtained with SNOROCKET, we designed software that allows correcting errors following top-down and bottom-up strategies. The top-down strategy allows getting all the auditable concepts from an ancestor. In this way, for example, pharmacology, procedure or clinical finding experts can retrieve the concepts they want to correct or navigate to more specific hierarchies. Figure 6 shows how to navigate through the hierarchies and filter the concepts using the GUI. Afterward, when is selected a concept, a modeler can execute the actions to amend the errors.

With a bottom-up strategy, modelers start in the concept that must be corrected. The tool has three types of views, as it is shown in figure 7, stated, inferred and graph view. In the stated view, the concepts are defined using relation that was assorted manually by a modeler; the inferred view is the SNOROCKET's output after the concepts were classified base on their logical definitions; graph view is used to understand the inferences source. In the graph, nodes represent SNOMED CT concepts, and edges represent relations, modelers can expand the nodes to get their ancestors and track the source of unexpected inferences.

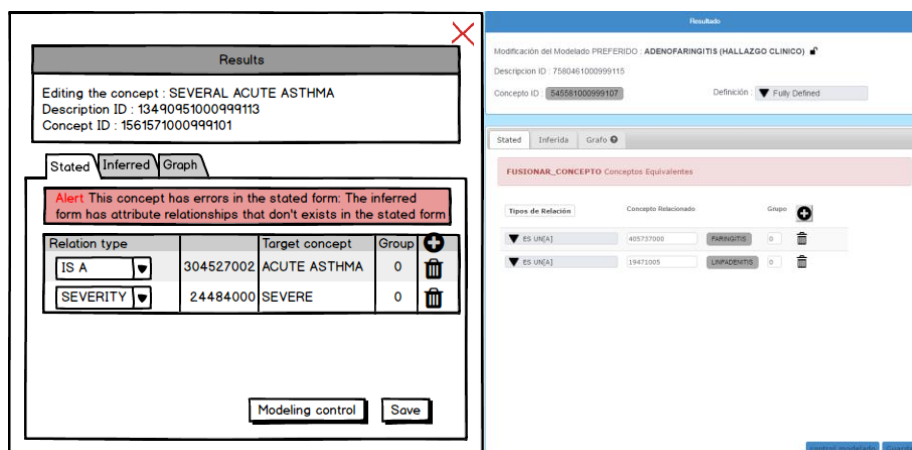
Two types of corrections can be made: 1) to change the definition in stated view 2) to merge concepts when equivalence with others concepts exists. This last case requires defining the new concept's synonym and keeping the referential integrity in the terminology system.



(a) Hierarchy navigator

Fig. 6. GUI to filter concepts with suspected error patterns, mockups(left) and implementation(right)¹

¹ See all the figures at <https://goo.gl/photos/zMN87ckguYzktz8Q7>



(a) Stated view

Fig. 7. GUI with different views from a concept that must be corrected, mockups(left) and implementation(right)²

5 Discussion

In this paper, we described a procedure to detect and correct errors in the definitions of the concepts from the local interface vocabulary of HIBA, after a classifier was applied. The classifier returns an inferred form for each concept and the equivalent concepts. From this output, software was designed to group concepts with suspected patterns in a centralized interface, that modelers can merge concepts or track and correct the error in the stated view.

In previous studies, using SNOMED CT Editorial Guide directives showed that 3.67% of the most used local interface vocabulary concepts at HIBA were not represented correctly[14]. Using the relations inferred by a classifier, the procedure detected 8,532 errors in the post-coordinated concepts, this is 1.63% of the whole vocabulary.

Detected errors were limited to the redundancies found by SNOROCKET and the patterns that are suspected for errors mentioned for SNOMED CT's draft[11] and GitHub[12]. In this procedure, we implemented two types of patterns that had the biggest probability of being errors, crossover type and when is inferred a more specific attribute relation type. As a consequence of the SNOMED CT updating, the most common errors were produced by redundancies since these concepts did not exist when were asserted by the modeler.

This procedure described in this paper was designed as a part of an integral cycle of quality assurance, and the GUI were addressed to ease the track of error and its source. To achieve this goal, the suspected patterns were divided by types and then for the concept was suggested the action that removes the detected problem.

² See all the figures at <https://goo.gl/photos/jmqMuP6aLvtUE9bQA>

Although the procedure is used in a local interface vocabulary in the field of medicine, it is applicable in another domain thesaurus with formal definitions as well.

6 Conclusion and future work

This paper shows 3 types of errors that can be detected in the definitions of the concepts from a local interface vocabulary using the relations inferred by a classifier. Even though these redundancies and errors unnecessarily impede the work of modelers, the manual search for the causes of the errors can be a tedious task[15]. On the other hand, keeping an interface vocabulary with these errors can propitiate the spread of errors, compromise the interoperability and meaningful use[3]. For example, when the concept with errors was in the internal nodes into the hierarchy the errors are propagated in cascade; or when artificial intelligence is used to automatically make the assertions can compromise the new assertions since the manual assertions are used as input to create models.

The software created to help modelers to correct errors is in its beta version and has not been extensively tested. In further developments, the software is going to be tested using usability metrics to ease the tracking of error in the stated form keeping a design oriented to minimize errors and their impact. As this approach is scalable to be implemented with other types of errors, we are going to implement the detection of all the suspected definitions described by SNOMED CT and measure their recall and precision.

Overall, it can be concluded that the application of this procedure contribute to pointing out where the errors happen and offer a centralized interface to correct them. It's a step to improve the quality of our local interface vocabulary.

References

1. Alan Rector and Luigi Iannone. Lexically suggest, logically define: quality assurance of the use of qualifiers and expected results of post-coordination in SNOMED CT. *Journal of biomedical informatics*, 45(2):199–209, April 2012.
2. T. Benson. *Principles of Health Interoperability HL7 and SNOMED*. Health informatics. Springer London, 2009.
3. Alan L Rector, Sam Brandt, and Thomas Schneider. Getting the foot out of the pelvis: modeling problems affecting use of SNOMED CT hierarchies in practical applications. *Journal of the American Medical Informatics Association : JAMIA*, 18(4):432–40, January.
4. Ronald Cornet and Ameen Abu-Hanna. Auditing description-logic-based medical terminological systems by detecting equivalent concept definitions. *International Journal of Medical Informatics*, 77(5):336–345, 2008.
5. Michael Lawley Ro Metke-jimenez. Snrocket 2.0: Concrete Domains and Concurrent Classification.
6. J E Rogers. Quality assurance of medical ontologies. *Methods of information in medicine*, 45(3):267–74, 2006.

7. Guoqian Jiang and Christopher G Chute. Auditing the semantic completeness of SNOMED CT using formal concept analysis. *Journal of the American Medical Informatics Association : JAMIA*, 16(1):89–102.
8. Stefan Schulz, Boontawee Suntisrivaraporn, Franz Baader, and Martin Boeker. SNOMED reaching its adolescence: ontologists' and logicians' health check. *International journal of medical informatics*, 78 Suppl 1:S86–94, April 2009.
9. S Schulz, S Hanser, U Hahn, and J Rogers. The semantics of procedures and diseases in SNOMED CT. *Methods of information in medicine*, 45(4):354–8, 2006.
10. Stefan Schulz, Udo Hahn, and Jeremy Rogers. Semantic Clarification of the Representation of Procedures and Diseases in SNOMED((R))CT. *Studies in health technology and informatics*, 116:773–8, 2005.
11. International Health Terminology Standards Development Organisation. Snomed CT Quality Evaluation. http://qa.snomed.org/docs/SNOMEDCT_Quality_Evaluation_Descriptive_Release_Statistics_20160219_v1_00.pdf, 2016. [Online; accessed 07-July-2016].
12. International Health Terminology Standards Development Organisation. SNOMED CT Pattern and Changes Evaluation Toolkit. <https://github.com/IHTSDO/sct-statistics-qa/>, 2016. [Online; accessed 07-July-2016].
13. Michael J Lawley and Cyril Bousquet. Fast classification in protégé: Snorocket as an owl 2 el reasoner. *Proc. 6th Australasian Ontology Workshop (IAOA'10). Conferences in Research and Practice in Information Technology*, 122:45–49, 2010.
14. H Navas, A Lopez Osornio, G Elias Leguizamon, N Orrego, SWassermann, D Luna, et al. *Implementación de reglas para el control de modelado con SNOMED CT*. INFOLAC, 2008.
15. Kathrin Dentler and Ronald Cornet. Redundant Elements in SNOMED CT Concept Definitions. pages 186–195. Springer Berlin Heidelberg, 2013.

An Expert-Driven Approach for Collaborative Knowledge Acquisition

Leo Ordinez^{§1}, Rodrigo Santos^{‡2}, Damián Barry^{§3}, Sergio Firmenich^{§4},
Alejandro Sánchez^{§5}

[§]Laboratorio de Investigación en Informática (LINVI), Departamento de Informática,
UNPSJB, Bvd Brown 3050, 9120, Puerto Madryn, Chubut, Argentina

[‡]Dep. de Ing. Eléctrica y Computadoras, Instituto de Inv. en Ingeniería Eléctrica,
UNS-CONICET, Av. Alem 1253, 8000 Bahía Blanca, Argentina

¹leo.ordinez@gmail.com - ²ierms@criba.edu.ar - ³damian_barry@unpata.edu.ar -
⁴sergio.firmenich@lifia.info.unlp.edu.ar - ⁵asanchez@unsl.edu.ar

Abstract. Knowledge management is key for any organization. Huge amount of data is made available to organizations by pervasive technologies such as smart mobile devices. However, the knowledge to use such data is still missing, and organizations typically fail to exploit it. This paper proposes an architectural design that aims at addressing such problem. It focuses on knowledge management for collaborative systems in which complex and multicausal situations are presented to interacting actors on a large geographical area with possible low connectivity.

1 Motivation

In terms of information and communications technology (ICT), an effective information system is one that provides users with the necessary information to meet environmental requirements [1]. Web technology worldwide and electronic networks have created an environment where the physical location is not important. From the last two decades, advances in ICT have created the conditions to integrate and coordinate information from different actors and build knowledge from them. At this point it is important to distinguish between data, information and knowledge [2]. Data are facts of the physical world obtained through observation. The information adds to those data the necessary context to obtain a meaning of them. Although the information may be used to make decisions, by itself is not enough. It is required prior learning and experience to interpret this information (*i.e.*, contextualized data). This context, together with the corresponding information is knowledge, which allows to constitute a judgment on reality and take actions on it.

The purpose of this paper is the proposal of a reference model for describing and implementing collaborative systems based on ubiquitous computing. The primary motivation of this proposal is based on the field of application of the systems to be developed, which have to do with complex, multi-causal, possibly extraordinary situations involving the interaction of different actors, in a large geographical area, with potentially low level of connectivity. In a different way,

these systems must interact with unfavorable contexts both in its technological infrastructure and its environmental particularities. The work focuses on the collection of data in a non-urban terrain, with sporadic visitors, like Patagonian region. In this regard, transformation of the data into information, will allow the building and extraction of knowledge of the Patagonian region. It is worth noting that in the context of this research knowledge is represented through statistical indicators that will support a wise decision making.

It is important to remark that the software application, for the previously described domain, are intended to be specified by *domain experts*, possibly with no computing skills. This is, according to the variability of the potential scenarios, it is necessary to count with a certain degree of experience in order to manipulate the collected data. In this regard, a domain expert is appropriate for this task. However, in general, the expert will not have sufficient computing or programming skills to develop a complete system, as it would a *computer expert*. Consequently, the reference model proposed should consider this aspect and provide the flexibility and mechanisms to allow an expert user to be able to define the software application.

In the here presented approach, collaborative systems are meant for tackling situations in large, heterogeneous and even unpredictable contexts. As mentioned above, a main actor is the *Domain Expert*, which is interested in obtaining information from the environment. Furthermore, he is the one who should define the software application in order to meets his requirements. Nonetheless, another necessary actor present in the system is the one in charge of collecting data, the *Data Collector*. This is an actor possibly inexperienced and yet with no particular interest in the information obtained, but in the process of gathering that information.

With all the contribution of this paper is the proposal of an architecture for collaborative systems, addressed to expert users of a particular domain, who require to manage information in order to make decisions that:

- P1. Allow *Domain Experts* define the data they were seeking and the way in which these data are converted into displayable information automatically (no third-party development, only involve the intervention of the expert or team of experts in the domain).
- P2. Allow changing the above definition at any time, due to volatile information requirements, which emphasizes the variability of the final applications both for displaying information (for *Domain Experts*) as for the implementation of data collection (for *Data Collectors*).
- P3. Allow data collection regardless of connectivity or other conditions that are already mentioned in the article.
- P4. Allow data sources to be managed in a certain agnostic way, allowing to obtain:
 - (a) Data collected by *Data Collectors*, mainly focused on the use of mobile devices
 - (b) Data from sensors and other such devices
 - (c) Data obtained from repositories

- P5. Allow deriving automatically the mobile data collection applications for use by *Data Collectors*
- P6. Allow users to reward collaborators, for example, by creating a scenario of gamification.

The rest of the paper is organized as follows. In Section 2 a broad review of related works is performed; the proposal of an architecture to develop this kind of systems is done in Section 3; in Section 4 an application case is presented to illustrate the proposal; finally, conclusions are exposed in Section 5.

2 Related Works

Some works propose different levels of standardization of communication protocols to ease the interoperability of heterogeneous systems. A generic framework for cooperative multi-issue one-to-many negotiations is proposed in [3]. The need for a common way to abstract the heterogeneity of devices is outlined in [4], when working on collaboration policy generation technique to govern dynamic collaboration in Service Oriented Architecture (SOA) systems. Dynamic collaboration allows two parties not knowing each other to establish their collaboration protocol at runtime [5]. In [6] the authors employ code generation techniques and use common abstractions, operations and mechanisms at all system levels to support uniform system of systems composition. The complexity of the communication in collaboration system depends on the integration and negotiation of diverse and heterogeneous community societies like companies, governments, universities, ONG, etc. Some of them in the direction of infrastructure towards coordination, switching and choreography of information in heterogeneous systems [3,5,7], others in specifying semantics and standardizing protocols to communicate devices and systems [4,6]. The two previously mentioned alternatives need the concept of data transformation like a graph-based transformation approach proposed in “Towards Choreography Model Transformation via Graph Transformation” [8] or a XSLT transformation model based on XML syntax proposed in “Sociedad de la Información-Switch Transaccional” [7]. The works mentioned above are focused in the description of a particular part of a middleware. They proposed different alternatives to amalgamate heterogeneous messages, transactions or another communication unit more abstract than a packet.

As was mentioned in Section 1, in this paper a distinction is made between a *Computer Expert* and a *Domain Expert*. Following some relevant works oriented to the former are contrasted. In [9] a new interpretation of the Internet-of-Things (IoT) is presented. The authors allow end users to manage the sensing resources populating the involved devices, not just the mere data output they produce, by using virtual devices in the cloud. The model is focused in sensing/actuating devices without human intervention. In this regard, heterogeneity is given by the different kinds of resources (devices) available, which in our proposal would constitute one data source. A similar approach concerning management of heterogeneity is presented in [10] and [11]. The proposal is centered in Cyber-Physical Systems, which report data from sensors to different databases. Although the

architecture proposed is very interesting and could be extended to other data sources, the objective of the papers was to reduce energy consumption by lowering the communication rate. In [12] the authors proposed an architecture similar to the one proposed in this paper. One of the aspects they focused is maintain consistency between transactions. To this, they based their approach in graphs and logic clocks.

Other important aspect of the collaboration system architecture is the adaptive functionality of the components needed to interconnect systems to solve a domain specific problem. In this regard an interesting approach is presented in [13], which provides a generic software and middleware components to ease the design and development of mass market context-aware applications built above the Internet of Things. In [14] the authors made a proposal to ease the development process of Agent-Based Pervasive Systems, providing the user with a set of abstractions that ease the implementation of Pervasive Systems and the deployment of a platform for their execution. Finally, also in IoT application domain, [4] presents a semantic web knowledge representation technologies and interaction with devices and the physical world to achieve access and modifying their virtual representations.

The approaches described in the previous paragraph are centered in a specific domain. In what follows, a set of articles which pursue the objective of being oriented to Domain Experts, are analyzed. A recent work can be found in [15]. The authors report an ontology-based visual query system to specify and extract data of interest for non-technical users. The article is centered in usability and consequently not the architecture nor the middleware are described. In addition, the work does not consider heterogeneous sources in the sense that they are established here. A full architectural description that manages heterogeneous data sources is presented in [16]. Finally, another detailed description of an architecture and its components is proposed in [17]. An interesting contribution is given by a file format used for exchanging multimedia content. Another contribution is the inclusion of a Message agent, that handles message queuing through a “store and forward” mechanism and that translates messages, providing the usable information to upper layer applications.

To sum up, it is important to emphasize the difference between the different approaches presented in this section and the proposal of this paper. In particular, some of the approaches do not consider the expert user as playing a “developer role” (also known as *end-user development*). On the contrary, the user is taken as a passive receiver of the information, but not as the one who set the basis, parameters and design the application to gather that information. Having an *Domain Expert* acting as a developer and counting with an architecture and mechanisms to support that task, extends the number of possible scenarios and domains to consider. Consequently, the experts reduce their dependence on *Computer Experts*, when their information requirements slightly change or new decision scenarios are presented. In addition, another actor plays a key role, the *Data Collectors*, which are not interested in the results nor in the development, but in the process.

3 An Architecture for Collaborative Systems

A Collaborative System is one in which its components are inherently distributed, data is collected from different sources, information is obtained by processing data in a particular context and knowledge is produced to fulfill the requirements of an expert in a particular field of interest. Any entity or role played by a person or another system consisting in gathering data through automatic, electric, mechanical, observed or indirect means, satisfying sources D1., D2. or D3. is considered a *Data Collector*.

As mentioned in Section 1, in general three data sources are considered:

- D1.** Data collected by collaborating users
- D2.** Data from sensors and other such devices
- D3.** Data obtained from repositories

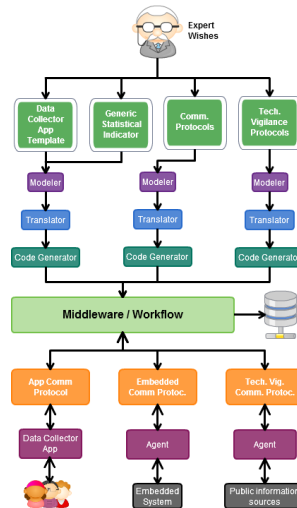


Fig. 1. Architecture for Collaborative Systems based on Ubiquitous Computing.

Figure 1 shows the architecture proposed in this paper for tackling Collaborative Systems. From top to bottom it can be seen how the expert's wishes traverse multiple phases until being concretely implemented. The boxes represent components which perform data transformations or products which are generated by those transformations. The arrows represent the direction of the information flows. The idea behind it is as follows.

The expert in a particular field intends to obtain information from its application domain. That information comes from data that is gathered from multiple sources that is eventually contextualized. The different sources from which it can obtain the information gives the context in a certain level of detail. For example, if information is taken from sensors they must be normalized and electrical

measures turned into physical quantities. On the other hand, if information retrieval from a patents repository is chosen (so technological vigilance protocols are used), the gathered information is already contextualized.

In addition, the expert defines a series of statistical indicators the he wants to get from the information obtained.

Note that the previous paragraphs also establish a workflow for developing a Collaborative Systems. Figure 2 shows a sketch of that workflow.

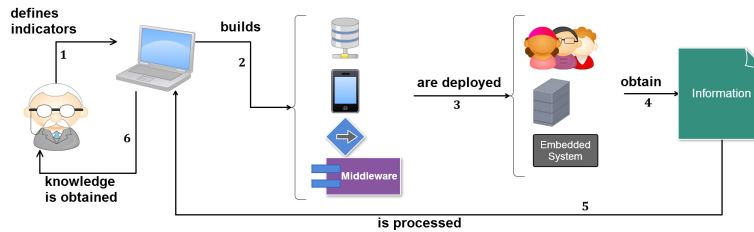


Fig. 2. Workflow for Developing a Collaborative Systems.

To the aims of this proposal, information could be obtained from three main sources. The first one is collected through human intervention by means of a smartphone app (D1.). This app could be as simple as a data collector similar to filling in a spreadsheet; or more complex like a game [18] or citizen science [19] app. The second information source is from an embedded system, in which case the expert has to define the communication protocol with that system (D2.). The third source is constituted by online repositories (D3.), which are accessed through technological vigilance protocols [20].

The former concerns the requirements definitions made by an expert in a particular application domain. In the specific case of this paper, the architecture has to consider some non-functional constraints, which have to do with the environmental conditions of Patagonian region itself. These aspects include communication mechanisms in a wide area, energy-aware devices, data consistency and availability of data and devices, among others.

Independently of the information source, the decision made by the expert is then traversed through a chain of tools that perform transformations on each step resulting in products that feed the next step. In general, the graphical model defined by the expert is then transformed into a syntactical model, which is then transformed into an executable code. The generated code would be downloaded to a smart device, integrated to the embedded system or deployed in a server to retrieve information.

4 Collaborative Knowledge in the Wool Production

In this section, an application case in which the authors are working is presented. It is focused in a particular productive sector in the agriculture field, which is the wool sector.

Currently the process of shearing is done manually and with registration of all information with low-tech methods: analog scales, bale processing cards made by hand, recording of all information made on handwritten forms. In order to establish a set of best practices, the Ministry of Agriculture implemented the PROLANA Program for the purpose of regulating the activity, but mainly to ensure the wool export process, improving international ratings of Argentinian wool. By improving this rating, producers are enabled to obtain better international prices.

PROLANA has detected that the main problems encountered in the shearing process are:

1. Errors in weighing the bale;
2. Failures in registering the burden weighing on the scoresheet wool book;
3. Inability to record the weight of the burden on the individual card thereof;
4. Mismatch between the kind of wool recorded on the card and the one registered on the scoresheet wool book;
5. Registration of kinds of “non-existent” wool;
6. Letter and / or illegible numbers;
7. Forget to include the identification card of each bale;
8. Creating bundles “twins”, by doubling the number of burden;

The actors involved in the shearing process are:

Wool Conditioner: His main purpose is to supervise the overall process of shearing and quality control of wool packaging. PROLANA certifies its work and is responsible of capturing data in field.

Shearing Company: Its purpose is to execute the overall shearing process in the field. PROLANA enables his work and guarantee and regulate their contracts.

Producer: His main purpose is to offer his wool to the association of the wool export system. PROLANA guarantee the quality process control of the shearing and proper stowage and storage of wool until exporting process.

PROLANA: Its main purpose is to guarantee the overall quality process of shearing wool, packaging, stowage and storage, and properly export of it.

In addition to the multiplicity of actors involved in the shearing process, interested organizations such as the Ministry and the Livestock Associations are added. It is therefore necessary to have a platform to work with all the actors mentioned in the first instance and to ensure the reliability of information acquired at different points in the process.

This project seeks to provide sheep producers in the Patagonian region a semiautomatic electronic recording mechanism. This device (commonly a rugged

handheld) can be connected to an electronic scale to automatically record the weight of the bales. In addition, the equipment must be compatible with a printing mechanism which has its own battery and battery charger. This would allow to comply with all registration requirements that PROLANA has and minimizes errors in loading and transferring of information.

The information stored by a team will be loaded in a software specially designed to automatically make the summary of the wool book with electronic signature (Wool Conditioner, Producer, Shearing Company) and then move to print worksheets and their respective summary wool book with the necessary information.

The information will be transmitted to the PROLANA server located in Rawson, the capital city of Chubut. This will allow each wool book information, to be available immediately in the PROLANA's central of information. In addition to the information for the producer, PROLANA builds statistical indicators that allow a conscious decision making process and political planning.

For the particular case of a *D1.-based system*, such as the one presented in this section, the architecture becomes a double case of a 3-tier architecture [21], as it is shown in Figure 3: a global 3-tier, which includes a server for collecting information and processing it; and a local 3-tier, deployed in several smart devices.

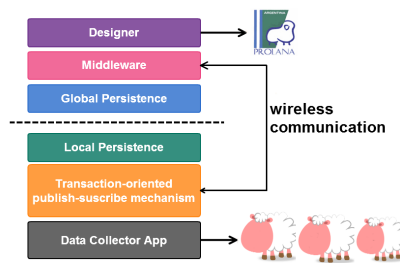


Fig. 3. Two 3-tier architectures.

The main objective is to avoid loading the wrong information about wool production of the Province of Chubut. Nonetheless, PROLANA also uses the information to plan the actions to be taken in the next years. Thus, building knowledge. Note that an inherent consequence of it, is the variability of the information requirements that may exist for the next production of wool, because of the knowledge acquired. This exposes a mutual reliance between the information system and the physical world. To sum up, the presented case shows how the architecture proposed adapts to the following premises enumerated in Section 1: P2., P3., P4.(a), P4.(b), P5. and P6.

5 Conclusions and Future Works

The fact of having a reference model to obtain knowledge from field data is extremely useful in decision making. Knowledge is built from different information sources, which in terms are contextualized data gathered from the field. In the context of this paper, the system must interact with unfavorable contexts both in its technological infrastructure and its environmental particularities. The work focuses on the collection of data in a non-urban terrain, with sporadic visitors, like Patagonian region. These kinds of systems can be classified as Collaborative based on Ubiquitous Computing, since the collection of data is done in an environment of sparse population and, in many cases, by means of embedded autonomous systems.

In order construct that knowledge, a reference model is needed. In this paper, the contribution made is the proposal of an architecture to manage those kinds of collaborative systems. Finally, a representative application case was exposed. The case illustrated the importance and applicability of the proposal done in this paper. Our approach could fit with unexpected and variable scenarios where information could be defined by Domain Experts without computing skills. For instance, urban traffic information, businesses' statistical information or biological observation.

The future works are related to two lines that we are working on. In the first one, we are detailing the proposed architecture under the 4+1 architectural view model, defining formally the model for supporting this kind of systems and applying a consistency protocol for the middleware. Secondly, and considering some good practices of the end-user development literature, we are working in the domain experts tool prototype. With it, he would be able to define the mappings of the model, from establishing communication protocols to setting the data he wants to collect, even when he does not have programming skills.

References

1. El-Bibany, H., Katz, G., Vij, S.: Collaborative information systems. Technical Report TR048, Stanford University (april 1991)
2. Bellinger, G.D.C.A.M.: Data, information, knowledge, and wisdom. (2004)
3. Scafes, M., Badica, C., Pavlin, G., Kamermans, M.: Design and implementation of a service negotiation framework for collaborative disaster management applications. In: 2010 International Conference on Intelligent Networking and Collaborative Systems, IEEE (2010) 519–524
4. Kiljander, J., D'elia, A., Morandi, F., Hyttinen, P., Takalo-Mattila, J., Ylisaukko-Oja, A., Soininen, J.P., Cinotti, T.S.: Semantic interoperability architecture for pervasive computing and internet of things. Access, IEEE **2** (2014) 856–873
5. Tsai, W.T., Huang, Q., Xiao, B., Chen, Y., Zhou, X.: Collaboration policy generation in dynamic collaborative soa. In: Autonomous Decentralized Systems, 2007. ISADS'07. Eighth International Symposium on, IEEE (2007) 33–42
6. Blair, G.S., Bromberg, Y.D., Coulson, G., El Khatib, Y., Réveillère, L., B Ribeiro, H., Rivière, E., Taïani, F.G.D.: Holons: towards a systematic approach to composing systems of systems. (2015)

7. Barry, D.P.: Sociedad de la información-“switch transaccional”. In: XV INTERCON – Trujillo 2008, IEEE (2008)
8. Han, F., Kathayat, S.B., Le, H., Brek, R., Herrmann, P.: Towards choreography model transformation via graph transformation. In: Software Engineering and Service Science (ICSESS), 2011 IEEE 2nd International Conference on, IEEE (2011) 508–515
9. Distefano, S., Merlino, G., Puliafito, A.: A utility paradigm for iot: The sensing cloud. *PERVASIVE AND MOBILE COMPUTING* **20** (2015) 127–144
10. Mundra, A., Rakesh, N., Tyagi, V.: Query centric cps (qcps) approach for multiple heterogeneous systems. *CoRR* **abs/1306.6397** (2013)
11. Mundra, A., Rathee, G., Chawla, M., Rakesh, N., Soni, A.: Transport information system using query centric cyber physical systems (QCPS). *CoRR* **abs/1401.3623** (2014)
12. Ives, Z.G., Green, T.J., Karvounarakis, G., Taylor, N.E., Tannen, V., Talukdar, P.P., Jacob, M., Pereira, F.: The orchestra collaborative data sharing system. *SIGMOD Rec.* **37**(3) (September 2008) 26–32
13. Arcangeli, J.P., Bouzeghoub, A., Camps, V., Canut, M.F., Chabridon, S., Conan, D., Desprats, T., Laborde, R., Lavinal, E., Leriche, S., et al.: Income–multi-scale context management for the internet of things. In: Ambient Intelligence. Springer (2012) 338–347
14. Agüero, J., Rebollo, M., Carrascosa, C., Julián, V.: Mdd-approach for developing pervasive systems based on service-oriented multi-agent systems. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal* **2**(3) (2013) 55–64
15. Soyly, A., Giese, M., Schlatte, R., Jimenez-Ruiz, E., Ozcep, O., Brandt, S.: Domain experts surfing on stream sensor data over ontologies. In: Proceedings of the 1st Workshop on Semantic Web Technologies for Mobile and Pervasive Environments (SEMPER@ESWC 2016). Volume 1588., Heraklion, Greece, CEUR-WS.org (2016)
16. Prasad, S.K., Madiseti, V., Navathe, S.B., Sunderraman, R., Dogdu, E., Bourgeois, A., Weeks, M., Liu, B., Balasooriya, J., Hariharan, A., Xie, W., Madiraju, P., Malladi, S., Sivakumar, R., Zelikovsky, A., Zhang, Y., Pan, Y., Belkasim, S. In: SyD: A Middleware Testbed for Collaborative Applications over Small Heterogeneous Devices and Data Stores. Springer Berlin Heidelberg, Toronto, Canada (2004) 352–371
17. Su, X., Prabhu, B.S., Cheng Chu, C., Gadh, R.: Middleware for multimedia mobile collaborative system. In: in Wireless Telecommunications Symposium, 2004. (2004) 112–119
18. : Online citizen science games: Opportunities for the biological sciences. *Applied & Translational Genomics* **3**(4) (2014) 90 – 94 Global Sharing of Genomic Knowledge in a Free Market.
19. Rogers, D., McCann, T., Cooper, A.: Integrating paper-based habitat mapping with mobile electronic field recording procedures. *Ecological Informatics* **5**(6) (2010) 479 – 483
20. Barry, D., Cortez22, J.M., Aita12, L.I., Stickar, R.: Extracción automatizada para la unidad de vigilancia tecnológica e inteligencia competitiva de la patagonia
21. Eckerson, W.W.: Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications. In: Open Information Systems 10. (January 1995)

Development of a datalogger with open source hardware and software for the study of desert animals

Daniel Díaz^{a,b}, Sandra Oviedo^{a,b}, Ramiro Madriaga^b, Alejandra Otazu^b, Francisco Ibañez^b

^a LIAI- Instituto de Informática- ^bDpto. de Informática FCEFyN - Universidad Nacional de San Juan CUIM - Av. Ignacio de la Roza 590 (O), Rivadavia - J5402DCS San Juan {ddiaz, soviedo, rmadriaga, otazuale, fibannez }@iinfo.unsj.edu.ar

Abstract. The territory of the province of San Juan is composed by 80 percent by mountains. Its climate is quasi desert. It is in this place, where a group of researchers from the San Juan National University, is dedicated to the study of amphibians. To accomplish this task a device that can record data is necessary. These data are some environmental variables and the sound that these animals produce at certain temporal points. Generically to this device type is denominated data logger. This article describes the process of developing a datalogger of technical characteristics that have been specified by biologist researchers. It is a device built with open source hardware and software. The capture and data processing is done with the Arduino platform. The human computer interface is developed using the Android platform. That is, the management of datalogger is carried out by a mobile device, tablet or Smartphone based on Android

Keywords: Innovate products, open source hardware, methodology, datalogger

1 Introduction

A datalogger is an electronic device that records data over time or in relation to location either with a built in instrument or sensor or via external instruments and sensors. The datalogger are often used in biology experiments and researches, as it is case of researchers that are dedicated to study of amphibians. Here, recording some environmental variables and the sound of amphibians (frogs and toads) is very useful for studying taxonomy and systematics of these species. This type of devices are present on the market, but almost all of them have generic functionalities. Also, the cost of datalogger is a problem. A research project needs an important amount of dataloggers. It is impossible accomplish this requirement with the project budget of a typical research project of Argentine universities.

Commonly, the technologies to develop a device such features come from electronic engineering. New technologies have emerged in recent years that allow build such devices. These technologies are known as open source hardware and software platforms[1], [2], [3] y [4]. This paper describes the engineering process that allowed the development of a datalogger to prototype level using these technologies. This

device allows biologists researchers to record the sounds produced amphibians under different environmental conditions (temperature, humidity, sunlight, etc.).

2 Fundamentals

The OSHWA(Open Source Hardware Association) statement of principles say “Open source hardware is hardware whose design is made publicly available so that anyone can study, modify, distribute, make, and sell the design or hardware based on that design. The hardware's source, the design from which it is made, is available in the preferred format for making modifications to it. Ideally, open source hardware uses readily-available components and materials, standard processes, open infrastructure, unrestricted content, and open-source design tools to maximize the ability of individuals to make and use hardware. Open source hardware gives people the freedom to control their technology while sharing knowledge and encouraging commerce through the open exchange of designs”[4].

Arduino is an example of open source hardware. According to creators “Arduino is an open-source prototyping platform based on easy-to-use hardware and software” [5]. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. From an innovation perspective Arduino is a medium to transforms ideas in innovative products. It is possible to transform Arduino in a robot, alarm system, irrigation system and others [6].

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. From Android 3.1 (Honeycomb) is introduced the Android Open Accessory that allows external USB hardware (Android USB accessories) to interact with Android-powered devices in accessory mode. Some previously-released Android-powered devices are capable of acting only as a USB device and cannot initiate connections with external USB devices. Android Open Accessory support overcomes this limitation, enabling to build accessories that can initiate connections and interact with an assortment of Android-powered devices [7]. Arduino boards uses this mode for interacting with android devices. This attribute allows to develop a product composed of Arduino and Android device. The decision of designer determines that functionality can be assigned either to Arduino or to Android device.

3 Development methodology

The datalogger was developed following a methodology that was designed for this purpose called Andari. It is a methodology to agile and flexible development that aims to guide the development of software to integrate both platforms, Arduino and Android. Andari proposes a process of constant, iterative and incremental work and is based on

the agile method Scrum. The result of each phase is a functional prototype product. The figure 1 presents the various stages including the methodology.

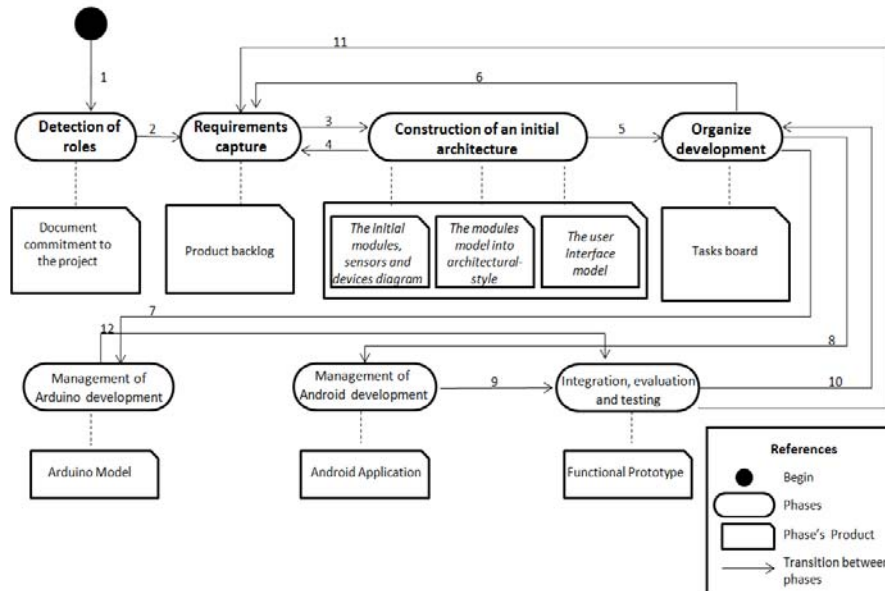


Figure 1. Andari methodology

3.1 Methodology phases.

1. **Detection of roles:** tasks and responsibilities of the participants are defined. Members of team must sign a document committing themselves to complete the project. Product: Document commitment to the project.
2. **Requirements capture:** it is defined as obtaining project requirements through different techniques and strategies. Product: List of requirements. Usually called product backlog.
3. **Construction of an initial architecture:** it is the description system through views using diagrams and models. It is a software architecture that serves only to functional requirements. The architecture is described by three components: the initial modules, sensors and devices diagram, the modules model into architectural-style and user interface model.
 - The initial modules, sensors and devices diagram* allow to describe the first overview of the major components of software and hardware and how they interact.
 - The modules model into architectural-style* allow to determine the more suitable architectural style as a base architecture for the product backlog and bring the initial modules, sensors and devices diagram to an architectural-style.

—*The user interface model* is a model that contains all user interfaces according to the initial view that the developer has about system. It is composed of hardware interfaces (Arduino interfaces) and Android interfaces.

Product: initial architecture.

4. **Organize development:** this process consists of organizing the development. Time and people are assigned to tasks. This organization is based on Sprint. The goal of each sprint is to develop certain requirements and consists of a list of tasks that compose the tasks board. The development in Arduino and Android is managed using the task board. Product: tasks board.
5. **Management of Arduino development:** in this phase Arduino development is managed. Product: Arduino Model
6. **Management of Android development:** in this phase in Android development is managed. Product: Android Application.
7. **Integration, evaluation and testing:** It is the process that integrates development activities in order to obtain a prototype of product. For each sprint the prototype evolves to the final product. Product: Functional Prototype.

3.2 Transitions between different phases of methodology

1. The process starts, and the first task is determining roles.
2. Upon completion determining roles of participants, we continue with the capture of requirements.
3. When finished with the capture of requirements should build an initial architecture which will do the base for the construction of the project.
4. New requirements can appear in the phase construction of the modules of the initial architecture that must be captured and analyzed.
5. With the initial architecture, the work-team passes to organize how modules will be developed, how long it will take us to do it and who are involved.
6. In Organize development, like phase of Initial architecture, new requirements may also appear, this capture and analysis is important.
7. Development of functionality in Arduino platform.
8. Development features in Android platform.
9. The work-team proceed to the integration of features developed in Arduino with the features of Android, the product prototype is tested and verified.

4 Development of prototype.

The datalogger prototype is developed following the Andari methodology. Below, tasks of each phase are presented.

4.1 Detection roles

At this stage the team is formed, it will be formed by customer, developer, technical experts and scrum master. The client is expert biologists. The developer is a senior student career Computer Science, who will make their final thesis. Technical experts

are teachers with technical skills required for product development technically assist the student. The scrum master is professor director of student final work.

4.2 Requirements Capture

Once completed the steps of defining roles the team started with requirements analysis where a detailed final product description desired with all functional specifications required by the customer was obtained. These requirements are summarized in the product backlog, shown in table 1.

Table 1. Product Backlog

ID	Functional requirements
R1	The system should record audio
R2	The system must record temperature
R3	The system should record humidity
R4	The system should display messages start and stop audio recording
R5	The system should display messages start and stop recording temperature
R6	The system should display messages start and stop recording humidity

4.3 Construction of the initial architecture

The main modules to form the system are determined in this phase. The most appropriate architectural-style is selected and the model of the user interface is defined.

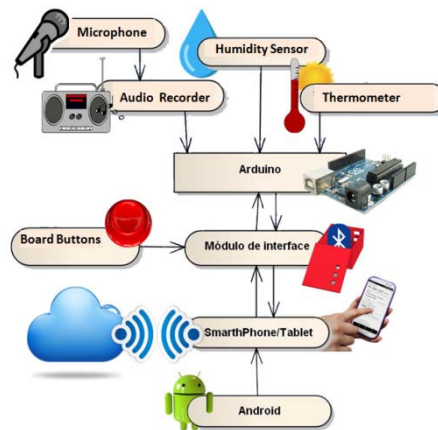


Figure 2. Initial modules, sensors and devices diagram of datalogger.

The initial modules, sensors and devices diagram.

The product backlog is the starting point to build the initial modules, sensors and devices diagram. Requirements indicate that the datalogger will have to ensure

functionalities to record environmental humidity and temperature variables, in addition to environmental audio recording. Another requirement is the communication using mobile devices. The figure 2 shows the diagram.

Description of the modules, sensors and devices

- Microphone: This device can transform sound waves into electrical energy and send all this information to the recorder module to be processed
- Module recorder: records sounds as ogg media files on a removable microSD card, attached to this module.
- Humidity sensor: performs the quantification of humidity in decimal numbers and sends this information to the Arduino.
- Temperature sensor: performs the quantification of temperature on decimal numbers and sends this information to the Arduino.
- Arduino is a device consisting of a plate with inputs and outputs that has a development environment that implements a language Processing / Wiring programming. Arduino receives data that were sent from the modules and sensors, determines the appropriate instruction and executes some action using the Arduino IDE tools.
- Pushbutton: a device that adds extra features to the standard Arduino interface.
- Interface Module: represents the functional connection between the Arduino and the smartphone and allows remote control of the smarphone with Android OS all the features of the prototype, uses SSP (Serial Port Profile).
- Android is comprised of a mobile operating system based on Linux, aimed for use on mobile devices with touch screen as smartphones, tablets, Google TV, and other device.
- Cloud is a device consisting of a platform for digital distribution of mobile applications for devices with Android Operating System. This platform allows users to browse and download applications. Through Cloud applications can be automatically updated if the user so requires, or can do it manually one by one.

The modules model into architectural-style

Once built the initial model of module system, it was determined that the most appropriate architectural style is the three-layer architecture. The figure 3 shows the datalogger architecture.

- *Physical Layer*. It is where the data resides and responsible for accessing them. Physical layer consists of a microphone, an audio converter, two sensors of humidity and temperature. These modules perform all storage data, receive the request for storage and retrieval of information from the processing layer.
- *Processing Layer*. It is where programs that run from the Arduino device reside. It is called processing layer because this is where all the rules are set to be met. This layer communicates with the user interface layer, to receive requests and deliver results. The processing layer communicates with the physical layer, to request the data manager to store or retrieve data from it.

- *User Interface Layer.* User Interface deals with present the system to the user, communicates information and capture user information in a minimum of process (performs pre-filtering to check for any formatting errors). The user interface resides on the android device. This layer communicates only with the layer processing.

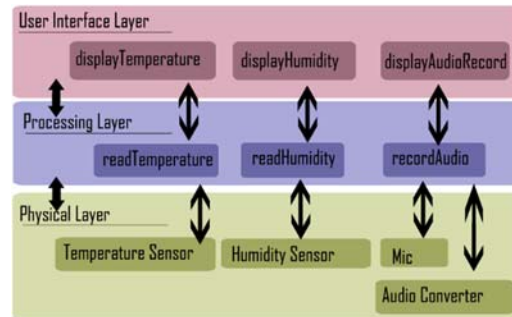


Figure 3. Datalogger architecture

User Interface Model.

From architecture it can be understood broadly general features. According to the proposal, the user interface will only be in the Android device, this was taken into account in designing an outline of the Android application.

4.4 Organize development

The project duration was estimated at 120 days with working day 5 hours daily. The first step in the project organization was grouped requirements of each sprint (see table 2). Subsequently the tasks board was generated (see table 3).

Table 2. Sprints

ID	From	To	Days	Name	Goal	Req.
S1	1-oct	1-nov	30	Record Sound	Reading and converting analog audio into digital audio in order to be stored into flash memory	R1
S2	1-nov	1-dic	30	Record temperature and humidity	Reading, displaying and storing of temperature and humidity	R2 y R3
S3	1-dic	1-en	30	Android interface design	Android interface design and message display	R4, R5 y R6
I4	1-en	1-feb	30	Communication	Integrate Android and Arduino using Bluetooth communication	R4, R5 y R6

4.5 Development Management in Arduino and Android

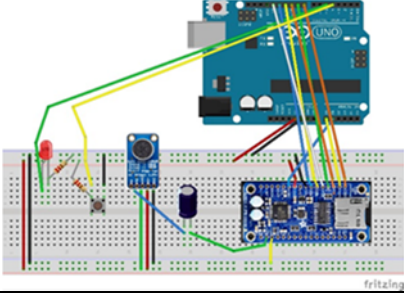
It is aimed to achieve in the each sprint an evolvable functional prototype as seen in tasks board (see table 3). The tasks for the development of Arduino are: record sound, record temperature, record humidity and communication. These tasks are present in 3 of the 4 sprint. The task of managing Android development was carried out in the sprint 3, and basically a layout object is used as a container for one or more views. The main development effort was in the programming and integration of Arduino with external devices. To this purpose, various experiments were conducted with identical functionality devices. For example, with the microphone were carried out three experiences with different types of microphones. This allowed to evaluate different devices with the same functionality to determine which device is best suited to the functionalities required by customers. Each experiment is described by the description, the diagram of connectivity, the program developed, results and conclusions. By way of summary, the development of experiment 3 included in the sprint 1 is presented in table 4.

Table 3. Tasks board

Sprint	Task	Management of Arduino development				Management of Android development				Integration, evaluation and testing			
		Subtasks	To do	In progress	Done	Subtasks	To do	In progress	Done	Subtasks	To do	In progress	Done
1	Record audio	Read audio		X									
		Analogic to digital	X										
		Save audio											
2	Record temperature	Read Temperature	X										
		Display Temperature	X										
		Save Temperature											
2	Record humidity	Read Humidity	X										
		Display Humidity	X										
		Save Humidity											
3	Android Interface Desing					Screen	X						
						Buttons	X						
4	Communication	Bluetooth connection	X			Bluetooth connection	X			Bluetooth connection	X		

Table 4. Sprint1 – Sound Record

Objetive	The device must capture sound waves. These waves must be transformed into data, that will then be stored in some type of mass storage.
Experiment 3	
Description	VS1053 Adafruit device was used. This contains the decoder chip VS1053 and audio encoder [8]. In addition to converting analog to digital signals, also it allows you to save data in a micro-sd.

Diagram of connectivity	
Program Developed	Sound record routine.
Results	The sounds were recorded and reproduced with acceptable fidelity
Conclusions	After performing various tests, and test several alternatives could be achieve the objective of sprint with the MAX9814 microphone [9], and the V1053 chip which satisfied the expectations of expert biologists.

4.6 Integration, evaluation and testing

Integration is the process that allowed in each sprint obtain a functional prototype. This was tested in order to assess the features that were added in each sprint. The final prototype is showed the figure 4.



Figure 4. Final prototype

5 Conclusions

The main objective of this work was to develop a datalogger to support the research activities carried out by biologist researchers of San Juan National University. The developed device has been specified by biologist researchers. It aims to meet current needs, capture sound, temperature and humidity. In addition, the device has been designed taking into account the reuse and / or adaptation in order to incorporate or adapt functionalities needed in future research. Another problem that motivated this work was the cost. In this regard, this work makes an important contribution because it allows the cost of acquiring a device this composed of only the costs of building the hardware. The cost of software is null or nonexistent, because the software is the result of this work and is available to be used in future devices.

References

- [1] S. Oviedo, D. Diaz, A. Otazu, and F. Ibañez, "Software y Hardware de Código Abierto para Desarrollo de Habilidades de Innovación Tecnológica," *LACLO 2014- IX Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje. Colombia. Manizales.*, 2014.
- [2] J. M. Osier-Mixon. (2010) Open hardware: How and why it works. *IBM developerWorks*
- [3] M. Hinkle, "Open Source: A Platform for Innovation," <http://insights.wired.com/profiles/blogs/open-source-a-platform-for-innovation#axzz2oOFSuMAI>, 2013.
- [4] oshwa, "Open Source Hardware Association," <http://www.oshwa.org/>, 2016.
- [5] Arduino, "Web Page," <http://www.arduino.cc/es/>, 2016.
- [6] D. Díaz, S. Oviedo, and L. Muñoz, "Software e innovación: desarrollando productos con hardware y software flexible. ," *CACIC 2013 V Workshop Innovación en Sistemas de Software (WISS)*, 2013.
- [7] AOA, "Android Open Accessory (AOA)," <https://source.android.com/devices/accessories/protocol.html>, 2016.
- [8] V. ADAFRUIT, "Recording with the VS1053," <https://learn.adafruit.com/adafruit-vs1053-mp3-aac-ogg-midi-wav-play-and-record-codec-tutorial/ogg-recorder>, 2016.
- [9] A. MAX9814, "MAX9814," <https://learn.adafruit.com/adafruit-agc-electret-microphone-amplifier-max9814/overview?view=all>, 2016.

Librería Multiplataforma para Videojuegos en Dispositivos Móviles

Cristian Costa, Federico Bricker y Claudio Aciti¹

¹ Universidad Nacional del Centro de la Provincia de Buenos Aires
Pinto 399 (7000) – Tandil – Argentina

caciti@exa.unicen.edu.ar

Abstract. This paper describes how to homogeneously use several game-engagement components included on different mobile platforms, that helps increasing gameplay on video games. Each platform provides a service API for handling leaderboards, achievements and social interaction between players. Design choices made by each platform's provider, impacts on the game's development tasks and process. Using the multi-platform language Haxe, individual libraries were built to allow easy integration of Amazon, Apple and Google's gaming services. With them an extra library that homogenizes the common services of each provider was generated, which -based on the target platform- determines the implementation to include in the final application and therefore which services to use.

Resumen. Este trabajo describe la utilización homogénea de componentes lúdicos facilitados por proveedores de distintas plataformas móviles. Tales componentes permiten incrementar la mecánica de juego de aplicaciones lúdicas en el contexto de los dispositivos móviles. Cada proveedor es el encargado de otorgar al desarrollador, una API de servicios para la manipulación de elementos como logros y tablas de clasificación. Las decisiones políticas y tecnológicas tomadas por cada uno impacta en la tarea del desarrollador de aplicaciones móviles. A partir del lenguaje multiplataforma Haxe, se construyeron librerías individuales que permitan usar los servicios de las implementaciones de Amazon, Apple y Google. Con ellas se generó una librería extra que homogeniza los servicios comunes de cada proveedor y, a partir de la plataforma objetivo, determina cual implementación del proveedor incluir en la aplicación final y por ende, cuáles servicios utilizar.

Keywords: Componentes lúdicos, API homogénea, multiplataforma, Haxe, OpenFL, videojuegos, dispositivos móviles.

1 Introducción

En el último año se observó un crecimiento en el mercado de los teléfonos inteligentes con 1200 millones de unidades vendidas. Actualmente, estos dispositivos

cuentan con grandes capacidades de procesamiento, almacenamiento y conectividad, siendo su plataforma móvil, la encargada de administrar tales recursos. Estas plataformas son proporcionadas por proveedores tales como: Apple, Google, Windows, etc. Aquí, cada uno determina el lenguaje de programación en el que está concebida su plataforma, restringiendo el desarrollo y funcionamiento de aplicaciones. En cada una de estas plataformas, se encuentra un gestor de aplicaciones que permite a los usuarios instalar aplicaciones de acuerdo a sus necesidades [1].

No hay restricciones sobre quien puede crear aplicaciones, pudiendo ser: el proveedor de la plataforma, el proveedor del dispositivo, una empresa de desarrollo móvil o un desarrollador particular. Independientemente de quien sea el autor, las aplicaciones pueden seguir estrategias de monetización [2] para generar ganancias. Para insertar aplicaciones al gestor, el proveedor de la plataforma móvil ofrece un portal web llamado consola de desarrollador.

A partir de estas condiciones, es válido pensar que cualquier desarrollador desee cubrir la mayor cantidad de plataformas con la misma aplicación. De esta forma abarcar una mayor cantidad de público y, a partir de las estrategias de monetización, mayor ganancia. Pero, como cada proveedor utiliza su lenguaje de programación predilecto para concebir su plataforma móvil, el desarrollador está obligado a generar la misma aplicación en cada lenguaje [3].

Ningún tipo de aplicación está exento a la problemática de múltiples plataformas, ni siquiera los videojuegos. Adicionalmente, las aplicaciones de este estilo conviven con el desafío de cautivar o “engañar” a los usuarios que las utilicen. En los videojuegos, el número de participantes a atraer está determinado por las mecánicas del juego [4]. Estos son los componentes base del juego: reglas o acciones que puede realizar un jugador en el mismo. Para crear mecánicas de juego existe un conjunto de herramientas o componentes lúdicos tales como: logros, niveles, tablas de clasificación, sistemas de puntajes, etc..

En el ámbito de los dispositivos móviles, los proveedores permiten la generación de logros y tablas de clasificación desde la consola de desarrollador. Estos mismos buscan, mediante la difusión del progreso de cada jugador a los restantes, conformar un estado de competitividad entre los participantes. Para manipular estos componentes (por ejemplo: guardar un puntaje en una tabla dada) cada proveedor ofrece una API de servicios para utilizar desde el código fuente de la aplicación. Ahora bien, en caso de que el desarrollador desee simultáneamente:

- Cubrir con el mismo videojuego varias plataformas.
- Utilizar los componentes lúdicos de los proveedores de cada plataforma.

Se encontrará con desventajas impuestas por las API's que manipulen tales componentes:

- El proceso de integración de cada API es distinto para con el mismo proyecto.
- Existe una falta de estandarización entre API's. Por ejemplo, ciertos servicios que otorgue la implementación de un proveedor, pueden no estar en la de otro.
- Cada API se encuentra implementada en el mismo lenguaje de programación que la plataforma móvil del proveedor. Por ende, la lógica del videojuego también debe ser desarrollada en tal lenguaje.

1.1 Motivación

La situación que motiva a afrontar un problema de esta índole se hace presente en el contexto de una empresa de desarrollo de videojuegos para dispositivos móviles. Aquí, sus objetivos constan en:

1. Comercializar sus productos en las distintas tiendas.
2. Abarcar la mayor cantidad de usuarios a partir del nivel de “enganche” que generen sus aplicaciones.

Para lograr el primer objetivo la empresa utiliza un lenguaje multiplataforma, mientras que para el segundo, la misma hace uso de componentes lúdicos propios y facilitados por los proveedores. Para manipular a estos últimos, es necesario contar con la API de cada proveedor e integrarla al código fuente. Esta situación genera conflictos con el lenguaje multiplataforma:

- Incompatibilidad entre los lenguajes de programación utilizados. Esto dificulta la utilización directa de los servicios desde el código fuente de la aplicación.
- Cada API está restringida a la plataforma móvil donde funciona correctamente, anulando la capacidad multiplataforma del lenguaje para crear la aplicación final (condicionando la salida para la cual debe generarse la aplicación final).

Todos estos condicionamientos se traducen en incrementos de tiempo y esfuerzo invertido que serían deseables evitar.

1.2 Objetivos

Se propone la creación de una API que permita utilizar, de forma homogénea, los servicios de las implementaciones de cada proveedor. Así mismo se establecerán como objetivos parciales la generación de módulos específicos para cada proveedor.

1.3 Alcances y Limitaciones

La solución se realiza en el lenguaje multiplataforma Haxe utilizando las implementaciones de los proveedores Amazon, Apple y Google. Las limitaciones existentes son causadas por la falta de estandarización a partir de las políticas adoptadas por cada proveedor para materializar su API de servicios.

1.4 Organización del Trabajo

En este primer capítulo se introduce la problemática abordada. En el segundo, se realiza una reseña del estado del arte. En el tercer capítulo se establecen los requerimientos y el punto de partida del proyecto. En el capítulo cuatro se describe la solución y se narran las estrategias utilizadas para su producción. Posteriormente, en el quinto capítulo se muestran análisis que permiten conocer las ventajas que ofrece el uso de la solución. En el sexto se exponen las conclusiones obtenidas de este trabajo. En el séptimo capítulo son enumerados una serie de trabajos futuros propuestos como

continuación de este estudio. Finalmente se listan las referencias bibliográficas utilizadas.

2 Estado del Arte

2.1 Componentes Lúdicos de los Proveedores

Actualmente Apple, Amazon y Google ofrecen sus servicios de juegos [5] que permiten a los desarrolladores el agregado de componentes lúdicos como son los logros y las tablas de clasificación. Así mismo, estos proveedores ofrecen sus API's de servicios para manipular los atributos de cada elemento. Por otro lado, la ejecución de los servicios de cualquiera de las API's se realiza de manera asíncrona. En términos de diferencias conceptuales, estas se evidencian en los Logros: Google utiliza la noción de pasos, mientras que Amazon y Apple utilizan la noción de porcentaje de avance. Por último, en el aspecto tecnológico, Amazon y Google ofrecen el servicio de sincronización de partidas entre distintos dispositivos dentro de su implementación, mientras que Apple lo ofrece con otra API dedicada exclusivamente a dicha tarea.

2.2 Haxe/OpenFL

Haxe [6] es un lenguaje de programación de código abierto multiplataforma que permite generar código para lenguajes como JavaScript, Flash, PHP, C++, C# y Java. La ideología que sigue es, permitir al desarrollador seleccionar la plataforma objetivo más acorde al problema a resolver aumentando la flexibilidad en el desarrollo a la vez que se maximiza el retorno de una inversión. Una de las características más importantes de Haxe es que permite, a partir de macros de compilación condicional, ejecutar código específico para una determinada plataforma.

OpenFL [7] es un framework de código abierto multiplataforma que, haciendo uso de Haxe permite la generación de aplicaciones o videojuegos ejecutables, pudiendo correr sobre plataformas como: Windows, GNU/Linux, navegadores web con soporte de HTML5 o Flash, teléfonos inteligentes y tablets con Android o iOS. OpenFL permite materializar librerías a partir de implementaciones no existentes en Haxe ni en dicho framework, vinculando y haciendo uso de servicios provistos por terceros.

3 Descripción del Problema

A partir de las implementaciones de los proveedores en conjunción con Haxe/OpenFL se pretende generar una API que permita hacer uso de los servicios de cada uno de las implementaciones de los proveedores que manipulan los componentes lúdicos de forma homogénea. Para ello, la solución debe satisfacer los siguientes requerimientos:

- Pertenecer al lenguaje Haxe.

- Facilitar la actividad de integración al desarrollador. Transitivamente, este aspecto reducirá el tiempo invertido en realizar tal actividad.
- A partir de la plataforma objetivo para la cual sea generada la aplicación final, la solución debe determinar cual implementación del proveedor incluir en la misma.
- Por último, la solución, al tratarse de una API, debe ofrecer a los desarrolladores de videojuegos en Haxe los servicios presentados en la Tabla 1.

Tabla 1. Requerimientos solicitados.

Componente	Descripción del servicio
Tabla de clasificación	Visualizar una tabla en particular.
	Visualizar todas las tablas del juego.
	Almacenar un puntaje en una tabla dada.
	Obtener el puntaje de una tabla dada.
Logro	Visualizar todos los logros del juego.
	Desbloquear un logro en particular.
	Establecer el progreso de un logro dado.
	Revelar un logro en particular.
	Incrementar el progreso de un logro dado.
	Obtener el progreso de un logro dado.
	Obtener el estado de un logro dado.
Almacenamiento	Almacenar partida.
	Recuperar partida.

Por otro lado, se debe realizar una “línea base” del trabajo, presentada en la Tabla 2. Esto se debe a que inicialmente, la comunidad de Haxe/OpenFL ya contaba con dos complementos individuales de los proveedores de Apple y Google. A partir de los existentes es necesario completarlos y generar los complementos faltantes (Amazon y API Homogénea).

Tabla 2. Línea base del proyecto.

Servicio	Apple	Google	Amazon	API Homogénea
Visualizar tabla.	Si	Si	Posible	Posible
Visualizar tablas.	No	Si	Posible	Posible
Almacenar puntaje.	Si	Si	Posible	Posible
Obtener puntaje.	Posible	Posible	Posible	Posible
Visualizar logros.	Si	Si	Posible	Posible
Desbloquear logro.	No	Si	No	Posible
Establecer progreso.	Si	Si	Posible	Posible
Revelar logro.	No	Si	No	Posible
Incrementar progreso.	No	Si	No	Posible
Obtener progreso.	Posible	Posible	Posible	Posible
Obtener estado.	Posible	Posible	Posible	Posible
Almacenar partida.	No	Si	Posible	Posible

Recuperar partida.	No	Si	Posible	Posible
--------------------	----	----	---------	---------

Las referencias para cada valor que puede contener una celda se aclara a continuación:

- Las celdas que indican “Si”, denotan que la funcionalidad se encuentra implementada.
- Las celdas con el estado “Posible”, indican que la funcionalidad no se encuentra implementada pero, a partir de la documentación de la API del proveedor, es posible implementarla.
- Las celdas con “No” establecen que no es posible generar la funcionalidad ya que la implementación del proveedor no la ofrece.

4 Solución Propuesta

Se propone brindar al desarrollador, una API que permite utilizar servicios para videojuegos de diversos proveedores, de forma homogénea y en lenguaje Haxe. En la Figura 1 se presenta en alto nivel el rol que cumple la solución.

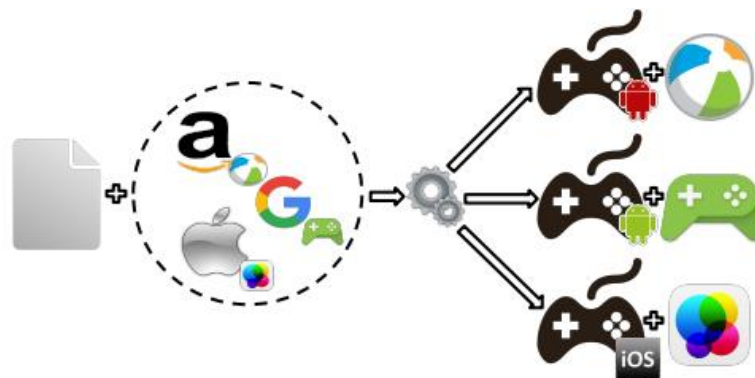


Figura 1. Esquema de la solución en alto nivel.

Dado que cada proveedor utiliza aspectos tecnológicos diferentes para concebir su API, la solución está compuesta por módulos específicos que permitan la utilización de cada implementación desde el lenguaje Haxe (estos módulos se llamarán Módulos Puente).

Esta API, entiende cómo utilizar los componentes lúdicos disponibles en cada proveedor; y dependiendo de la plataforma para la cual sea compilado el videojuego, utiliza la implementación correspondiente sin que esto requiera adaptación alguna por parte del desarrollador. Así mismo facilita el proceso de integración para cualquier proyecto reduciendo tiempo y esfuerzo insumido para tal actividad.

Para los casos en los que el juego sea generado hacia una plataforma para la cual no hay un módulo puente integrado con la solución, la API resulta inocua, permitiendo la compilación y ejecución del juego sin errores ni condicionamientos.

En la Figura 2 se muestra a más bajo nivel la predisposición de módulos puentes que conforman a la API Homogénea solución. Cada uno de los módulos fue concebido a partir de la herramienta de creación de extensiones de Haxe/OpenFL. Los módulos puentes utilizan las implementaciones de los proveedores en conjunción con la capacidad de vincular e invocar servicios desde el lenguaje Haxe hacia el lenguaje de programación utilizado en la implementación del proveedor. Por último, la API Homogénea, hace uso de la compilación condicional de Haxe para agrupar en un único servicio los servicios comunes de cada módulo puente. A lo largo del desarrollo de cada módulo puente, se buscó ofrecer la funcionalidad lo más completa posible que brinda cada proveedor como así también mantener un enfoque que respete los estándares de desarrollo en Haxe/OpenFL.

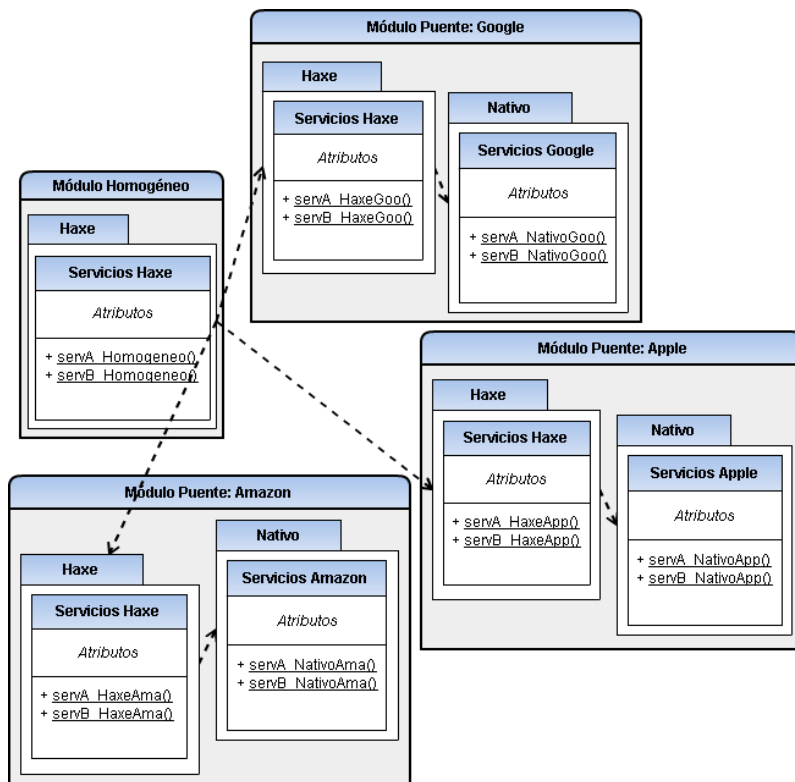


Figura 2. Predisposición de módulos puentes que conforman a la solución.

Desde el punto de vista del desarrollador, la solución es relativamente simple de utilizar, aunque se destaca la utilización de un sistema de retrollamadas (callbacks) que aumenta la complejidad de uso para todas las funciones de las cuales se espera una respuesta o valor de retorno. Esto se debe a que los servicios de los proveedores trabajan de manera asíncrona, con lo cual se separa la forma de utilizar las funciones que no entregan información al desarrollador (almacenar un puntaje, mostrar todas las

tablas, mostrar logros, etc.); de aquellas que sí lo hacen (obtener el puntaje de una tabla, obtener el progreso de un logro, etc.). Para el segundo tipo de funciones será necesario procesar la respuesta una vez que esté disponible. Las retrollamadas establecen que: en vez de esperar el resultado de una función (blocking), o de consultar esporádicamente por el estado de retorno (polling), el desarrollador debe informar qué función debe ser invocada cuando se obtenga el resultado.

5 Pruebas de Campo

Para hacer uso de la solución es necesario incluir la misma a cada proyecto Haxe. Esto se realiza utilizando dos directivas, una en el archivo de definición de proyecto y otra en el código fuente de la aplicación. Una vez hecho esto, se está en condiciones de poder realizar invocaciones de los servicios de la solución.

Para el análisis, se utilizaron los proyectos de la empresa como modelos de pruebas. El estudio consistió en comparar cada aplicación consigo misma, con la diferencia de que:

- El primer caso, supone la integración de las implementaciones de los tres proveedores sin la intervención de la solución.
- El segundo, supone el uso de la API homogénea.

De la tarea de integración se contemplaron las actividades que influyen en aspectos como: Tiempo insumido y Cantidad de sentencias necesarias a agregar en cada proyecto. En la Tabla 3 se presentan las condiciones a considerar de ambos casos.

Tabla 3. Condiciones previas al estudio de ambos casos.

Sin API Homogénea	Con API Homogénea
<ul style="list-style-type: none"> • Módulos existentes e incompletos (Apple y Google). • Modulo inexistente (Amazon). • Interfaz homogénea inexistente. 	<ul style="list-style-type: none"> • Módulos de los proveedores completos y validados. • Interfaz homogénea existente.

El primer caso se corresponde a la “línea base” establecida en la sección 3:

- Por un lado, los módulos puente de Apple y Google ya existen, pero ambos se encuentran incompletos.
- Otro aspecto a tener en cuenta es que: el único módulo que debe ser creado de cero es el de Amazon.
- Por último, en el código fuente de cada proyecto se debe encontrar la misma lógica que presenta la API Homogénea, agrupando los servicios comunes de cada módulo puente en un único servicio a partir de la compilación condicional de Haxe.

De estas condiciones se concluye que: los puntos 1 y 2 (creación y completitud de cada módulo) se deben realizar una única vez para el primer proyecto, mientras que el punto 3 debe repetirse en todos los proyectos. Por último, se deben tener en cuenta las

sesiones de pruebas a los que debe someterse cada servicio de juegos y eventuales correcciones que deban realizarse en caso que se presenten errores. Estos dos últimos aspectos insumirán tiempo extra a la tarea de integración.

En cambio, con el segundo caso ya nos encontramos con una API que presenta una interfaz homogénea y módulos puentes completados, verificados y validados. Con lo cual es intuitivo pensar que la cantidad de sentencias necesarias para utilizarla y el tiempo de integración disminuyan con relación al primer caso.

Entonces, se tomó un proyecto de la empresa y, a partir de ambos supuestos, se estimaron los siguientes datos mostrados en las Tablas 4 y 5:

Tabla 4. Estimación del caso Sin API Homogénea.

	#Días inclusión	#Sentencias
Proyecto	34	320

Tabla 5. Estimación del caso Con API Homogénea.

	#Días inclusión	#Sentencias
Proyecto	2	2

El valor de 34 días, es la cantidad de tiempo aproximado que demandó generar el módulo de Amazon, adicionado al tiempo necesario para completar los módulos de Google y Apple con sus correspondientes sesiones de pruebas para cada uno de los tres.

Las 320 sentencias necesarias para agrupar los servicios comunes de cada módulo son la cantidad de sentencias que conforman a la lógica de la API Homogénea. La diferencia radica en que, para el primer caso, la lógica forma parte del código fuente de cada videojuego, mientras que para el segundo caso dicha lógica es ajena.

En conclusión, hacer uso de la API Homogénea para el desarrollo de videojuegos en el lenguaje Haxe significa un ahorro en el tiempo de lanzamiento del mismo a un mes.

En aspectos cualitativos se destaca:

- La mantenibilidad de la API Homogénea a partir de su diseño centralizado es una ventaja frente al caso donde no se utiliza la misma, reduciendo el costo en términos de tiempo y esfuerzo. Realizar los cambios sobre un único lugar impacta sobre todas las aplicaciones que lo utilicen.
- La escalabilidad en el caso de que surjan nuevos servicios de juegos, insumiendo menor esfuerzo crear el módulo puente e integrarlo a la API Homogénea que integrarlo en cada proyecto.

6 Conclusiones

Se implementó una librería que permite utilizar, de forma homogénea, servicios para manipular los componentes lúdicos que ofrecen los proveedores desde el lenguaje

Haxe. La solución es capaz de, a partir de un único código fuente y la plataforma objetivo deseada, determinar cual implementación del proveedor incluir en la aplicación final sin perjudicar al lenguaje multiplataforma. En base a lo observado durante las pruebas se apreció el nivel de impacto y las ventajas que ofrece utilizar este complemento en relación a no hacerlo. Adicionalmente, la solución posee flexibilidad ante eventuales expansiones, permitiendo agregar nuevos servicios para los proveedores considerados o permitiendo agregar implementaciones de proveedores no contemplados en el presente trabajo.

7 Trabajos Futuros

A partir de lo realizado, se proponen como trabajos futuros las siguientes consideraciones:

- Expandir la API Homogénea tanto en número de servicios como en número de proveedores.
- Otorgar la posibilidad a los jugadores de plasmar su progreso desde el mismo videojuego en los servicios de juegos de cada proveedor.
- Agregar el soporte necesario para la sincronización de partidas en dispositivos que contengan la plataforma móvil de Apple.
- Permitir que la API Homogénea mantenga el estado de ciertos atributos de interés para los componentes lúdicos utilizados. Almacenar ciertos valores en estructuras pertenecientes a la solución evitarían utilizar los servicios asíncronos para consultarlos.

8 Referencias

1. Jansen, S., Bloemendal E.: *Defining App Stores: The Role of Curated Marketplaces in Software Ecosystems*. Springer Berlin Heidelberg (2013).
2. Cravens, A.: *A demographic and business model analysis of today's app developer*. Application Developers Alliance (2012).
3. Joorabchi, M.E., Mesbah, A., Kruchten, P.: *Real Challenges in Mobile App Development*. IEEE International Symposium on Empirical Software Engineering and Measurement (2013).
4. Zichermann, G., Cunningham C.: *Gamification by Design Implementing Game Mechanics in Web and Mobile Apps*. O'Reilly Media, Inc. (2011).
5. de Prato, G., Feijoo, C., Simon, J-P.: *Innovations in the Video Game Industry: Changing Global Markets*. Digiworld Economic Journal (2014).
6. Ponticelli, F., McColl-Sylvester, L.: *Professional haXe and Neko*. Wiley Publishing, Inc. (2008).
7. Vallejo Fernández, D., González Morcillo, C., Frutos Talavera, D.: *Videojuegos Multiplataforma con OpenFL*. Edlibrix. (2014).

Experimentando un uso no tradicional de PhoneGap

Agustina M. Zimbello¹ and Cecilia Challiol^{1,2}

¹ LIFIA, Facultad de Informática, UNLP, 50 y 120 Primer Piso,
La Plata, Argentina
{azimbello,ceciliac}@lifia.info.unlp.edu.ar
<http://www.lifia.info.unlp.edu.ar>
² CONICET, Argentina

Resumen. En este trabajo presentamos un plugin de *PhoneGap* para Android que permite una comunicación genérica entre los componentes visuales de una aplicación *PhoneGap* y una clase que funciona como punto de entrada de algún modelo Java. Dicho plugin fue utilizado para la creación de una aplicación móvil basada en posicionamiento. En este trabajo se brindan detalles de nuestro plugin, como así también aspectos aprendidos durante el desarrollo y futuros usos del mismo.

Palabras Claves: PhoneGap, Aplicaciones Móviles basadas en posicionamiento, Prototipado, Computación Móvil.

1 Introducción

El avance de la tecnología ha traído en consecuencia la creciente y variada gama de dispositivos y plataformas móviles disponibles (Android, iOS, etc.). Esto genera un desafío a la hora de desarrollar aplicaciones móviles, sobre todo poder cubrir esta variada gama en corto tiempo. Varios autores (por ejemplo, [3] y [13]) coinciden en que se pueden identificar tres posibles enfoques de desarrollo: aplicaciones móviles nativas, aplicaciones web móviles o aplicaciones móviles híbridas. La elección de cada uno de ellos conlleva ventajas y desventajas [3]. Es decir, las características propias de la aplicación móvil a desarrollar determinarán cuál es el enfoque que se ajuste mejor a la solución buscada.

Las aplicaciones móviles basadas en posicionamiento, y más aún aquellas sensibles al contexto [8], poseen aspectos particulares que están asociados, en muchos casos, con los datos que proveen los sensores de los dispositivos móviles. Varios autores coinciden ([3],[11],[13]), que para lograr una solución multiplataforma y además poder acceder a los sensores, la mejor opción es optar por desarrollar aplicaciones móviles híbridas. Actualmente, existen diferentes plataformas para crear este tipo de aplicaciones (móviles híbridas). Varios autores (por ejemplo, [3],[4],[7],[9],[12] y [13]) presentan y describen comparaciones de algunas de ellas. En todos estos trabajos, los

autores coinciden en que *PhoneGap*¹ (distribución libre de *Apache Cordova*²) es la plataforma más popular, con más documentación y cuenta con la posibilidad de ser extendida mediante la definición de plugins. *PhoneGap* es un framework que se basa en tecnología web estándar como: HTML, JavaScript y CSS. Cabe destacar que usar esta tecnología permite realizar prototipos rápidos.

En nuestro grupo de investigación hace varios años que venimos trabajando en la temática de aplicaciones móviles sensibles al contexto ([5],[6]). En el 2014 nos invitaron a participar del evento *TEDx Diagonal73*³, que se desarrolló en La Plata. La participación consistía en poner en práctica durante el evento una aplicación móvil basada en posicionamiento, la cual denominamos *Caminos Alternativos*. Para darle más atractivo a la aplicación la combinamos con actuaciones teatrales, las cuales estaban relacionadas con las consignas que se le presentaban al usuario desde la aplicación. En la puesta en práctica de la aplicación había actores que interpretaban escenas en diferentes lugares del edificio. Más detalles de la puesta en práctica de esta aplicación se mencionan en [1]. Es importante mencionar que esta aplicación fue desarrollada usando *PhoneGap* de una manera no tradicional. La motivación de este trabajo es precisamente esta, poder compartir el aprendizaje que experimentamos durante este desarrollo.

El objetivo del trabajo es describir un uso no tradicional de *PhoneGap*. En este trabajo presentaremos un plugin de *PhoneGap*, para Android, que permite una comunicación genérica entre los componentes visuales de una aplicación *PhoneGap* y una clase que funciona como punto de entrada de algún modelo de dominio implementado en Java. Este plugin fue desarrollado motivados por la necesidad de contar con aplicaciones puramente clientes que hicieran usos de aspectos internos del dispositivo móvil (por ejemplo, la cámara). Y, además, poder reusar modelos de dominios existentes (implementados en Java) con funcionalidad específica para aplicaciones móviles basadas en posicionamiento. La necesidad de contar con un plugin viene dada por la forma en que funciona *PhoneGap*, ya que requiere de plugins para comunicar la vista con clases Java. En nuestro caso particular, queríamos conectar las vistas con modelos de dominios que ya teníamos implementados. Y dado que estos modelos tenían diferentes puntos de entradas (clases Java específicas) queríamos encontrar una forma genérica de solución, y ahí es donde surge el plugin genérico que presentamos en este trabajo. La decisión de elegir *PhoneGap* fue aprovechar la ventaja de poder desarrollar usando las tecnologías HTML, JavaScript y CSS, sin tener que aprender aspectos particulares de la definición de las vistas, por ejemplo, en Android. En este trabajo se brindan detalles de nuestro plugin como así también aspectos aprendidos que pueden servir para otros desarrolladores.

La estructura del trabajo es la siguiente. En la Sección 2 se presentan las características de la problemática que se busca resolver. En la Sección 3 se presenta el plugin de *PhoneGap* para Android propuesto. Los resultados obtenidos y las lecciones aprendi-

¹ Página de *PhoneGap*: <http://phonegap.com> (Último Acceso: 13-5-2016)

² Página de *Apache Cordova*: <https://cordova.apache.org> (Último Acceso: 13-5-2016)

³ Página de *TEDx Diagonal7*: <https://www.ted.com/tedx/events/10077> (Último Acceso: 13-5-2016)

das se presentan en la Sección 4. Las conclusiones y trabajos futuros se presentan en la Sección 5.

2 Características de la Problemática a Resolver

En esta sección se describirán algunas características del contexto en el cuál trabajamos y cómo llegamos a usar *PhoneGap* de una forma no tradicional. Como mencionamos anteriormente venimos investigando temáticas afines a las aplicaciones móviles basadas en posicionamiento, en particular, distintos aspectos relacionados al diseño y modelado de las mismas. Por lo tanto, en la aplicación *Caminos Alternativos* presentada en [1] queríamos poner en práctica aspectos de modelado en los cuales veníamos trabajando, por ejemplo, la separación en capas de los contenidos de las posiciones.

Como mencionamos anteriormente *PhoneGap* se basa en HTML, JavaScript y CSS, por lo tanto, toda la lógica de control queda contenida en los JavaScript. Este es el uso tradicional de *PhoneGap*, sin embargo, esto nos generaba pasar nuestros modelos de clases implementados en Java a código JavaScript. Estos modelos de clases los venimos desarrollando como parte de un framework Java que estamos definiendo para brindar soporte en la creación de aplicaciones móviles basadas en posicionamiento [2].

Por otro lado, *PhoneGap* provee el concepto de plugin para extender funcionalidad. Estos plugins proveen una interfaz JavaScript a los componentes nativos del dispositivo (por ejemplo, para acceder a los sensores), permitiendo así que la aplicación pueda usar características nativas del dispositivo más allá de lo que está disponible para aplicaciones web puras. Cada plugin es desarrollado para una plataforma particular, por ejemplo, Android. Los plugins para Android se definen con una clase Java (que posee la funcionalidad para acceder, por ejemplo, a un sensor), un archivo JavaScript (que especifica cómo se debe llamar el plugin desde los HTMLs) y, por último, un archivo XML (el cual permite especificar la relación entre el llamado del plugin con la clase puntual a ejecutarse). Actualmente, *PhoneGap* provee plugins generales que pueden ser descargados y utilizados, por ejemplo, para acceder a la cámara del dispositivo o comunicarse con una base de datos.

Tomando este concepto de plugins, como se mencionó en la Sección 1, nos surgió la idea de usarlo para acceder a clases Java que pudieran servir como punto de entrada de nuestros modelos, y así poder reusar nuestros modelos de clases. Los plugins en *PhoneGap* están definidos para acceder a una clase puntual invocando un método puntual con sus parámetros. Esto nos generaba que por cada método que fuera punto de entrada de nuestro modelo debíamos crear un nuevo plugin, algo que claramente no escalaba. Esto nos motiva a crear una solución genérica, y la misma es presentada en este trabajo en la siguiente sección.

3 Plugin Desarrollado para PhoneGap

El plugin general que desarrollamos está implementado para Android. Al definirlo de manera genérica, puede ser usado para comunicar cualquier JavaScript de la aplicación con alguna clase Java (que sea parte de un modelo). Esta característica nos permite que pueda ser usado con cualquier modelo de dominio. Como se mencionó en la sección anterior, cualquier plugin en *PhoneGap* para Android se debe definir con alguna clase Java, un archivo JavaScript y un archivo XML.

Para entender mejor el funcionamiento del mismo se presenta en la Figura 1 la comunicación que se establece entre la vista (HTML/JavaScript), nuestro plugin y un modelo. La característica genérica nos permite tener definida tanto la vista como el modelo acorde a los requerimientos propios de cada aplicación.

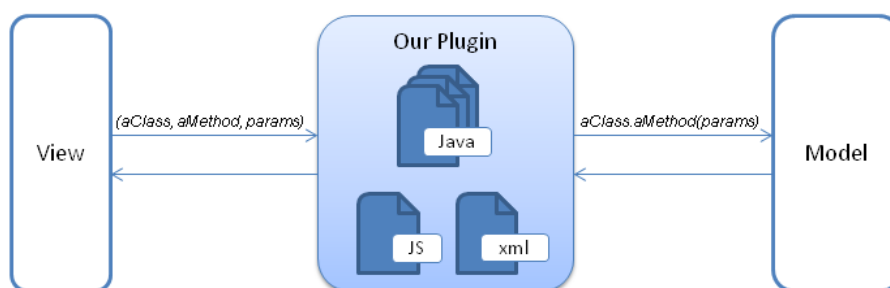


Fig. 1. Comunicación relacionada a nuestro plugin.

Se puede apreciar en la Figura 1 que nuestro plugin es invocado con: *aClass* (representa el nombre de la clase a la cual se desea invocar un método), *aMethod* (representa el nombre del método que se desea ejecutar, el mismo debe existir en el momento de invocarlo) y *params* (representa los parámetros a enviarse en el método indicado, es una colección con la forma <tipo, valor>). Esta información que llega a nuestro plugin es usada con reflection⁴ de Java, para invocar el método de la clase con los parámetros correspondientes. La respuesta brindada por esta clase es enviada a la vista que realizó la invocación. Por ahora, el plugin está implementado para enviar y recibir información en formato JSON (*JavaScript Object Notation*), pero este aspecto está diseñado para ser extendido por cualquier otro formato.

Cabe destacar que se realizó un análisis para determinar si reflection era la mejor forma de solucionar nuestro problema. Si bien reflection puede ocasionar errores en tiempo de ejecución, que no son detectados en compilación, la flexibilidad de invocar cualquier método de cualquier clase en forma dinámica se ajustaba perfectamente a la

⁴ Para más información sobre reflection se puede consultar:

<https://docs.oracle.com/javase/tutorial/reflect> (Último Acceso: 13-5-2016).

problemática que teníamos que dar solución. Por ende, este fue el motivo que prevaleció en la elección del uso de reflection.

A continuación, se presenta el código relacionado a los archivos JavaScript y XML de nuestro plugin como así también la clase Java del mismo.

En la Figura 2 se puede apreciar la definición del archivo JavaScript; se puede observar que el plugin debe ser invocado usando una función que recibe una función de *success* y otra de *error* (estas funciones son invocadas para devolver el control al JavaScript que invocó nuestro plugin), como así también los parámetros ya mencionados en la Figura 1.

```
var pluginHandler = {
  pluginHandler: function(successCallback, errorCallback, theClass, theMethod, arrayParameters) {
    cordova.exec(successCallback,
      errorCallback,
      "PluginHandler",
      "executePlugin",
      [
        {
          "theClass": theClass,
          "theMethod": theMethod,
          "arrayParameters": arrayParameters,
        }
      ]
    );
  }
}
module.exports = pluginHandler;
```

Fig. 2. Archivo JavaScript de nuestro plugin.

En la Figura 3 se puede apreciar la clase Java que representa nuestro plugin, la cual denominamos *PluginHandler*. Esta clase extiende de *CordovaPlugin*, que es la clase que se debe extender para representar un plugin personalizado en *PhoneGap*. Se debe implementar el método `execute()` como se puede visualizar en la figura. Si bien contamos con otras clases que colaboran con la clase *PluginHandler*, a fin de entender el funcionamiento y para simplificar la descripción no entraremos en detalle.

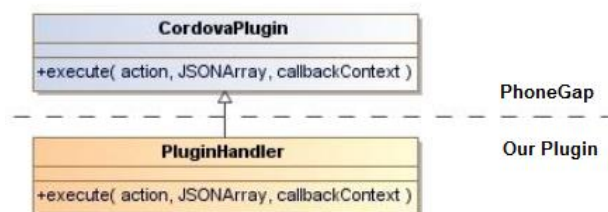


Fig. 3. Clase *PluginHandler*.

En la Figura 4 se puede apreciar el código del archivo XML de nuestro plugin, donde se establece la relación entre el archivo JavaScript y la clase *PluginHandler*.

```

<plugin xmlns="http://www.phonegap.com/ns/plugins/1.0"
  id="com.lifia.plugin.pluginHandler"
  version="0.1.0">
  <name>pluginHandler</name>
  <description> </description>
  <license> </license>
  <keywords>pluginHandler</keywords>
  <js-module src="www/pluginHandler.js" name="pluginHandler">
  <clobbers target="pluginHandler"/>
  </js-module>
  <!--android-->
  <platform name="android">
  <config-file target="res/xml/config.xml" parent="/*">
  <feature name="PluginHandler">
    <param name="android-package"
      value="com.lifia.plugin.pluginHandler.PluginHandler" />
  </feature>
  </config-file>
  <source-file src="src/android/PluginHandler.java"
  target-dir="src/com/lifia/plugin/pluginHandler" />
  </platform>
</plugin>

```

Fig. 4. Archivo XML de nuestro plugin.

De esta manera, queda definido nuestro plugin de *PhoneGap* genérico (para Android) que puede ser usado con cualquier modelo de dominio, más allá del que probamos, en particular, para la aplicación presentada en [1].

4 Resultados Alcanzados y Lecciones Aprendidas

Para lograr que nuestro plugin interactuara con cualquier clase Java (que sea punto de entrada de un modelo), tuvimos que restringir a que los métodos invocados reciban datos simples (por ejemplo, *String*) o a lo sumo colecciones con datos simples. Como mencionamos anteriormente la variable *params* que recibe el plugin al ser invocado es una colección con la forma <tipo, valor>. Esto nos permite transformar el *String* que se recibe en el JSON al tipo correspondiente y así hacer la invocación del método (de la clase Java) con los parámetros con el tipo adecuado. Sin esta especificación todas las clases Java (del modelo a invocar) debían definir sus métodos recibiendo solo *String*. Con esta característica damos más flexibilidad a las clases que son invocadas usando nuestro plugin.

Una desventaja de usar plugins, es que se pierde la característica multiplataforma que propone *PhoneGap*, ya que en nuestro caso solo definimos una solución para Android. Es decir, para brindar soporte multiplataforma se debería desarrollar un plugin específico para cada una de ellas. Esto requiere incorporar conocimientos de cada una de las plataformas para las cuales se quiera desarrollar un plugin. En nuestro caso, al querer usar este desarrollo para el prototipado solo nos limitamos a desarrollarlo para Android.

Nuestro plugin se pudo probar en una aplicación específica cómo se presentó en [1]. Esto nos permitió probarlo con un modelo particular, pero además poder proyectar su uso en otro tipo de aplicaciones como son aquellas aplicaciones móviles educati-

vas⁵ basadas en posicionamiento que es un área que también venimos trabajando. Cabe destacar que la aplicación presentada en [1], además de usar nuestro plugin, hace uso de un plugin provisto por *PhoneGap*, que permite la lectura de códigos de barra (llamado *BarcodeScanner*⁶). Este último plugin, abre la cámara, detecta que se está leyendo un código y devuelve el valor leído. Usamos códigos QR (Quick Response) para determinar dónde está posicionado el usuario y acorde a esto brindarle información o servicios.

Contar con el plugin para Android presentado en este trabajo nos permitirá en un futuro focalizarnos solo en la parte visual y de modelado, mientras que la comunicación entre estos estará dada por nuestro plugin. Esto nos permitirá agilizar los tiempos de prototipado, y a su vez reusar modelos que ya tenemos definidos para dominios específicos. Por otro lado, podremos trabajar con tecnologías HTML, JavaScript y CSS para definir las vistas, sin requerir aprender aspectos particulares de la definición de vistas en Android.

Varios de los prototipos que hemos desarrollamos son probados en lugares donde no hay conectividad (acceso a internet), lo cual, nos llevó a desarrollar aplicaciones clientes, en particular, en Android. Hasta el momento, las aplicaciones que venimos prototipando están haciendo uso solo del posicionamiento del usuario, en un futuro se incorporará el uso de otros sensores. Esto seguramente requerirá un análisis del plugin presentado para lograr también agilizar el uso de dichos sensores. La ventaja que tiene *PhoneGap* en este sentido es que al contar con el concepto de plugin se puede acceder a cualquier aspecto interno del dispositivo, por ejemplo, sensores. Otro aspecto a mencionar, es que las aplicaciones que hemos desarrollado no requieren intercambio de información entre usuarios; esto es un tema a explorar, y también requerirá un análisis del plugin presentado para poder incorporar esta característica.

Es recomendable antes de hacer cualquier desarrollo, evaluar cuáles de las soluciones disponibles se ajusta mejor a los requerimientos de la aplicación que se quiere desarrollar. Se recomiendan lecturas actuales como, por ejemplo, [10] para tener en cuenta otros aspectos que pueden ser críticos para algunos desarrollos como pueden ser el consumo de energía o la seguridad de la información.

6 Conclusiones y Trabajos Futuros

En este trabajo se presentó un plugin de *PhoneGap* para Android que permite una comunicación genérica entre la vista de una aplicación y una clase que funciona como

⁵ Venimos trabajando en el modelado de este tipo de aplicaciones, más información se detalla en Lliteras, A.B.: “*Un enfoque de modelado de Actividades Educativas Posicionadas que contemplan elementos concretos*”. Tesis de Magister en Tecnología Informática Aplicada en Educación, Facultad de Informática (UNLP), 2016

<http://sedici.unlp.edu.ar/handle/10915/50030> (Último Acceso: 13-5-2016)

⁶ Más información sobre el plugin *BarcodeScanner* se detalla en:

<https://build.phonegap.com/plugins/951> (Último Acceso: 13-5-2016)

punto de entrada de un modelo Java. Se describió el funcionamiento del mismo, como así también se indicó que dicho plugin se utilizó en la aplicación *Caminos Alternativos* (la puesta en práctica de la misma se presentó en [1]).

El trabajo presentado es producto de haber detectado ciertos aspectos repetitivos en el prototipado de aplicaciones móviles basadas en posicionamiento, y acorde a eso, lograr agilizar esta tarea, en pos de reducir el tiempo de desarrollo. Además, se buscaba no tener que aprender aspectos puntuales de Android, por ejemplo, cómo definir las vistas; usando *PhoneGap* podemos usar tecnologías como HTML, JavaScript y CSS. En un futuro, con la solución presentada en este trabajo, nos focalizaremos solo en la especificación de las vistas como así también del modelo de la aplicación en sí.

Todavía nos queda explorar otros aspectos como, por ejemplo, la comunicación entre usuarios. Para esto se analizará, por ejemplo, el uso de NFC para que dos usuarios puedan intercambiar información sin requerir conectividad (acorde al tipo de aplicaciones que venimos probando). Otro tema a explorar es el uso de sensores, para esto, se tomará como punto de partida los plugin que ya provee *PhoneGap* para Android. Se analizará si los mismos se necesitan generalizar y cuál es la mejor forma para combinarlos con nuestro plugin genérico. Esto nos permitirá crear aplicaciones móviles sensibles al contexto, por ejemplo, usando sensores internos del dispositivo.

Estamos trabajando para que el plugin presentado nos agilice el empaquetado de aplicaciones móviles basadas en posicionamiento definidas usando la herramienta presentada en [2]. Por ejemplo, agilizar el prototipado de aplicaciones móviles educativas basadas en posicionamiento, para poder realizar pruebas más fácilmente y con distintas características.

Referencias

1. Alconada Verzini, F.M., Tonelli, J.I., Challiol, C., Lliteras, A.B., Gordillo, S.E.: Combing Location-Aware Applications with in-situ Actors Performances. In: the 2015 Workshop on Narrative & Hypertext, pp. 27-31. ACM, New York (2015)
2. Alconada Verzini, F.M., Tonelli, J.I., Challiol, C., Lliteras, A.B., Gordillo, S.E.: Authoring Tool for Location-Aware Experiences. In: the 2015 Workshop on Narrative & Hypertext, pp. 21-25. ACM, New York (2015)
3. Brucker, A. D., Herzberg, M.: On the Static Analysis of Hybrid Mobile Apps. In: Engineering Secure Software and Systems, pp. 72-88. Springer International Publishing (2016)
4. Dalmasso, I., Datta, S. K., Bonnet, C., Nikaen, N.: Survey, comparison and evaluation of cross platform mobile application development tools. In: 9th International Wireless Communications and Mobile Computing Conference, pp. 323-328. IEEE (2013)
5. Fortier, A., Challiol, C., Fernández, J.L., Robles, S., Rossi, G., Gordillo, S.: Exploiting personal web servers for mobile context-aware applications. *The Knowledge Engineering Review* 29(02), 134-153 (2014)
6. Fortier, A., Rossi, G., Gordillo, S., Challiol, C.: Dealing with variability in context-aware mobile software. *Journal of Systems and Software* 83(6), 915-936 (2010)

7. Heitkötter, H., Hanschke, S., Majchrzak, T.A.: Evaluating cross-platform development approaches for mobile applications. In: Web information systems and technologies, pp. 120-138. Springer-Verlag Berlin Heidelberg (2012)
8. Hong, J.Y., Suh, E.H., & Kim, S.J.: Context-aware systems: A literature review and classification. *Expert Systems with Applications* 36 (4), 8509-8522 (2009)
9. Jobe, W.: Native Apps vs. Mobile Web Apps. *International Journal of Interactive Mobile Technologies* (4), 27-32 (2013)
10. Nagappan, M., Shihab, E.: Future trends in software engineering research for mobile apps. In: 23rd IEEE International Conference on Software Analysis, Evolution, and Reengineering. (2016) <http://www.se.rit.edu/~mei/publications/pdfs/Future-Trends-in-Software-Engineering-Research-for-Mobile-Apps.pdf>
11. Palmieri, M., Singh, I., Cicchetti, A.: Comparison of cross-platform mobile development tools. In: 16th International Conference on Intelligence in Next Generation Networks, pp. 179-186. IEEE (2012)
12. Lim, S.: Experimental Comparison of Hybrid and Native Applications for Mobile Systems. *Int. J. Multimed. Ubiquitous Eng* 10(3), 1-12 (2015)
13. Xanthopoulos, S., Xinogalos, S.: A comparative analysis of cross-platform development approaches for mobile applications. In: 6th Balkan Conference in Informatics, pp. 213-220. ACM, New York (2013)

Procesamiento de sentencias NMEA-0183 para el análisis de la geometría satelital utilizando receptores GPS de bajo costo

Alberto Eduardo Riba¹, Jorge Damián Tejada¹,
Nelson Acosta^{2,3}, Juan Manuel Toloza^{2,3}

¹Departamento de Básicas y Aplicadas, Universidad Nacional de Chilecito
9 de Julio 22, Chilecito, La Rioja, Argentina
{ariba, jtejada}@undec.edu.ar

²Universidad Nacional del Centro de la Provincia de Buenos Aires
General Pinto 399, Tandil, Buenos Aires, Argentina
{nacosta, jmtoloza}@exa.unicen.edu.ar

³Universidad Nacional de Tres de Febrero
Mosconi 2736 - Sáenz Peña (B1674AHF), Buenos Aires, Argentina

Abstract. El sistema de navegación por satélites más utilizado en la actualidad es NAVSTAR-GPS. Existen una gran cantidad de dispositivos receptores GPS de bajo costo que pueden ser utilizados para diversas aplicaciones, pero estos no entregan la precisión que algunas de estas necesitan. Este artículo presenta una herramienta que permite analizar las sentencias NMEA obtenidas de uno o múltiples receptores GPS de bajo costo para postular posibles indicadores de la relación entre la geometría satelital y la precisión posicional.

Keywords: GPS, NMEA-0183, geometría satelital, bajo costo, data fusion.

1 Introducción

Un gran número de actividades en la actualidad requieren de la navegación sobre la corteza terrestre y del geo-posicionamiento, algunas con mayor precisión que otras.

La tendencia actual en geo-referenciación radica en la utilización de sistemas GNSS (Global Navigation Satellite System) que utilizan satélites para poder ubicar un receptor en tierra basado en técnicas de triangulación y medición de retardo de la señal [1].

En la actualidad existen dos sistemas GNSS en condiciones totalmente operativas y con cobertura global: NAVSTAR GPS del Departamento de Defensa de EEUU, y GLONASS perteneciente Ministerio de Defensa de la Federación Rusa. Otros dos sistemas se encuentran en desarrollo y se espera que estén en pleno funcionamiento en los próximos años: Beidou-2/COMPASS perteneciente al Gobierno Chino, y Galileo de la Unión Europea - GSA European GNSS Agency.

La tecnología NAVSTAR-GPS es el GNSS más antiguo, conocido y difundido existiendo una gran oferta de dispositivos receptores, desde equipos de uso civil de bajo costo, hasta equipos de nivel profesional que pueden costar miles de dólares, esta diferencia en el precio está dada por la precisión de la posición obtenida.

Los equipos de nivel profesional como los utilizados en instituciones catastrales o el receptor StarFire GPS 3000 de John Deere[2], pueden alcanzar una precisión centimétrica, pero su costo es muy elevado, mientras que los receptores de uso civil son de bajo costo y entregan una precisión en la posición de unos 10 a 15 metros en el 95 % de las veces[3] dependiendo del fabricante.

Existe un gran número de aplicaciones que pueden llevar a cabo su tareas sin inconvenientes en cuanto a la precisión obtenida, pero algunas áreas como la robótica, la agricultura de precisión, la aeronavegación, la navegación marítima, actividades de rescate, entre otras, necesitan una mayor precisión [2], [4], [5].

Actualmente distintos sistemas permiten aumentar la precisión posicional, por ejemplo el DGPS (Differential GPS) [6], AGPS (Assisted GPS) [7], RTK (Real-Time Kinematic) [8] o e-Dif (extended Differential), pero estos sistemas de aumentación no están disponibles en ciertas regiones, son muy costosos de adquirir o implementar y en muchos casos su uso implica un abono oneroso de un servicio mensual.

Por estas razones de costo y disponibilidad es muy importante desarrollar soluciones tecnológicas que cubran estas necesidades. Varias instituciones afrontan a diario el desafío de encontrar nuevas técnicas para mejorar la precisión del posicionamiento, muchos de ellos con resultados exitosos y comprobables.

El presente trabajo se encuentra en el marco de la línea de investigación presentada por los autores en WICC 2016 denominada “Mejora de la precisión posicional utilizando receptores GPS de bajo costo”. Esta línea de investigación pretende reducir la brecha existente entre estos desarrollos y los usuarios finales que necesitan realizar tareas con mayor precisión posicional que la entregada originalmente por el sistema NAVSTAR-GPS, utilizando receptores GPS de bajo costo.

El objetivo de este artículo es presentar una herramienta que permita procesar múltiples archivos con información de sentencias del estándar NMEA-0183 obtenidas de receptores GPS de bajo costo, para relacionar la geometría de los satélites con la precisión posicional.

El artículo se estructura de la siguiente manera, la Sección 2 describe los aspectos generales del estándar NMEA-0183 y señala la estructura de las sentencias que se utilizarán en el trabajo, la Sección 3 presenta trabajos relacionados con esta temática, la Sección 4 muestra la etapa de experimentación determinando los puntos de capturas muestrales, escenarios propuestos y la herramienta para el análisis de la información correspondiente a la geometría de los satélites, finalmente en la Sección 5 se exponen las conclusiones y los trabajos a futuro.

2 Estándar NMEA-0183

La interface estándar NMEA-0183 (Standard For Interfacing Marine Electronic Devices) desarrollada por National Marine Electronics Association, define los requisitos eléctricos de señalización, protocolo de transmisión de datos y formatos de sentencias específicas para una transmisión con formato serial a una velocidad de 4800 baudios. Esta norma soporta la transmisión en un solo sentido desde un único emisor a uno o varios receptores. Los datos transmitidos están codificados en formato ASCII de 7 bits y cada línea es una sentencia que sigue un formato bien definido compuesto por campos separados por comas, los cuales identifican el tipo de información contenida como la posición, la velocidad, la profundidad, entre otros.

Entre las sentencias que entregan los receptores GPS de bajo costo en sus salidas de datos podemos encontrar las siguientes:

- GPRMC (Recommended Minimum Specific): información referida a la posición de latitud y longitud, fecha y hora, variación magnética, etc. Un ejemplo:
\$GPRMC,194421,A,2909.1567,S,06730.2625,W,000.1,095.4,270716,001.3,W*68
- GPGGA (Global Positioning System Fixed Data): información referida a la cantidad de satélites en uso, HDOP, altura, etc. Un ejemplo:
\$GPGGA,194421,2909.1567,S,06730.2625,W,1,07,0.9,1150.4,M,29.6,M,,*72
- GPGSA (GNSS DOP and Active Satellites): información referida al PRN (identificador de satélite) en uso, PDOP, VDOP, HDOP, etc. Un ejemplo:
\$GPGSA,A,3,07,13,30,05,28,09,08,,,,,1.6,0.9,1.3*36
- GPGSV (GNSS Satellites in View): información referida a los satélites en vista elevación, azimut, SNR, etc. Un ejemplo:
\$GPGSV,3,1,12,07,35,139,44,13,21,232,33,30,59,167,43,05,44,238,36*71

Para información más detallada referida a la estructura de estas sentencias se puede consultar en el manual de referencia [9].

De las instrucciones anteriormente mencionadas se procesará la información correspondiente a:

- Fecha: fecha de la captura.
- Hora: hora de la captura.
- Latitud: de la posición con formato GGMM,MMMM (2909.1567).
- Longitud: de la posición con formato GGGMM,MMMM (06730.2625).
- Variación Magnética:
- Satélite en uso: cantidad de satélites usados en la solución de la posición.
- Altitud: altitud de la posición.
- Geoide: valor de geoide de la posición calculada en el área de la muestra.
- PRNs: un arreglo de 12 valores con el PRN de cada satélite usado en la determinación de la posición.
- PDOP: dilución de la precisión en la posición.
- HDOP: dilución de la precisión en la componente horizontal.
- VDOP: dilución de la precisión en la componente vertical.

- Satélites en Vista: cantidad de satélites en vista al momento de la adquisición. Para cada satélite:
 - PRN: identificador del satélite
 - Elev: elevación
 - Az: Azimut
 - SNR: relación señal ruido.

3 Trabajos relacionados

Del conjunto de trabajos relacionados con la temática se destacan los siguientes.

Di Lecce y otros [10] utilizan redes neuronales con un solo receptor GPS consiguiendo una mejora en la precisión de la posición de hasta un 25 %, este sistema de corrección puede aplicarse en dispositivos estáticos y dinámicos.

Tolozá J.M. [11] presenta una metodología y el estudio de diferentes técnicas y algoritmos para el tratamiento de la información entregada por los receptores GPS estándar. Desarrolla una herramienta que permite procesar en crudo las sentencias NMEA-0183 de los receptores GPS utilizados, para su posterior tratamiento con las técnicas y algoritmos propuestos, implementando un sistema diferencial GPS de posicionamiento relativo para mejorar la precisión de las posiciones entregadas por el sistema GPS originalmente. Esta herramienta se basa en la portabilidad de manera que funcione en regiones donde no se cuenta con servicios de aumentación para mejorar la precisión.

Schrader y otros [12], propone una herramienta para mejorar la precisión posicional utilizando las sentencias NMEA-0183 en un sistema de múltiples receptores GPS de bajo costo conectados a microcontroladores de ocho bits. Presenta dos esquemas de hardware uno centralizado y otro descentralizado basándose en cálculos relacionados con el promedio de las latitudes y longitudes, sin recurrir a sensores adicionales ni a complejos cálculos para la obtención de la posición.

De la lectura de estos trabajos hemos identificado que en ninguna de las propuestas se realiza un estudio que relacione los parámetros de la geometría satelital (elevación y azimut) con la precisión posicional obtenida en las mediciones.

4 Experimentación

Para el desarrollo de la herramienta en primer instancia se tomaron diferentes conjuntos de muestras con múltiples receptores GPS. Estas muestras fueron recolectadas en diferentes días, horarios y lugares a fin de tener un muestreo heterogéneo.

4.1 Escenarios

Los escenarios elegidos para las capturas corresponden a puntos geodésicos pertenecientes al Instituto Geográfico Nacional de la República Argentina cuyas

coordenadas son conocidas y precisas, que servirán como punto de referencia para establecer posteriores comparaciones.

Las capturas se realizaron a cielo abierto, sin obstáculos significativos cercanos, ni objetos que pudieran causar alteraciones en la recepción de las señales que degraden la calidad de las mismas, lo cual asegura una mayor cantidad de satélites visibles. Las muestras se tomaron de día y de noche considerando que la ionosfera de noche está menos ionizada.

Durante los experimentos fueron recolectadas 360000 muestras de distintos puntos de datos con distintas cantidades de receptores GPS. En aquellos lugares donde las capturas fueron realizadas con múltiples receptores los datos se podrán fusionar para realizar distintos análisis y cálculos de las geometrías y posiciones obtenidas.

4.2 Equipamiento utilizado

Para realizar las capturas se utilizaron tres receptores GPS marca Garmin conectados a una notebook. El costo de estos receptores no supera los 100 dólares.

El dispositivo incluye un receptor y antena embebidos. Puede trackear múltiples satélites (12 como máximo) al mismo tiempo, proveyendo una actualización de datos de navegación con una frecuencia de 1 Hz. Su consumo es bajo e incluye también la capacidad de trabajar con el sistema de aumentación WAAS de GPS diferencial para aumentar la precisión a 3 metros. El problema que se presenta es que este sistema de aumentación no funciona en nuestra región por la ausencia de señal de los satélites geostacionarios. Este receptor está diseñado para trabajar bajo condiciones extremas e incluso es resistente al agua pudiendo ser sumergido a 1 metro durante 30 minutos.

Tiene una memoria flash interna que le permite retener datos críticos como los parámetros orbitales de los satélites, la última posición conocida, fecha y hora; provee una salida de datos estandarizada NMEA-0183; están disponibles los drivers para Windows y Macintosh; la tensión de trabajo está entre los 4.4 y 5.5 volts; la precisión en modo normal, es menor a los 15 metros el 95% de las mediciones, especificaciones aportadas por el fabricante.

4.3 Herramienta Propuesta

Para el análisis de los datos obtenidos de los distintos receptores se desarrolló una herramienta que permite tomar uno o varios archivos con formato NMEA y analizar la evolución de la geometría de los satélites. A cada archivo capturado se le asigna un nombre y una forma con un color para poder identificarlos en el gráfico cuando estos sean consultados. La Figura 1 muestra la pantalla para la selección de los archivos.

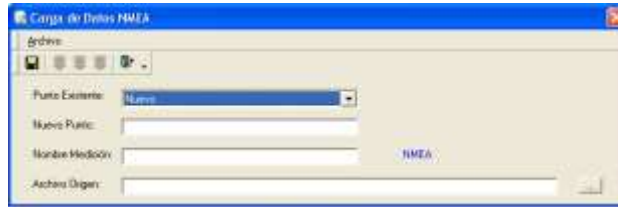


Fig. 1. Interfaz de selección de archivos

Importados los archivos con las sentencias a ser procesadas, se podrán establecer distintos filtros como por ejemplo la fecha, hora de inicio y la hora de fin, la cantidad de satélites en uso, la relación señal ruido y los parámetros de dilución de precisión, PDOP, VDOP y HDOP. Estos filtros permiten eliminar tuplas que pueden contener errores.

Establecidos los filtros la herramienta muestra una grilla con una lista que detalla la hora de la medición con su latitud y longitud y seleccionando una tupla se muestra en un gráfico con el mapa de cielo de la medición como muestra la Figura 2. Haciendo doble click sobre una fila determinada muestra el resto de la información relacionada con esa medición.

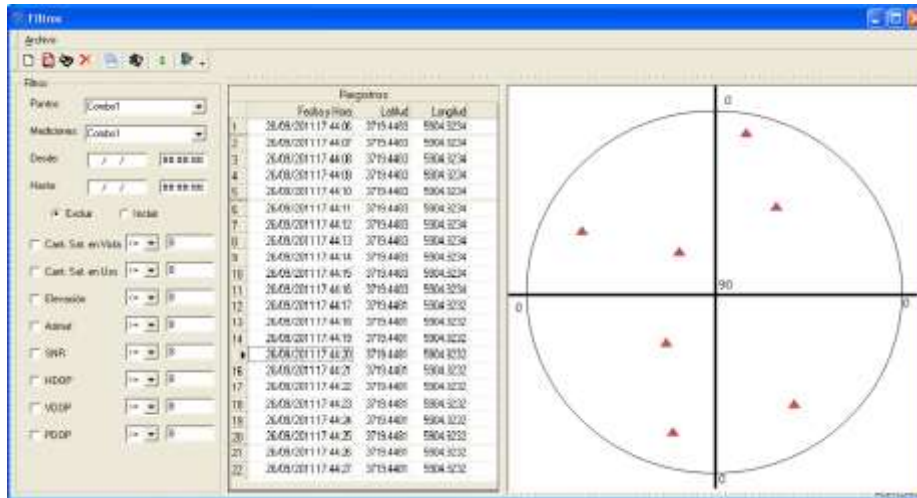


Fig. 2. Interfaz de visualización de la geometría de los satélites

La posición estimada se calcula a través de la media aritmética, la moda y la determinación del par de puntos que más se repite

La herramienta presenta una interfaz avanzada para que un usuario experto personalice la búsqueda de información según ciertos criterios que no puedan estar contemplados en los filtros propuestos.

5 Conclusiones y Trabajos a Futuro

La herramienta propuesta organiza y ordena los datos provenientes de las sentencias NMEA permitiendo que estos puedan ser manipulados de manera dinámica, ágil y visual, facilitando el análisis de la información obtenida relacionada con la geometría satelital.

El procesamiento de datos en base a los ensayos realizados permitió postular posibles indicadores de la relación entre la calidad de la geometría satelital y la precisión posicional.

La dispersión satelital es uno de estos posibles indicadores y se relaciona con los parámetros de elevación y azimut de cada satélite trackeado presente en las sentencias NMEA obtenidas de los receptores GPS.

En trabajos a futuro, se proyecta determinar nuevos indicadores y calcular un coeficiente que permita evaluar la dispersión satelital, con el fin de mejorar la precisión posicional entregada por los receptores GPS sin comprometer el costo computacional, para que pueda ser implementado sobre un microcontrolador.

6 Referencias

1. Gleason S., Gebre-Egziabher D.: GNSS Applications and Methods. Artech House, 508 pp. (2009).
2. Tomkiewicz, S. M., Fuller, M. R., Kie, J. G., Bates, K. K., Global positioning system and associated technologies in animal behaviour and ecological research, *Philosophical Transactions of the Royal Society B: Biological Sciences* 365 (1550) 2163–2176. (2010).
3. Arnold, Lisa. L., Zandbergen, Paul. A.: Positional accuracy of the Wide Area Augmentation System in consumer-grade GPS units. *Computers & Geosciences Volume 37 Issue 7*, Elsevier, pp. 883-892. (2011).
4. Cui, Y., Ge, S. S.: Autonomous vehicle positioning with gps in urban canyon environments, *Robotics and Automation, IEEE Transactions on* 19 (1) 250 15–25 (2003).
5. Elnabwy, M. T., Kaloop, M. R., Elbeltagi, E.: Talkha steel highway bridge monitoring and movement identification using rtk-gps technique, *Measurement* 46 (10) 4282–4292. (2013)
6. Clarke, Bill: *Aviator's Guide to GPS*, 3rd Edition, McGraw-Hill, p [8-20 and 47-55] 1998.
7. Van Diggelen, Frank: "A-GPS, Assisted GPS, GNSS, and SBAS", p [1-5]. (2009).
8. Dardari, Davide, Falletti, Emanuela and Luise, Marco: *Sattellite and Terrestrial Radio Positioning Techniques: A Signal Processing Perspective.*, 1st Edition, Elsevier, p[62-74] (2012).
9. Garmin International, Inc.: *GPS 18x Technical Specifications*. (2008).
10. Di Lecce, V., Amato, A., y Piuri V.: Neural technologies for increasing the GPS position accuracy, *IEEE International Conference on Computational Intelligence for Measurement Systems and Applications*, pp. 4–8, (2008).
11. Toloza, Juan Manuel. "Algoritmos y técnicas de tiempo real para el incremento de la precisión posicional relativa usando receptores GPS estándar". SEDICI, Universidad Nacional de La Plata. (2012).
12. Schrader, D.K., Min, B.-C., Matson, E.T.: Real-time averaging of position data from multiple GPS receivers, *Measurement* (2016).

Reparación eficiente de estructuras de datos en tiempo de ejecución basada en SAT

Marcelo Uva¹, Pablo Ponzio^{1,2}, Germán Regis¹, and Nazareno Aguirre^{1,2}

¹ Departamento de Computación, FCEFQyN, Universidad Nacional de Río Cuarto, Río Cuarto, Argentina. Email:

{uva,pponzio,gregis,naguirre}@dc.exa.unrc.edu.ar

² Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

Resumen Las fallas de un programa pueden producir estados internos inconsistentes, cuya propagación puede provocar la terminación abrupta del programa, la pérdida de datos del usuario, la incapacidad de realizar alguna tarea, etc. La reparación de estructuras de datos consiste en reemplazar estos estados inconsistentes por estructuras generadas a partir de especificaciones formales, utilizando algún procedimiento de decisión, que permitan al programa continuar su ejecución sin mayores problemas. En este trabajo se presenta un enfoque para resolver este problema basado en SAT solving, cuya característica distintiva es el aprovechamiento de dos técnicas del estado del arte para mejorar su eficiencia y escalabilidad: el uso de rotura de simetrías y de cotas ajustadas. La evaluación experimental preliminar realizada muestra que nuestro enfoque es más eficiente y es capaz de lidiar con estructuras de mayor complejidad que técnicas relacionadas.

1. Introducción

La constante evolución y expansión de los sistemas de software en innumerables dispositivos, ha generado la necesidad de contar con aplicaciones cada vez más confiables. Actualmente es posible encontrar software en teléfonos, cámaras digitales, televisores, dispositivos de localización satelital, aviones, electrodomésticos, etc. Todo esto ha generado un especial interés en las áreas de investigación vinculadas a los métodos formales e ingeniería de software. Un alto porcentaje de los recursos asignados a los proyectos de desarrollo de software son destinados a tareas de detección y corrección de fallas. Los problemas se agravan si el sistema ya se encuentra instalado y funcionando. La corrección de fallas en estos casos involucrará una detención del sistema, su corrección y posterior puesta en marcha. Es posible que el sistema no pueda ser detenido durante un período determinado de tiempo por lo que, en estos casos, sea preferible “soportar” un leve deterioro del sistema a someterlo al proceso de corrección descripto.

Una alternativa al procedimiento anterior es aplicar técnicas automáticas, basadas en especificaciones, que permitan restablecer el sistema a un estado consistente luego de la aparición de una falla. En general, estas técnicas utilizan

un procedimiento de decisión para generar un estado que satisface las especificaciones (invariantes, pre y pos condiciones), que sirve de reemplazo al estado del sistema inconsistente producido por un defecto. En particular, diversos trabajos han atacado la reparación de programas que manipulan estructuras de datos dinámicas, como listas, árboles binarios, etc. [1,2,3,4]

En este trabajo, se propone una técnica de reparación de estructuras de datos en tiempo de ejecución basada en satisfactibilidad booleana (SAT). La técnica es aplicada una vez que se ha detectado una estructura que viola un invariante o la poscondición luego de la ejecución de un método (y corresponde a una falla), y emplea procedimiento de decisión basado en SAT (Alloy [5]) para generar una estructura concreta válida respecto al invariante y la poscondición del método erróneo.

Existen en la literatura diversos trabajos vinculados a la reparación de estructuras de datos basada en SAT. En menor o mayor medida, estos enfoques presentan problemas de escalabilidad (como la mayoría de los análisis basados en SAT), y se vuelven ineficientes cuando las estructuras crecen en tamaño y complejidad. Los investigadores han abordado este problema de diferentes maneras. Uno de los trabajos que ha logrado lidiar mejor con los problemas de escalabilidad es el presentado en [1], implementado en la herramienta Cobble, que propone aprovechar la información de la ejecución del programa que origina la falla. Este enfoque consiste en almacenar información sobre los campos de la estructura a reparar accedidos por el programa, e intentar modificar sólo dichos campos en una primera instancia de reparación, acotando así el espacio de búsqueda de estructuras. Si este primer intento es exitoso, el enfoque obtiene ganancias sustanciales en tiempo de ejecución y escalabilidad respecto de enfoques previos [1]. Sin embargo, en situaciones en las que la reparación implica modificar partes de la estructura que no han sido accedidas por el programa, Cobble simplemente invoca a un procedimiento tradicional de generación de estructuras basado en SAT solving, y su escalabilidad se ve seriamente afectada (ver sección 2).

En este trabajo, presentamos un enfoque más general para lidiar con los problemas de escalabilidad de la reparación de estructuras de datos, que involucra la utilización de dos técnicas de optimización recientes de análisis de programas basados en SAT. Estas técnicas son: la canonización del heap para eliminar estructuras “isomorfas” y así acelerar el análisis (comúnmente llamada rotura de simetrías), y el uso de cotas ajustadas para eliminar variables proposicionales innecesarias de la fórmula proposicional a analizar mediante SAT solving. Estas técnicas fueron introducidas originalmente en [6], y aplicadas a diversos tipos de análisis de programas basados en SAT [7,8]. La contribución principal de este trabajo es la adaptación de estas técnicas al contexto novedoso de reparación de estructuras de datos en tiempo de ejecución, y una evaluación experimental preliminar que muestra los beneficios del uso de estas técnicas en este contexto.

Cabe destacar que nuestra técnica permite mejorar la performance y la escalabilidad de la reparación en general, a diferencia de Cobble, cuyas ventajas sólo se dan bajo ciertas circunstancias (como se discute en 2 y como evidencia

nuestra evaluación experimental). También es importante mencionar que Cobler y nuestro enfoque son complementarios, y podrían utilizarse conjuntamente para producir mejores resultados.

El resto del trabajo se organiza como sigue. En la sección 2, se describe el lenguaje de especificación Alloy, que se utiliza aquí como lenguaje de especificación y como procedimiento de decisión, y el enfoque de reparación de Cobler. En la sección 3, se presenta nuestra técnica, y se describen la rotura de simetrías y el uso cotas ajustadas como optimizaciones para mejorar la eficiencia y la escalabilidad de la reparación. La sección 4 presenta un análisis experimental preliminar de nuestra técnica, comparada con la presentada en [1]. En 5 se discuten trabajos relacionados, y en 6 se presentan las conclusiones y posibles trabajos futuros.

2. Preliminares

2.1. Alloy

Alloy[5] es un lenguaje de especificación declarativo basado en la lógica relacional, una extensión de la lógica de primer orden con operadores como composición y clausura transitiva de relaciones. Alloy es un lenguaje simple, fácil de aprender, y posee una herramienta de software asociada para analizar especificaciones. Esta herramienta, llamada Alloy Analyzer, implementa un análisis automático de las especificaciones Alloy, basado en SAT solving, y es de libre disponibilidad.

Alloy posee un gran poder expresivo, por lo que resulta útil para especificar estructuras de datos con invariantes complejos, como así también los programas que las manipulan (pre y postcondiciones por ejemplo). Por este motivo, y por el análisis eficiente que implementa el Alloy Analyzer, Alloy se ha usado como lenguaje de especificación y como backend de análisis en numerosos trabajos, en particular, en [1] y en este trabajo.

2.2. Reparación de estructuras de datos basada en SAT, Cobler

En esta sección se discute la reparación de estructuras de datos basada en SAT, junto con el enfoque de optimización llamado Cobler [1], a través de un ejemplo. En la Figura 1 se muestra un fragmento de una implementación tradicional de listas simplemente encadenadas en el lenguaje JAVA. Esta implementación posee un método `addFirst` para agregar un nuevo elemento a la cabeza de la lista. Además, la implementación posee contratos declarativos, especificados en Alloy. El invariante de la clase postula que las listas deben ser acíclicas, y que su tamaño es igual a la cantidad de nodos alcanzables a partir de la cabeza de la lista. En la figura también se muestra (parte de) la poscondición de `addFirst`. Esta indica que luego de la ejecución del método el tamaño de la lista debe ser al tamaño previo a la ejecución más uno (notar que se utiliza la convención usual de que `this` y `this'` representan el estado anterior y posterior a la ejecución del método de `this`, respectivamente).

```

public class Nodo {
    int value;
    Nodo next;
}

public class Lista {
    /*
    Invariante:
    all l: Lista, n: Nodo |
        ((n in l.head.*next => n not in n.^next) and
        l.size = #l.head.*next))
    */
    Nodo head;
    int size;
    /*
    Post:
    . . .
    this'.size = this.size + 1
    */
    public void addFirst(int i) {
        Nodo n = new Nodo();
        n.value = i;
        n.next = this.head;
        this.head = n;
        // size++; ERROR: linea omitida por el programador
    }
}

```

Figura 1. Programa de ejemplo: `addFirst` de listas simplemente encadenadas

El método `addFirst` de la figura contiene un defecto: el programador olvidó incrementar el campo `size`, que representa la cantidad de elementos de la lista. Así, ejecutar `addFirst(1)` sobre la primera lista (de arriba hacia abajo) de la figura 2 resulta en la segunda lista de la misma figura: un estado inconsistente que viola el invariante de listas al poseer tres elementos pero tener un tamaño de 2. Una implementación correcta de `addFirst` (que satisfaga la postcondición del método) debería retornar la tercera lista de la figura 2. Una reparación basada en SAT solving para este caso consistiría en generar una especificación Alloy que incluya la primera lista de la figura 2, la postcondición de `addFirst(1)` y el invariante de listas. Alimentando el Alloy Analyzer con esta especificación y los scopes apropiados (límites en la cantidad de nodos y número máximo de enteros, debido a que el análisis implementado por el Alloy Analyzer es acotado), la herramienta produce como output la tercera lista de la figura 2, es decir, una reparación que satisface el invariante de `Lista` y la postcondición de `addFirst()`. Esta lista puede usarse para reemplazar el estado inconsistente producido por la ejecución del defecto, y continuar normalmente con la ejecución de la aplicación

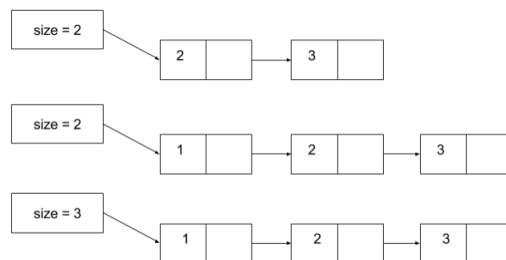


Figura 2. De arriba a abajo: entrada de `addFirst(1)`, salida incorrecta de `addFirst(1)`, posible reparación generada con Alloy

que invoca a `addFirst`. Lo descrito anteriormente es el enfoque tradicional de reparación de estructuras de datos basado en SAT, presentado en [2].

Sin embargo, como es usual con los enfoques de análisis basados en SAT, el enfoque anterior sufre de problemas de escalabilidad cuando se aplica a estructuras complejas, y/o cuando se desean reparar estructuras de mayor tamaño (es decir, incrementar los scopes). Una de las propuestas más exitosas para lidiar con este problema es la implementada en la herramienta Cobbler [1]. Cobbler se basa en la observación de que en muchas ocasiones es posible obtener una estructura correcta (reparación) mediante SAT manteniendo fija la parte de la estructura no “visitada” por el código, y permitiendo al SAT solver modificar sólo los campos accedidos por el código. De esta manera, en muchos casos se logra reducir ampliamente el espacio de búsqueda del SAT solver, acelerando su performance y permitiendo reparar estructuras de tamaño mucho mayor.

Cobbler instrumenta el código (`addFirst` en nuestro ejemplo) para almacenar los campos accedidos y modificados de las estructuras durante la ejecución. El algoritmo de reparación de Cobbler consta de tres etapas. Primero, Cobbler intenta obtener una reparación permitiendo al SAT solver cambiar sólo los campos modificados por la ejecución particular del código. Si esto no es posible, el siguiente paso generar una estructura correcta permitiendo la modificación de los campos modificados y los accedidos durante la ejecución. Si ninguno de los dos pasos anteriores resulta exitoso, Cobbler utiliza el UNSAT core para determinar cuales campos deben ser modificados para obtener una reparación. En nuestra experiencia con la herramienta, Cobbler es muy eficiente si logra reparar la estructura en las primeras dos etapas. Sin embargo, en la tercera etapa por lo general Cobbler permite modificar la estructura completa (o casi completa), con lo cual en este punto se comporta de manera muy similar a los enfoques tradicionales de reparación basada en SAT. Este es el caso en el ejemplo de la figura 1, y en muchos otros ejemplos prácticos relevantes.

A diferencia de Cobbler, nuestro enfoque apunta a mejorar la eficiencia de la reparación en el caso general.



Figura 3. Dos listas isomorfas

3. Un enfoque eficiente para la reparación de estructuras de datos basada en SAT

En esta sección se describe la principal contribución de este trabajo: la aplicación de rotura de simetrías y cotas ajustadas a la reparación de estructuras de datos en tiempo de ejecución basada en SAT, con el objetivo de aumentar la eficiencia y la escalabilidad de la reparación a estructuras de mayor tamaño y complejidad. Estos enfoques se explican a continuación.

Es importante destacar que la detección de fallas durante la ejecución del programa y la recuperación de las fallas son problemas diferentes, muy complejos por sí mismos, y no se abordarán aquí. Para este propósito podría utilizarse el framework Armor [14], que monitorea la ejecución del programa y en caso de aparición de una falla es capaz de reemplazar estados inconsistentes por estados correctos, como los provistos por nuestra técnica, y continuar con la ejecución normal del programa a partir de los mismos.

3.1. Rotura de Simetrías

Un problema del modelo Alloy de listas presentado en la sección 2 es que admite estructuras isomorfas, es decir, dos o más estructuras concretas que representan una misma estructura abstracta, y son indistinguibles desde el punto de vista de la reparación.

Por ejemplo, si consideramos 3 nodos e identificadores N_0 , N_1 y N_2 para los mismos, las listas de la figura 3 son instancias válidas de nuestra especificación de la sección anterior. Sin embargo, estas estructuras son isomorfas, ya que ambas poseen los mismos valores en el mismo orden, y por lo tanto representan la misma lista (los identificadores de nodos usualmente se consideran abstracciones de direcciones de memoria, y en lenguajes orientados a objetos como JAVA la dirección de memoria donde están alojados los nodos no es importante).

En [6] se propone canonizar el heap para eliminar instancias isomorfas, instrumentando de manera automática las especificaciones con predicados de rotura de simetrías. Estos predicados asignan identificadores a los nodos de las estructuras en orden creciente (N_0 , N_1 , N_2 , etc.) siguiendo un recorrido primero en anchura. Intuitivamente, una vez llevada a cabo la instrumentación de la especificación Alloy con predicados de rotura de simetrías, la lista que se muestra a la izquierda en la figura 3 se convierte en la representación canónica de las listas de la figura, siendo ahora la lista de la derecha una instancia no permitida

de la especificación. La evaluación experimental de [6] indican que la reducción de la cantidad de instancias válidas producto de la rotura de simetrías permite mejorar sustancialmente la escalabilidad y la eficiencia de la verificación acotada de programas basada en SAT.

3.2. Cotas Ajustadas

El modelo Alloy de listas de la sección 2 posee un invariante de representación que establece que todas las listas deben ser acíclicas. Ahora bien, bajo el supuesto de que se desea generar un estado de reparación para un método de listas donde se ha producido una falla y donde además las especificaciones cuentan con predicados de rotura de simetrías, el conjunto dominio para el campo *next* contendrá los pares de nodos (N_i, N_j) o (N_i, null) , donde el par (N_i, N_j) representa que $N_i.\text{next} = N_j$, para $0 \leq i, j \leq 2$, fijando un scope de 3 para la signatura *Nodo*. Sin embargo, el invariante de representación afirma que sólo las listas acíclicas son estructuras válidas. Por lo tanto, muchos de los valores posibles en el dominio de *next* nunca podrán darse en una reparación válida. Decimos que un par (N_i, N_j) pertenece a la cota ajustada del campo *next* si $N_i.\text{next} = N_j$ en alguna instancia que satisface el invariante de representación. Así, estos valores de campos pueden eliminarse de la codificación del problema de reparación como una fórmula proposicional, simplificando la fórmula y haciéndola más sencilla de analizar mediante SAT solving.

En esta aplicación particular, el cómputo de cotas ajustadas consiste en computar el conjunto de valores que efectivamente puede tomar un campo bajo las restricciones del invariante de la estructura. Para nuestro ejemplo con 3 nodos, los valores permitidos por el invariante para el campo *next* (y la canonización del heap) definen la siguiente cota ajustada:

$$\text{cota_next} = \{(N0, \text{null}), (N1, \text{null}), (N2, \text{null}), (N0, N1), (N1, N2)\}$$

De esta manera, de 12 posibilidades para la codificación del dominio del campo *next*, la cantidad de valores a representar se reduce a 5 si se dispone de la cota ajustada anterior.

Observar que, mientras más precisa sea la cota ajustada, mayor será el beneficio para el análisis basado en SAT, ya que se podrán eliminar mayor cantidad de variables proposicionales de la fórmula proposicional que codifica el problema de reparación. En [12] se definen una serie de algoritmos para computar cotas ajustadas de manera eficiente. Estos algoritmos fueron los utilizados para computar las cotas ajustadas en este trabajo.

Notar que el cómputo de cotas ajustadas es relativamente costoso computacionalmente, pero en este contexto se puede efectuar previamente a la instalación (deployment) del sistema. Por esta razón, el tiempo de cómputo de cotas ajustadas no se tiene en cuenta en nuestra evaluación experimental de la sección 4.

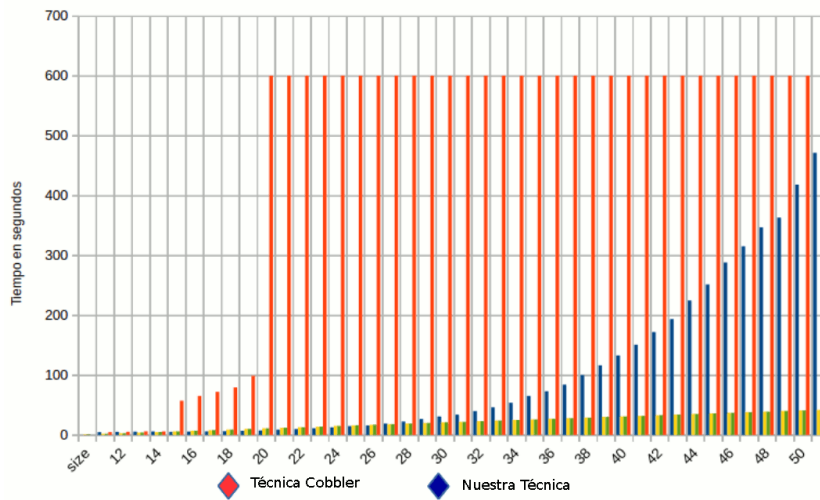


Figura 4. Comparación experimental entre Cobbler y nuestro enfoque.

4. Evaluación experimental de la técnica

Con el objetivo de evaluar nuestra técnica y comparar su rendimiento con Cobbler[1], se han realizado una serie de pruebas experimentales. Nuestra técnica fue implementada mediante un prototipo desarrollado en Python haciendo uso del Alloy Analyzer. Todos los experimentos fueron ejecutados sobre una PC con un procesador Intel(R) Core(TM) i5-4460 CPU 3.40Ghz con 8GB de RAM corriendo un sistema operativo GNU/Linux 3.2.0.

El caso de estudio seleccionado ha sido *listas simplemente encadenadas* y el método a reparar `addFirst()` el cual cuenta con el error de no incrementar el campo `size` (como se discute en la sección 2). Para probar la efectividad de nuestra técnica se generaron aleatoriamente 100.000 listas simplemente encadenadas por medio de Randoop[13] (herramienta de generación automática de casos de tests basada en aleatoriedad), de las cuales fueron seleccionadas aleatoriamente 41 listas de tamaño entre 11 y 51. Estas listas se utilizaron como entradas para `addFirst()` (con el defecto mencionado), para producir los estados erróneos que se utilizarán para medir la eficiencia y la escalabilidad de la reparación, utilizando Cobbler y nuestro prototipo. En ambos casos se estableció un tiempo máximo de reparación de 10 minutos, y se interrumpieron los experimentos que superaron este límite.

La figura 4 resume los resultados de los experimentos. En ella puede observarse que nuestra técnica logra reparar listas de hasta 51 elementos, mientras que Cobbler alcanza el timeout de 10 minutos sin lograr una reparación para todas las listas de 19 elementos o más. Por otro lado, Cobbler repara listas con un máximo de 19 elementos en 98,79 segundos, versus los 6,89 segundos que tarda nuestra técnica, esto es, nuestra técnica es un orden de magnitud más rápida en

este caso. Esta evaluación preliminar muestra resultados prometedores respecto de la eficiencia y la escalabilidad de nuestra técnica.

Cabe destacar que este es un caso difícil para Cobble, ya que las estructuras resultantes debido al defecto en `addFirst()` no pueden repararse modificando sólo los campos accedidos por `addFirst()`. Si bien pueden existir casos en los que Cobble sea más eficiente que nuestro enfoque (cuando el código accede a pocos campos, y es posible obtener una reparación cambiando sólo estos campos), es importante mencionar que Cobble y las técnicas de rotura de simetrías y uso de cotas ajustadas utilizadas por nuestro enfoque pueden combinarse para obtener los beneficios de ambas a la vez.

5. Trabajos Relacionados

Los trabajos relacionados de reparación más relevantes fueron discutidos anteriormente en nuestra sección de preliminares.

Nuestro trabajo utiliza cotas ajustadas para optimizar el análisis. Estas han sido explotadas exitosamente en trabajos previos pero en otros contextos. Por ejemplo, para la generación de casos de test basados en SAT [7], para la generación de casos de test mediante model checking acotado y ejecución simbólica [8], y para la verificación exhaustiva acotada de código [6]. En todos estos casos, las cotas ajustadas han sido de suma importancia para mejorar la eficiencia y la escalabilidad de las técnicas.

La rotura de simetrías para SAT solving fue introducida en [6]. Sin embargo, esta técnica ha sido aplicada a muchos otros contextos anteriormente, como por ejemplo, en model checking [9] [10], o en la generación de casos de tests basada en invariantes operacionales [11].

6. Conclusiones y trabajos futuros

En este trabajo, se ha presentado una técnica basada en SAT de reparación de estructuras de datos dinámicas en tiempo de ejecución. Si bien, como se menciona en las primeras secciones, existen algunos trabajos en la literatura en donde se aplica SAT con estos fines, nuestro enfoque realiza contribuciones en cuanto a mejorar escalabilidad en estructuras de datos de mayor tamaño y complejidad. Esto se ha logrado incorporando las técnicas de rotura de simetrías y uso de cotas ajustadas, aplicadas anteriormente a distintos análisis de software como generación automática de casos de test y verificación acotada, al contexto novedoso de reparación de estructuras de datos dinámicas. Esto agrega evidencia de que la rotura de simetrías y en particular las cotas ajustadas, pueden aprovecharse fuera de análisis de código más tradicionales como el testing y la verificación, y alienta la investigación de nuevos contextos de aplicación que puedan beneficiarse de estas técnicas.

La evaluación experimental preliminar realizada muestra que nuestra técnica tiene el potencial de superar ampliamente a Cobble. Estos escenarios desfavorables para Cobble son relativamente frecuentes en la práctica (el ejemplo utiliza-

do es un error común de programación), lo que muestra la relevancia práctica de nuestra técnica. Es sencillo imaginar otros escenarios problemáticos para Cobble, como por ejemplo, muchas veces la parte de la estructura visitada por el código debe modificarse si la estructura actual debe rebalancearse para satisfacer su invariante de representación (en AVLs, árboles rojos y negros, etc.). Como trabajo futuro se llevará a cabo una comparación extensiva de estas técnicas, lo que permitirá determinar sus virtudes y sus desventajas con mayor precisión.

Respecto a los escenarios favorables a Cobble, en tales casos ambas técnicas podrían usarse en conjunto, potenciando sus ganancias. En otras palabras, las optimizaciones de rotura de simetrías y aprovechamiento de cotas ajustadas pueden incorporarse en Cobble, mejorando su performance. Nuestro objetivo aquí fue mostrar la importancia de estas optimizaciones, debido a nuestra observación de que Cobble no siempre es exitoso en mejorar la eficiencia de la reparación. Evaluar la incorporación de las optimizaciones mencionadas en Cobble se deja como trabajo futuro.

Referencias

1. R. Zaeem, D. Gopinath, S. Khurshid. "History-Aware data structure repair using SAT". Proceeding TACAS'12 Proceedings of the 18th international conference on Tools and Algorithms for the Construction and Analysis of Systems Pages 2-17
2. B. Elkarablieh; I. Garcia; Y.L. Suen, S. Khurshid, S. " Assertion-based repair of complex data structures". ASE (2007).
3. F. Zaeem; R. Nokhbeh; S. Khurshid "Contract-Based Data Structure Repair Using Alloy". ECOOP 2010.
4. H. Samimi; H. Aung; E. Millstein. "Falling Back on Executable Specifications". ECOOP 2010.
5. Jackson, D. Alloy: a lightweight object modeling notation. ACM Transactions on Software Engineering and Methodology 11, 2, 2002, 256-290.
6. J.P. Galeotti, N. Rosner, C. Lopez Pomboy M. Frias. "Analysis of invariants for efficient bounded verification" ISSSTA 2010. ACM.
7. P. Abad; N. Aguirre; V. Bengolea; D. Ciolek; M. Frias et al. "Improving Test Generation under Rich Contracts by Tight Bounds and Incremental SAT Solving. 2013 IEEE International Conference on Software Testing, Verification and Validation.
8. N. Rosner; J. Geldenhuys; N. Aguirre; W. Visser; M.Frias. "BLISS: Improved Symbolic Execution by Bounded Lazy Initialization with SAT Support." IEEE Trans. Software Eng. 41(7): 639-660 (2015)
9. R. Iosif, "Symmetry reduction criteria for software model checking". 9th International SPIN Workshop on Model Checking of Software. 2002.
10. M. Musuvathi and D. L. Dill, "An incremental heap canonicalization algorithm". 12th international conference on Model Checking Software, SPIN, 2005.
11. C. Boyapati, S. Khurshid, and D. Marinov, "Korat: automated testing based on java predicates" International Symposium on Software Testing and Analysis 2002.
12. P. Ponzio, Tesis doctoral.
[http : //digital.bl.fcen.uba.ar/Download/Tesis/Tesis_5547_Ponzio.pdf](http://digital.bl.fcen.uba.ar/Download/Tesis/Tesis_5547_Ponzio.pdf)
13. C. Pacheco and M. Ernst. "Randoop: Feedback-directed Random Testing for Java." OOPSLA 2007 Companion, Montreal, Canada, Oct. 2007, ACM.
14. A. Carzaniga; A. Gorla y otros. "Automatic Recovery from Runtime Failures". ICSE 2013.

Principios, Roles y Métricas en alineamiento estratégico de nuevos requerimientos utilizando DevOps

Alberto Belalcázar¹, Javier Díaz¹, Lía Molinari¹, Christian Rodríguez¹

¹LINTI – Facultad de Informática – Universidad Nacional de La Plata, UNLP
albertob@tesista.linti.unlp.edu.ar, javierd@info.unlp.edu.ar, lmolinari@info.unlp.edu.ar,
car@info.unlp.edu.ar

Abstract. En un entorno de usuarios cada vez más exigentes y ansiosos de los servicios que les brinda la tecnología, las áreas de tecnología de la información y las comunicaciones (TICs) adquieren un papel estratégico en las organizaciones debido al apoyo y aporte que brindan a los usuarios. Las aplicaciones que utiliza cotidianamente, en forma ubicua, obliga a las áreas de TICs a dinamizar y acelerar los modelos tradicionales de desarrollo y entrega de aplicaciones, con estrategias que logren disminuir la brecha de alineamiento entre las áreas de desarrollo y operaciones. DevOps¹ ha surgido como una nueva tendencia orientada hacia la obtención de una relación de trabajo y colaboración entre áreas de desarrollo y operaciones de TICs. En este documento se presenta una hoja de ruta que permita alinear Organización-DevOps-Usuarios, y asegurar que los beneficios de aplicar las buenas prácticas de DevOps ayuden a superar la brecha entre operaciones con desarrollo. En este contexto se analizan principios, métricas, fronteras y roles y se presenta un caso de estudio realizado en estudiantes del ámbito universitario.

Keywords: DevOps, Integración Continua, Entrega Continua, Despliegue Continuo, Tiempos y Procesos Estratégicos, Procesos Sensibles.

1 Introducción

En una nueva concepción integradora promovida por los trabajos colaborativos entre desarrollos/operaciones, la automatización de tareas es una de las metas más deseadas y complejas.

El flujo de trabajo tradicional de desarrollo y operaciones en TI, en el que interviene personal con sus roles y responsabilidades se resume de la siguiente manera, haciendo una pronunciada simplificación: desarrollo produce sus programas, luego, estos son testeados y desplegados a producción sea en forma manual o automática.

Internamente el ciclo desarrollo-testing producción/despliegue puede estar formal o informalmente definido y ser efectivo en cuanto a la determinación de las funciones y responsabilidades involucradas entre desarrolladores y operadores.

¹ <http://devops.com/>

Sin embargo, cuando la aplicación se encuentra en producción, el usuario puede encontrar fallos o que no cumple con lo esperado por diferentes características que pueden ser la accesibilidad, el tiempo de respuesta, o relacionadas con la ubicuidad.

La detección temprana de los incidentes y una respuesta ágil desde TI, permitiría reducir el tiempo para solucionar el incidente y cerrarlo a satisfacción del usuario.

Al no existir una comunicación fluida entre las áreas de desarrollo de aplicaciones y operaciones, se pueden presentar fallos que ponen en riesgo la imagen de TI.

Para definir una hoja de ruta que permita alinear Organización-DevOps-Usuarios, es necesario realizar un análisis conceptual de componentes DevOps, identificando su ubicación y alcance en el marco de desarrollo y despliegue de software. La granularidad de componentes DevOps que permita identificar los Principios y las Fronteras de los Roles de DevOps, ayuda a determinar las actividades que pueden retrasar el avance de los procesos y flujos de trabajo colaborativo involucrados en el desarrollo y despliegue de software. Las métricas que se toman como referencia son parte de los conceptos DevOps cuyo análisis en un estudio de caso propuesto ayuda a establecer el factor determinante que tiene la relación de apoyo colaborativo de DevOps en el cumplimiento de objetivos con el desarrollo y despliegue de automatización de tareas corporativas.

2 Componentes DevOps

DevOps (Developer-Operators) es un conjunto de prácticas que permiten fomentar la colaboración y relación de trabajo entre desarrollo y operaciones de TI. DevOps es una mezcla de patrones destinados a mejorar la colaboración entre el desarrollo y operaciones [1][2][3][4][5][6][14]. El objetivo de esta colaboración es mejorar los tiempos de respuesta del ciclo de requerimiento-satisfacción del usuario, mediante aplicaciones y medios que ayuden a alinearlos y cerrar las brechas entre desarrolladores y operadores, aplicando prácticas ágiles en las áreas de desarrollo y operaciones[10][12].

La arquitectura en la que se desenvuelve DevOps es la siguiente:

- Componentes de la Gestión de Desarrollo de Software
- Componentes de la Gestión de Data Center
- Herramientas DevOps como apoyo al Desarrollo y Despliegue de un Proyecto

2.1 Relación de Componentes de la Gestión de Desarrollo de Software y Componentes de la Gestión de Data Center en base a DevOps

DevOps sincroniza la labor del área de desarrollo y operaciones en forma continua, y lo hace desde la etapa de planificación de las tareas a automatizar, hasta el despliegue de aplicaciones al usuario.

Existen dos instancias en las que se desenvuelve la arquitectura DevOps:

- Desarrollo de Aplicaciones
- Despliegue de Aplicaciones

Para definir la relación tiene DevOps en el Desarrollo y Despliegue de aplicaciones, es necesario que cada instancia sea granulada en subcomponentes, con el fin de saber los límites en los roles y principios de influencia.

DevOps requiere para el **Desarrollo de Aplicaciones** metodologías ágiles de desarrollo de software. La guía de metodología ágil se lo analizará en base a Scrum.

En la siguiente tabla se visualiza los roles que intervienen en Scrum:

Tabla 1. Roles de Scrum

Roles	Scrum Máster
	Dueño del Producto
	Equipo de Desarrollo
	Backlog del Producto

DevOps asume los roles de Scrum y para realizar labores colaborativas incluye a los operadores como parte del equipo de desarrollo y se crea el equipo de desarrollo DevOps. Por lo tanto el equipo de Desarrollo DevOps tiene los siguientes roles:

Tabla 2. Roles de DevOps

Roles	Scrum Máster
	Dueño del Producto
	Equipo de Desarrollo
	Backlog del Producto
	Operadores

Scrum tiene definido la siguiente metodología de desarrollo:

Tabla 3. Metodología o ciclo de Desarrollo de Software en base a Scrum

Metodología de Desarrollo de Scrum	Planificación de Iteración
	Inicio Iteración
	Desarrollo Iteración
	Versión Candidata

DevOps asume el ciclo de vida de Scrum, y lo hace por las siguientes pautas:

- Maneja iteraciones cortas. De 2 a 4 semanas por iteración
- Scrum cumple con los principios de cultura de DevOps, desarrollo y operaciones miran en un solo camino, que es la satisfacción del servicio brindado al usuario.
- Realiza entregas parciales del producto a ser automatizado, a través del trabajo colaborativo del equipo de desarrollo. Scrum es dependiente de la automatización end to end [16].

Las pautas de manejo de iteraciones cortas se encuentran directamente relacionadas con las métricas de DevOps.

La Métrica se define como la medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado [18].

Los atributos de las métricas se dan por acuerdo de un protocolo entre directivos y personal operativo, en base a un valor que mide la gestión de un área en un tiempo determinado.

El marco colaborativo en el que se desarrolla DevOps rompe el silo en el que cada una de las áreas siga viendo sus métricas de acuerdo al paradigma de medición de rendimiento acostumbrado.

Gartner ha definido las siguientes métricas para ambiente DevOps:

- Colaboración
- Tiempos de Ciclo
- Frecuencia
- Satisfacción del usuario

Estas métricas tienen relación directa con las pautas de ciclo corto de DevOps.

La **métrica de Colaboración** es parte del concepto de DevOps, desarrollo y operaciones trabajan en forma colaborativa, para entregar un producto en tiempo corto y como único objetivo satisfacer el servicio brindado al usuario.

La **métrica de Tiempo de Ciclo** es un indicador que nos mide el nivel de eficiencia que tiene un desarrollo desde el inicio de la iteración hasta el despliegue de la versión candidata.

La **métrica de Frecuencia** nos indica el tiempo repetitivo en el cual se debe entregar cada iteración.

La **métrica Satisfacción al Usuario**, nos permite medir el grado de satisfacción del usuario en base a la eficiencia de la automatización de las aplicaciones.

DevOps maneja principios que son parte de la estructura colaborativa que son utilizados en toda la etapa de desarrollo y despliegue de aplicaciones. Los principios en los que se desenvuelve DevOps son los siguientes [1][2][17]:

- Integración Continua
- Entrega Continua
- Despliegue Continuo

La **Integración Continua** es la forma en la que el equipo de desarrollo de software integra su trabajo parcial o total, en un determinado tiempo establecido por el equipo de trabajo [17].

La Integración Continua permite estrecha colaboración entre el equipo DevOps que ayude a desarrollar una iteración en un ciclo de tiempo definido en forma satisfactoria.

La Integración Continua para cada iteración con objetivos satisfactorios para el usuario, permitirá frecuencias exactas planificadas en la iteración.

La Integración continua requiere de herramientas de automatización que son únicas para todo el equipo de desarrolladores. Estas herramientas ayudan a integrar en forma

continua partes de código que son validados por pruebas automáticas, lo cual vuelve más eficiente el trabajo del equipo de desarrollo, ya que permite detectar fallos en etapas tempranas del ciclo de desarrollo, lo cual va a favorecer en los tiempos y calidad del producto.

La **Entrega Continua** consiste en desplegar la versión candidata que contiene los requerimientos corporativos automatizados a ser usados por el usuario.

Los tiempos de ciclo definen los periodos de entregas de versiones candidatas.

El **Despliegue Continuo** es la configuración del ambiente de trabajo, que permita un funcionamiento efectivo de las versiones candidatas por parte de los usuarios. Se empieza con una pre-configuración durante todo el proceso de desarrollo y una configuración final antes que se termine la versión candidata.

El Despliegue Continuo visualiza tempranamente los fallos que puedan existir en la aplicación. El momento en que el usuario interactúa más tempranamente con la aplicación, revisa sus requerimientos y se puede volver atrás en el desarrollo.

La segunda instancia en la que se desenvuelve DevOps es el **Despliegue de Aplicaciones**.

El análisis de los componentes del Data Center es crucial para entender la relación que tienen con los componentes del desarrollo de software. Los componentes de software del Data Center, deben ser granulados a un nivel de detalle que permita tener un control preciso de los servicios que están relacionados con las aplicaciones que sirven al usuario.

Desde la etapa de inicio del desarrollo del proyecto, en el que intervienen directamente los desarrolladores, estos se pueden soportar en herramientas que ayuden a dar una verdadera integración continua, para cada una de las iteraciones que se planifiquen para dar solución a los requerimientos a ser automatizados.

2.2 Herramientas DevOps como apoyo al Desarrollo y Despliegue de un Proyecto

El objetivo de la colaboración de DevOps es actualizar y mantener los sistemas de producción estables en todo momento [1]. Las herramientas sirven de apoyo para superar la brecha de desarrollo y operaciones, permiten ser utilizadas en la etapa de desarrollo en forma exclusiva como Dev, y también en la etapa de despliegue como Ops. Tanto para Dev como Ops existen herramientas con software, código, scripts o secuencias de comandos que ayudan a controlar de manera eficiente el ambiente de desarrollo y despliegue de aplicaciones.

Las herramientas DevOps son parte de la estrategia de una buena gestión de administración de recursos. En los ambientes de desarrollo de software las herramientas ayudan a conseguir integración continua, control de versiones de programas, pruebas automáticas, y despliegue continuo. La gestión de operaciones se soporta en herramientas para despliegue automático de aplicaciones, configuración de máquinas virtuales, ejecución automática de scripts.

DevOps se posiciona desde el inicio de la planificación de la iteración como parte de la integración conjunta de conocimientos de desarrolladores y operadores de los objetivos de las etapas del proyecto.

Mientras la aplicación desarrollada se encuentre en producción, el equipo DevOps será el responsable de atender los cambios en las versiones de las aplicaciones.

3. Análisis de Fronteras en los Roles de DevOps

Las áreas de desarrollo y operaciones por las funciones divergentes que tienen establecidas están acostumbradas a trabajar como silos, DevOps ayuda a romper estas divergencias desde las etapas de planificación de la iteración.

En las etapas de desarrollo, pruebas e implementación de una iteración el trabajo colaborativo debe sostenerse con prácticas que ayuden a establecer el trabajo efectivo en la definición de tareas prioritarias del usuario, por lo cual se necesita regular las funciones de las dos áreas.

DevOps no quita ni aumenta funciones, las regula, a través de las siguientes actividades que son anotadas en [1][15]:

- Cultura
- Compartición
- Automatización
- Métricas

El concepto de dar valor a los requerimientos del usuario, es el principal objetivo del componente **Cultura** [1].

La Compartición, es un elemento de DevOps, donde la gente comparte conocimiento, ideas, innovación, procesos y herramientas.

La automatización permite que las tareas manuales, sean realizadas por componentes computacionales de acuerdo a los requerimientos del usuario [1][17].

Las métricas permiten a las áreas de desarrollo de software y operadores, determinar indicadores que le sirve para calificar la eficacia de su labor en base a valores de sus atributos o procesos, pero en forma aislada para cada área.

4. Caso de Estudio

El siguiente caso de estudio fue realizado en la Universidad Central del Ecuador.

Colaboraron en él, 40 estudiantes de pregrado, de la Carrera de Ingeniería Informática.

El objetivo del trabajo es analizar la relación de métricas con principios y fronteras y roles de componentes en el Desarrollo y Despliegue de Software con herramientas DevOps.

El trabajo a realizar es el desarrollo de un sistema que permita el ingreso, actualización y consulta de nota de los estudiantes. El límite de tiempo para desarrollar fue de 100 minutos. Las computadoras donde se realizaron los programas fueron las mismas ya que en los 4 laboratorios existen los mismos tipos de equipos y software para el efecto.

El software para desarrollo de los programas fue Java y Mysql, considerando que los alumnos tenían el conocimiento y experiencia adquiridos en otras asignaturas.

El software para integración continua y control de versiones fue capacitado a 30 estudiantes. Jenkins, Git, GitHub.

Se capacitó a 30 estudiantes en Chef y Puppet.

Se capacitó a 30 estudiantes en Nagios.

Considerando que los alumnos iban a ser distribuidos en diferentes equipos, la composición de los equipos se hizo de forma tal de garantizar que cada equipo estaba capacitado para realizar el desarrollo solicitado con las herramientas requeridas. El uso de controles y herramientas fue discrecional.

4.1 Flujo de tareas y actividades a seguir y resultados obtenidos

- 40 estudiantes formaron 8 equipos (A, B, C, D, E, F, G, H) de 5 desarrolladores cada uno con su líder de equipo.

Tabla 4. Resultados obtenidos con tiempos por cada equipo de trabajo

Flujo de Tareas realizadas por los estudiantes	Equipo							
	A	B	C	D	E	F	G	H
Organización de equipo de trabajo	Si	Si	Si	Si	Si	Si	Si	Si
Estructurar el equipo de trabajo y designar un líder de equipo. Incluir al equipo al dueño del producto	Si	Si	Si	Si	Si	Si	Si	Si
Definición de Procesos manuales y automatizados con los que interactúa el dueño del producto	Si	Si	Si	Si	Si	Si	Si	Si
Planificación y Definición de Tareas a automatizar	Si	Si	Si	Si	Si	Si	Si	Si
Planificación y Definición de Tareas Prioritarias. Definición del ciclo y frecuencia de la iteración	Si	Si	Si	Si	Si	Si	Si	Si
Planificación por equipo de trabajo para que exista Integración, Entrega y Despliegue Continuo de las aplicaciones a realizar	Si	Si	No	No	Si	Si	Si	Si
Identificación de Componentes de la Gestión de Despliegue en Data Center - Servidor identificado, sistemas operativo de servidor, manejador de base de datos, software de frameworks	Si	Si	Si	Si	Si	Si	Si	Si
Identificación de Herramientas DevOps como apoyo al Desarrollo, Java, Mysql	Si	Si	Si	Si	Si	Si	Si	Si
Identificación de Herramientas DevOps como apoyo al Despliegue continuo de las aplicaciones a realizar. Uso de Jenkins y plugin de GIT y repositorio GitHub para Integración Continua y Control de Versiones.	Si	Si	No	No	Si	Si	Si	Si
Uso de Vagrant para despliegue de máquinas virtuales donde van a residir las aplicaciones en servidores.	Si	Si	No	No	Si	Si	Si	Si
Uso de Chef Solo, para configurar y desplegar software Centos, Java, Mysql en el	Si	Si	No	No	Si	No	Si	No

servidor.								
Uso de Puppet, para configurar y desplegar software Centos, Java, Mysql en el servidor.	No	No	No	No	No	Si	No	Si
Instalación de Nagios para monitoreo de servicios, tanto de sistema operativo como de base de datos.	Si	Si	No	No	Si	Si	Si	Si
Uso de monitoreo de Nagios sin haberlo instalado	No	No	Si	Si	No	No	No	No
Análisis de cada equipo de la capacitación en herramientas y procesos a automatizar	Si	Si	Si	Si	Si	Si	Si	Si
Análisis y planificación por equipo que permita compartir en tiempos cortos ideas innovadoras, procesos, herramientas	Si	Si	Si	Si	Si	Si	Si	Si
Identificación de conceptos de métricas DevOps a cumplir	Si	Si	Si	Si	Si	Si	Si	Si
Colaboración. Se analizó si en cada grupo existió relaciones colaborativas que apoyen el avance del trabajo	Si	Si	Si	Si	Si	Si	Si	Si
Tiempo de Ciclo. Total 5 iteraciones	20	20	20	20	20	20	20	20
Frecuencia de Implementación	10	10	10	10	10	10	10	10
	0	0	0	0	0	0	0	0
Frecuencia real (minutos) utilizado en realizar 5 programas, por los equipos que utilizaron herramientas de integración continua	75	75			75	75	75	
Frecuencia real utilizado en realizar 5 programas, por los equipos que no utilizaron herramientas de integración continua			82	133				
Frecuencia real (minutos) utilizado en realizar 5 programas, por equipos que utilizaron herramientas de integración continua por cada programa								55
Fallos luego de una semana de monitoreo de las aplicaciones instaladas	No	No	No	No	No	No	No	No
Satisfacción del Usuario de resultados con herramientas de integración continua	Si	Si	No	No	Si	Si	Si	Si
Satisfacción del Usuario con Frecuencia real (minutos) utilizado en realizar 5 programas por cada equipo, utilizando herramientas de integración continua por cada programa.	No	No	No	No	No	No	No	Si

4.2 Conclusiones del trabajo realizado

El caso de estudio, con el trabajo de cada equipo permite concluir que:

- Los equipo C y D al no cumplir con la aplicación de conceptos de integración continua, demoraron más tiempo en desplegar sus aplicaciones.
- Los equipos A, B, E, F y G, a pesar que utilizaron conceptos y herramientas de integración continua en su desarrollo, no fue suficiente para satisfacer las necesidades del usuario.
- La Frecuencia real (minutos) utilizado en realizar 5 programas por el equipo H, por equipos que utilizaron conceptos y herramientas de integración continua por cada programa, establece que los 55 minutos.

- Se ha cumplido con los objetivos ya que la sincronización de los equipos de trabajo DevOps, que permitió el Análisis de Fronteras en los Roles de DevOps, como cultura, compartición, automatización y métricas y ayudaron a establecer en forma clara el factor determinante que tiene la relación de apoyo colaborativo de DevOps en todas las etapas del desarrollo de software, para alcanzar los objetivos corporativos en tiempos cortos.

5. Conclusiones

Se ha realizado un análisis de los Principios DevOps como Integración Continua, Entrega Continua, Despliegue Continuo, que permite visualizar la cultura como paradigma colaborativo en el desarrollo de software, y que junto a metodologías ágiles producen resultados eficaces en tiempos cortos y ordenados.

Se ha estudiado los Componentes en los que se desenvuelven las Fronteras y Roles DevOps con sus actividades: Cultura, Compartición, Automatización y Métricas, que permiten dar una idea clara de las buenas prácticas que hay que estructurar para llevar adelante en forma eficaz el desarrollo y despliegue de software en forma continua.

Se han definido los límites de DevOps que ayudan a establecer desde donde comienza la labor colaborativa del equipo de trabajo y hasta donde llega la responsabilidad luego del despliegue del todo el proyecto. Esto permite tener claro las responsabilidades de recuperación de fallos de las aplicaciones.

Se ha analizado los alcances del concepto de DevOps, que ayudan a entender los nuevos paradigmas de colaboración entre las áreas de desarrollo y operadores, para todo el proceso de desarrollo de software. De esta manera se logra que cada etapa de la metodología ágil en base a DevOps finalice en forma eficiente en tiempos cortos.

Se realizó una experiencia para analizar la aplicación de DevOps en cuanto a desarrollo y despliegue en alumnos de nivel preuniversitario con resultados interesantes en cuanto a la adopción de nueva forma trabajo, una nueva cultura, para muchos.

Se concluye que la elaboración de una hoja de ruta debe incluir estos pasos:

- Identificación de Componentes DevOps
 - Componentes de la Gestión de Desarrollo de Software con DevOps
 - ✓ Principios de DevOps
 - ✓ Integración, Entrega, Despliegue Continuo
 - Componentes de la Gestión de Despliegue en Data Center
 - Herramientas DevOps como apoyo al Desarrollo y Despliegue de un Proyecto
- Análisis de Fronteras en los Roles de DevOps
 - Cultura
 - Compartición
 - Automatización

Métricas

Referencias

1. Michael Hüttermann, DevOps of Developers, Integrate Development and Operations, The Agile Way, Available: Book of Apress
2. Jez Humble, David Farley, Continuous Delivery, Reliable Software, Releases Through Build, Test, and Deployment Automatic, Foreword y Martin Fowler. Available: <https://buildrelease.googlecode.com/hg-history/1998cd1d530b35b79740d7bf93f8915548136c25/Trunk/BreBooks/Continuous%2520Delivery.pdf>
3. Udo do Pracht, Menschen und IT, DevOps - Gemeinsam produktiv werden, Udo Pracht über Leute, Computer, Prozesse und ihr Zusammenspiel. Available: <https://menschenundit.wordpress.com/2011/05/25/devops/>
4. Benno Luthiger, DevOps bei den ID Build-Automatisierung statt Silo-Betrieb, SWS Entwicklertreffen vom 1.10.2015
5. Robert Schulze, Cezar Adam, Consulting AG, Gegensätze Ziehen Sich An Oder Wie Man Mit Devops Erfolgreich Brücken Bauen Kann. Available: <https://udf.de/files/it-consulting-unternehmensberatung/06-presse/02-veroeffentlichungen/mit-devops-erfolgreich-bruecken-bauen.pdf>
6. OOP Software Meets Business (2012), DevOps, Available: http://www.sigs.de/download/oop_2012/files/Di2-2__Pracht_DevOps.pdf
7. Markus Guske, Michael Kloss, Release a Day, Mythos oder Realität, Available: <http://docplayer.org/1524009-Sonntag-4-oktober-2009-release-a-day-mythos-oder-realitaet.html>.
8. Thomas Rümmler, Christian Schlag, DevOps und Continuous Delivery: sich gemeinsam kontinuierlich verbessern, Online Themenspecial DevOps 2014. Available: http://www.aitgmbh.de/fileadmin/user_upload/pdf/DevOpsUndContinuousDelivery.pdf
9. Peter Gfader, Use Scrum + Continuous Delivery to build the right thing, Scrum.org. Available: https://www.scrum.org/Portals/0/Documents/Community%20Work/Scrum.org%20Whitepaper_Continuous%20Delivery.pdf
10. Christian Rodríguez, Lia Molinari, Francisco Javier Díaz (2013), The Hard Way to Virtual Machine, Administration: towards DevOps, A Bridge between Developers and IT Operators, LINTI (Laboratory of Investigation in New Information Technologies) National University of La Plata UNLP, December 2013.
11. Carlos Lobos Medina, Auditoría de Controles usando Cobit 5 e ISO 27002, Académico Universidad Diego Portales, CISA, CISM, Conferencia Latinoamericana CASC/ISRM 2014.
12. Jez Humble, David Farley, Continuous Delivery, Reliable Software Releases Through Build, Test, and Deployment Automation., Foreword by Martin Fowler, 2011
13. OOP Software Meets Business (2012), DevOps, Available: http://www.sigs.de/download/oop_2012/files/Di2-2__Pracht_DevOps.pdf
14. New Relic, Navigating Devops, What it is and why it matters to you and your business, ebook. 2014
15. Udo Pracht, Meine oder deine Architektur? Von der Entwicklung zum Betrieb mit DevOps, 2011
16. Peter Gfader, Use Scrum + Continuous Delivery to build the right thing, Scrum.org, 2012
17. Michael Hüttermann, Agile ALM, Lightweight tools and Agile strategies, 2011
18. IEEE Software Engineering Standards, Standard 610.12-1990, 1993



VII Workshop Procesamiento de Señales y Sistemas de Tiempo Real (WPSTR)

Comité Científico:

Hugo Ramón (Argentina)
Isidoro Vaquila, (Argentina)
Javier Areta (Argentina)
Javier Orozco (Argentina)
Jorge Ierache (Argentina)
Jorge Lugo (Argentina)
Juan Miguel Santos (Argentina)
Remo Suppi (España)
Roberto Costantini (Argentina)



VII Workshop Procesamiento de Señales y Sistemas de Tiempo Real (WPSTR)

ID	Trabajo	Autores	
9109	Red de Sensores Inalámbricos para el Monitoreo de Adultos Mayores	Darío Weitz (UTN-FRRo), Franco Lianza (UTN-FRRo), Denis María (UTN-FRRo), Nicole Schmidt (UTN-FRRo), Nicole Nant (UTN-FRRo), Sara Feldman (UTN-FRRo)	1054
9130	Supresión de segundo plano en imágenes de tiempo de vuelo	Javier Giacomantone (UNLP), Lucía Violini (UNLP), Luciano Lorenti (UNLP), Armando E. De Giusti (UNLP)	1064
9140	Framework para la captura de eventos mediante interfaz natural de usuario - un caso de aplicación en robótica	Juan Gabriel Fernandez (UM), Gustavo Pereira (UM), Jorge Ierache (UM)	1073
9189	Comparación de Sistemas Operativos Embebidos sobre la Computadora Industrial Abierta Argentina	Santiago Medina (UNLP), Martin Pi Puig (UNLP), Matías Dell Oso (UNLP), Fernando Romero (UNLP), Armando E. De Giusti (UNLP), Fernando G. Tinetti (UNLP)	1083
9190	Prototipo funcional de un sistema de detección de caídas basado en la plataforma CIAA	Matías Dell Oso (UNLP), Laura Lanzarini (UNLP), Pablo Ridolfi (UTN-FRBA)	1093
9276	Arquitectura e Implementación de Equipamiento Prototipo de Bajo Costo para la Medición del Diámetro de Fibras de Cabra en Campo	Rafael Zurita (UNCOMA), Miriam Lechner (UNCOMA), Rodolfo Del Castillo (UNCOMA), Eduardo Aisen (UNCOMA)	1103

Red de Sensores Inalámbricos para el Monitoreo de Adultos Mayores

Darío Weitz¹, Franco Lianza¹, Denis Maria¹, Nicole Schmidt¹, Juan Pablo Nant¹, Sara Feldman²

¹ Facultad Regional Rosario – Universidad Tecnológica Nacional, Zeballos 1341, Rosario

² Facultad de Ciencias Médicas, Universidad Nacional de Rosario

dar.wtz@gmail.com

Resumen. La población mundial se encuentra en un proceso de envejecimiento debido al efecto combinado de una disminución en las tasas de fertilidad y el incremento en las expectativas de vida. Las Tecnologías de Monitoreo y Asistencia Remota permiten monitorear las Actividades de la Vida Diaria de las personas para prevenir el agravamiento de enfermedades crónicas cuya prevalencia se incrementa con la edad. Se propone una red de sensores inalámbricos para el monitoreo y asistencia remota de un adulto mayor que vive solo en su residencia habitual. Se seleccionaron tecnologías simples, económicas, de bajo consumo energético y de fácil instalación y mantenimiento. La evaluación del sistema propuesto se realiza mediante un modelo de simulación dada la dificultad de realizar experimentos intrusivos y de larga duración en domicilios de adultos mayores. El sistema genera resúmenes gráficos y presenta la información resumida a familiares y proveedores de salud mediante aplicaciones web.

Palabras clave: monitoreo y asistencia remota, sensores, modelo de simulación.

1 Introducción

La División Población del Department of Economic and Social Affairs de las Naciones Unidas indicó que la población mundial llegó en el año 2013 a los $7.2 \cdot 10^9$ habitantes en el planeta, con un crecimiento promedio anual de 81 millones de personas. Se estima que llegará a $8.1 \cdot 10^9$ para 2025 y a $9.6 \cdot 10^9$ a finales del año 2050. Analizando las franjas etarias, se pronostica que la población mayor a los 60 años será la de mayor crecimiento: la tasa de crecimiento en los países desarrollados será del 1% anual hasta el año 2050 y luego se reducirá al 0.11% anual para el período entre 2050 y 2100. Este grupo etario crecerá en tales países de $287 \cdot 10^6$ en 2013 a $417 \cdot 10^6$ para el año 2050. Por su parte, la tasa de crecimiento de los mayores de 60 años es muy superior en las regiones con menor desarrollo económico-social: 3.7% en el período 2010 – 2015; 2.9% para el período 2016 – 2050; 0.9% para 2050 – 2100. En número de personas implica pasar de $554 \cdot 10^6$ en 2013 a $1.6 \cdot 10^9$ en el año 2050 y $2.5 \cdot 10^9$ a finales de la centuria [1].

Estos números muestran con claridad que la población mundial está envejeciendo. Es el resultado de dos factores concurrentes: i.- disminución en la tasa de fertilidad; ii.- continuos avances en ciencias y tecnologías de la salud junto a mejoras en las condiciones económicas, sociales y ambientales que se traducen en un significativo aumento en las expectativas de vida en todas las regiones del planeta.

El envejecimiento inevitablemente resulta en la disminución de la agudeza sensorial (vista, audición, olfato, etc.), menor resistencia y fuerza muscular, inconvenientes en la movilidad, disminución de la claridad mental y alteraciones en la estabilidad. Más importante aún, con la edad se incrementa de manera significativa la prevalencia de enfermedades crónicas tales como hipertensión arterial, asma, enfermedades cardiovasculares, diabetes, enfermedad pulmonar obstructiva crónica (EPOC) y otras que son la principal causa de muerte en todos los países [2]. Un resultado inevitable de lo recientemente indicado es que los costos de los servicios públicos de salud crecen de manera muy sesgada porque los mayores de 65 años consumen de 3 a 5 veces más recursos cuando se los compara con los menores de esa edad [3].

La dramática aceleración en el número de pacientes crónicos junto al incremento en los costos de los servicios públicos de salud y la probable escasez de trabajadores de cuidado directo (enfermeras, paramédicos, ayudantes de la salud, auxiliares de atención domiciliaria) muestran la necesidad de un cambio radical en el proceso de cuidado de los adultos mayores. La reducción en los costos de los servicios de salud solo podrá ser provista por soluciones de gestión de las enfermedades que expandan el flujo de información hacia los centros de salud por parte de pacientes de alto riesgo o pacientes que sufren de enfermedades crónicas.

El modelo tradicional de provisión de servicios de salud “reactivo” debe ser modificado por un modelo proactivo que provea recursos en todo momento y en todo lugar. El modelo proactivo está basado en dos conceptos claves: i.- Sistemas de Salud Personalizados (SSP); ii.- Sistemas de Monitoreo y Asistencia Remota [4]. Los Sistemas de Monitoreo y Asistencia Remota tienen entre sus objetivos el transformar el hábitat del paciente para recolectar y compartir información de sus residentes con los mismos, sus familiares y con proveedores de servicios de salud asociados. Las Tecnologías de Monitoreo y Asistencia Remota permiten monitorear las actividades diarias de las personas y detectar cambios funcionales para actuar como sistemas de alerta temprana, y de esta manera prevenir la ocurrencia de un evento severo de salud.

La implementación y puesta a punto de Sistemas de Monitoreo y Asistencia Remota es un proceso de alta complejidad que intentó ser resuelto mediante la construcción de hogares inteligentes. Los proyectos más elaborados fueron: i) CASAS (Center for Advanced Studies in Adaptive Systems) de la Washington State University (WSU) [5], un hogar inteligente establecido en el campus Pullman de la WSU donde se prueban diversas tecnologías que monitorean las actividades de los residentes y mediante técnicas de inteligencia artificial se busca mejorar el confort, la seguridad y la productividad de los habitantes del hogar; ii) Proyecto PlaceLab del MIT [6], una estructura de investigación científica altamente instrumentada y con las dimensiones de una vivienda real donde se recolectaron datos ambientales y de comportamiento humano. Se monitoreó a un grupo de voluntarios y se evaluaron diferentes estrategias y tecnologías de monitoreo. El proyecto concluyó en el año 2008; iii) Gator Tech Smart House de la Universidad de Florida [7], una casa de amplias dimensiones transformada en un laboratorio experimental donde se implementan, prueban y validan diversos sistemas orientados a asistir adultos mayores o personas con necesidades especiales con el objetivo de que los mismos vivan de manera independiente y con alta calidad de vida. Las tareas de investigación son desarrolladas por personal del Mobile and Pervasive Computing Laboratory de la citada universidad.

Numerosas empresas privadas están lanzando al mercado sensores, sistemas de monitoreo, sistemas de asistencia remota y otras soluciones de telemedicina, de cuya eficacia y utilidad existe poca información confiable [8], [9]. A pesar de las numerosas innovaciones tecnológicas que ya se encuentran en el mercado, el desarrollo de un sistema eficiente de provisión remota de salud es una tarea compleja y masiva debido a la necesidad de integrar tecnologías heterogéneas en una estructura de servicio muy imbricada. La industria carece de

una apropiada integración de las tecnologías de sensado, recopilación de datos, transmisión remota y algoritmos y métodos computacionales que produzcan los resultados deseados [10].

La implementación y evaluación de un sistema real de monitoreo y asistencia de adultos mayores presenta varias dificultades: i) dificultad para reclutar participantes ii) necesidad de recolectar datos durante un período muy prolongado de tiempo; iii) recolección de datos de disímil nivel de confiabilidad; iv) dificultad en detectar un número significativo de anomalías; v) dificultad para evaluar escenarios que impliquen riesgos para los residentes [11]. Existe conjuntos de datos públicos que podrían ser utilizados para el entrenamiento y evaluación de sistemas de monitoreo [12]. El inconveniente para su aplicación radica en que los voluntarios reclutados no siempre pertenecieron a la franja etaria de adultos mayores, los datos son muy específicos de la cultura del país de registro y escasamente confiables porque los sujetos bajo estudio suelen comportarse de manera diferente al sentirse observados.

Las dificultades previamente detalladas indican la conveniencia de desarrollar una etapa previa de simulación antes de incorporar un sistema de monitoreo en la residencia de un adulto mayor. Un modelo de simulación es una herramienta muy eficaz para ganar comprensión y conocimiento sobre la conducta de un sistema complejo. Permite experimentar y poner a punto diversas configuraciones, plantear escenarios de riesgos y anomalías, responder preguntas y evaluar ideas, algoritmos y métodos. Como resultado, se podrá obtener un diseño más robusto, confiable y económico. La literatura reporta un cierto número de simuladores o modelos de simulación que se emplean para evaluar redes de monitoreo, pero la elevada complejidad del problema no ha permitido obtener aún una solución definitiva [13], [14].

En este trabajo se propone una red de sensores inalámbricos para el monitoreo y asistencia remota de un adulto mayor que vive solo en su residencia habitual. Se seleccionaron tecnologías simples, económicas, de bajo consumo energético y de fácil instalación y mantenimiento. La evaluación del sistema propuesto se realiza mediante un modelo de simulación dada la natural dificultad de realizar experimentos intrusivos y de larga duración en domicilios de adultos mayores con enfermedades crónicas. El sistema permite generar resúmenes gráficos a partir de enormes cantidades de datos de sensores y presentar la información resumida a familiares y proveedores de salud mediante aplicaciones web.

2 Modelo de Simulación

El modelo de simulación fue desarrollado utilizando el lenguaje de programación C#. Es un lenguaje de programación de propósito general orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET. El framework de desarrollo utilizado fue Microsoft Visual Studio 2012. El modelado de la vista 3D se realizó mediante el software SketchUp Pro 2015, utilizando componentes de la galería 3D Warehouse. El renderizado del modelo se realizó con el software SU Podium V2.5 Plus. Las líneas de redes de sensores se dibujaron mediante Visual Basic Power Packs Line and Shape controls. El motor de base de datos para el almacenamiento de datos y resultados fue Microsoft SQL Server 2008.

El algoritmo de simulación se desarrolló según el esquema de modelado de simulación de eventos discretos que utiliza el enfoque exploración de actividades [15]. Es una clase de modelo de simulación dinámico, discreto y estocástico. La dimensión dinámica involucra el pasaje del tiempo simulado y una representación explícita de la secuencia de actividades que realiza el residente del hábitat simulado. La dimensión discreta indica que los eventos de interés cambian en puntos separados en el tiempo. La dimensión estocástica permite modelizar la conducta relativamente impredecible que tienen los seres humanos en sus actividades

diarias. El enfoque exploración de actividades enfatiza una revisión de todas las actividades presentes en la simulación para determinar cuál puede iniciarse o finalizar al siguiente avance del reloj de la simulación. Es computacionalmente menos eficiente que los otros enfoques y por ello se lo utiliza menos, pero es el apropiado para la simulación de las actividades diarias de un adulto mayor. Se define a la actividad con un par de eventos: uno que inicia y otro que completa una operación que transforma el estado de una entidad. Las actividades tienen duraciones finitas y los cambios de estado se observan al comienzo o al final de la misma.

El modelo de simulación consta de tres componentes principales: i) el habitante de la residencia cuyas actividades diarias son monitoreadas; ii) la residencia simulada; iii) redes de sensores inalámbricos. El modelo está restringido a la simulación de una sola persona; no se considera una restricción importante debido a que la coexistencia de dos o más personas en una residencia permite una mutua evaluación del estado de salud superior al de cualquier sistema de información. La residencia simulada consta de un dormitorio, un baño, un estudio con computadora, un living con televisor y una cocina equipada que incluye una mesa y silla. Permite una vista en planta (2D) o una vista 3D. Las redes de sensores inalámbricos se encargan de registrar el conjunto de Actividades de la Vida Diaria (AVD) que desarrolla el habitante virtual de la residencia simulada. Al respecto, se consideraron dos tipos de actividades diarias: i) actividades de tiempo específico (desayunar, almorzar, dormir a la noche, etc.), caracterizadas por realizarse solo una vez por día y en horario acotado; ii) actividades que se repiten (ir al baño, tomar mate, ver televisión, etc.) que suelen realizarse más de una vez por día y en horario libre. Se seleccionaron aquellas AVD que podían ser registradas mediante sensores inalámbricos simples y económicos, y cuyos cambios significativos podrían ser un indicador de deterioro en las condiciones de salud del residente y así servir como alerta temprana.

Un módulo de gestión de sensores permite la incorporación de diversos tipos, marcas, modelos y proveedores de sensores de movimiento, temperatura, humedad, ocupación de cama y/o sillón. También sensores magnéticos de apertura de puertas, de sonido, fotocontrol y detectores de humo, gas natural y monóxido de carbono. Haciendo clic derecho sobre cuadrados azules dibujados en los distintos ambientes de la residencia simulada se pueden agregar o quitar sensores, activarlos o desactivarlos. El sistema permite configurar la probabilidad de ocurrencia de un siniestro (incendio, pérdida de gas, etc.), distribuyendo esa probabilidad entre las diferentes habitaciones de la residencia.

La simulación procede según un mecanismo de cuatro etapas: i) generación de la secuencia de actividades; ii) avance del tiempo simulado al tiempo de ocurrencia de la próxima actividad; iii) activación de los correspondientes sensores; iv) almacenamiento en base de datos del tiempo simulado, actividad y sensores activados. Una de las suposiciones fundamentales del modelo propuesto es que la secuencia de actividades se genera a partir de distribuciones de probabilidad específicas para cada actividad de tiempo específico y cada actividad repetitiva. Tales distribuciones de probabilidad se obtienen analizando datos empíricos y ajustando esos datos mediante software apropiado. Si el tiempo de ocurrencia de una o más actividades se solapan con otra actividad en proceso, el algoritmo coloca a las primeras en una cola de espera y procede a ejecutarlas según un esquema FIFO. La secuencia de números aleatorios imprescindibles para generar las variables aleatorias correspondientes a las distribuciones de probabilidad se obtiene a partir del generador provisto por el lenguaje de programación utilizando como semilla el reloj del microprocesador o un valor ingresado desde teclado. Esta última alternativa permite repetir la secuencia de números aleatorios y se la utiliza durante la etapa de verificación para asegurar que el modelo de simulación esté libre de errores lógicos.









Se incluyó un esquema de animación para mejorar la representación de la secuencia de actividades que realiza el residente del hábitat simulado. La animación colabora en el proceso de validación del modelo de simulación –asegurar que el modelo sea una adecuada representación de la realidad- al permitir que los familiares y/o profesionales de la salud involucrados visualicen la secuencia de actividades. Adicionalmente, el sistema incluye la posibilidad de realizar la simulación sin el proceso de animación cuando el objetivo es la prueba del sistema o la obtención rápida de resultados.

3 Descripción del Sistema

Un hogar inteligente es una residencia a la cual se le incorporan sensores para monitorear el ambiente y dispositivos para proveer servicios proactivos que mejoran la calidad de vida y seguridad de los residentes. Tres tipos diferentes de tecnologías de sensores se han utilizado para el monitoreo de personas en hogares inteligentes: (a) sensores corporales (wearables); b) sensores inalámbricos embebidos en el ambiente; c) sensores mediados por la infraestructura [16]. La alternativa (a) es excesivamente intrusiva, propensa a fallas y con limitaciones para la determinación de actividades. La alternativa (c) es compleja y costosa cuando los sensores se deben incorporar en una residencia ya construida (en cañerías, sistemas eléctricos, ductos, etc.). La alternativa (b) corresponde a un conjunto de dispositivos autónomos, distribuidos físicamente en la residencia a monitorear y que se comunican mediante algún tipo de tecnología inalámbrica. Su topología es flexible y no requieren de un número mínimo para su funcionamiento.

Para el desarrollo del proyecto se eligió una red inalámbrica de sensores que pueden ser embebidos en el ambiente y que satisfacen los siguientes requerimientos: i) no invasivos, discretos y relativamente “invisibles” de forma tal que no interfieren con las actividades diarias que realiza el residente; ii) mínimo consumo de energía; iii) comunicación fluida a través de un protocolo simple; iv) económicos y simples de instalar; v) mínimos requerimientos de mantenimiento; vi) garantizan la privacidad y confidencialidad de los datos. La Tabla 1 muestra imágenes y describe las características principales de los sensores seleccionados.

Tabla 1. Descripción de los sensores.

Modelo	Variable Medida	Imagen	Especificaciones	Modelo	Variable Medida	Imagen	Especificaciones
BH1750	Intensidad de Luz		*V. Entrada: DC 3.3V *Rango Medición: 1 a 65535lx	FC-04	Sonido		*V. Entrada: DC 3.3V – 6V *Sensibilidad: Ajustable
DHT11	Temperatura y Humedad		*V. Entrada: DC 3.3V – 6V *Consumo: 1.5mA *Rango Medición: 20% – 90% y 0 – 50°C *Tiempo salida: <5 seg *Precisión: ±5.0% y ±2.0°C	MPX201A	Presión		*V. Entrada: DC 3.3V – 6V *Consumo: 6mA *Rango Medición: 0 – 200 kPa
HC-SR501	Movimiento		*V. Entrada: DC 4.5V – 20V *Consumo: <60 uA *Rango Medición: 3 – 7m *Tiempo salida: <5 seg *Temp. de Trabajo: -20 – 80°C	MQ2	Gas, Humo, CO		*V. Entrada: DC 5V *Consumo: 122mA *Rango Medición: 10 – 1000 ppm
LDR	Intensidad de Luz (Binario)		*V. Entrada: DC 3.3V – 6V *Consumo: 10mA *Sensibilidad: Ajustable	ARD0086	Contacto (Magnético)		*V. Entrada: DC 3.33 – 5V *Consumo: 15mA

La Figura 1 muestra la arquitectura del sistema que corresponde a una topología de tres capas: el menor nivel –capa de percepción– está formado por distintos tipos de sensores según la actividad de monitoreo requerida. Los sensores se encuentran conectados mediante una

topología malla y se comunican mediante el protocolo ZigBee (IEEE 802.15.4). Es una tecnología de comunicación inalámbrica que en su variante estándar utiliza la banda de 2.4 GHz, y se caracteriza por su simplicidad, bajo consumo energético, bajo costo, alta capacidad de red y transferencia de datos confiable. Presenta baja latencia y se la utiliza en aplicaciones con baja tasa de transmisión y de corto alcance. En la capa 1 de la Figura 1 se muestra la ubicación de los tres dispositivos que conforman la tecnología: coordinador ZigBee; routers ZigBee; end devices. ZigBee es una tecnología comercialmente madura que puede mantenerse operativa desde seis meses hasta varios años y se considera que puede transformarse en el estándar global de las redes ubicuas a partir de los cientos de fabricantes que la desarrollan a escala mundial [17]. La capa 2 se denomina capa de procesamiento y está conformada por un servidor y un router. El servidor recibe los datos provistos por los sensores y los almacena en una base de datos donde serán procesados mediante modelos entrenados para detectar cambios en la frecuencia o duración de las AVD o para la detección de patrones anómalos. La información se encripta previa a su transmisión vía router a un servidor de internet. La tercera capa –capa de aplicación– resume la información procesada. Podrá ser accedida por el propio usuario, sus familiares y un servicio de salud remoto. Habrá un repositorio de datos donde se guardará la información para su evaluación por parte de profesional médico calificado. Se incluyen procedimientos de seguridad para evitar que la información llegue a conocimiento de personal no autorizado.

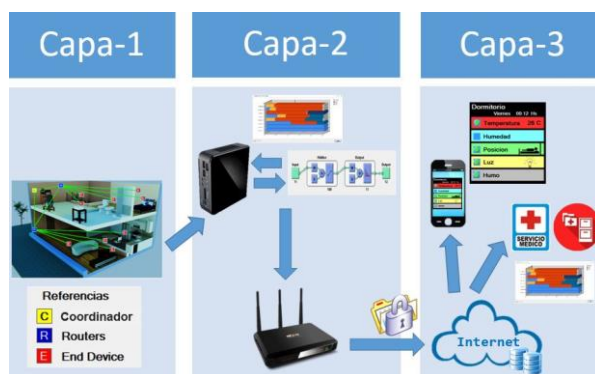


Fig. 1. Arquitectura del sistema.

4 Recolección de datos

El perfil genérico del residente simulado corresponde a una persona mayor que vive sola en su domicilio, posee autonomía de movimientos, sufre de alguna enfermedad crónica pero prefiere vivir de manera privada e independiente. Para poder generar la secuencia de actividades se solicitó a una mujer de 81 años que vive sola en su domicilio particular el llenado de un diario de actividades de la vida diaria (repetitivas y de tiempo específico) durante seis meses. Algunos datos ausentes por omisión u olvido fueron completados mediante la función Datos Aleatorios del Menú Calc del software Minitab. Los datos fueron ingresados a una planilla Excel y posteriormente procesados mediante el software Crystal Ball [15] para obtener las funciones de distribución de probabilidad (con sus correspondientes parámetros) de la hora de inicio y duración de las actividades de tiempo específico que mejor ajustaron a los datos recolectados. Las frecuencias de las actividades repetitivas se modelizaron mediante distribuciones empíricas.

La Figura 2 muestra el mejor ajuste determinado por el software Crystal Ball a los datos recolectados para la propiedad Duración de la actividad Merienda. Para los otras actividades de tiempo específico se utilizaron las siguientes distribuciones de probabilidad (entre paréntesis se indica el valor de la prueba Anderson – Darling): Dormir – Hora de Inicio: Weibull (0,5227); Desayunar: Hora de Inicio, Beta (0,1635); Duración, Binomial Negativa; Almorzar: Hora de Inicio, Logística (0,3434); Duración, Binomial Negativa; Merendar: Hora de Inicio, Normal (0,2004), Duración, Binomial; Cenar: Hora de Inicio, Logarítmico Normal (0,1940), Duración, Poisson.

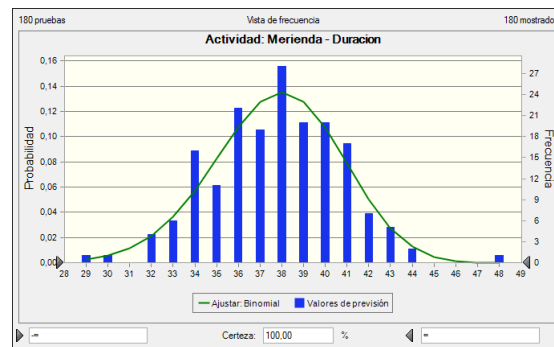


Fig. 2. Ajuste de la actividad Merienda – Duración.

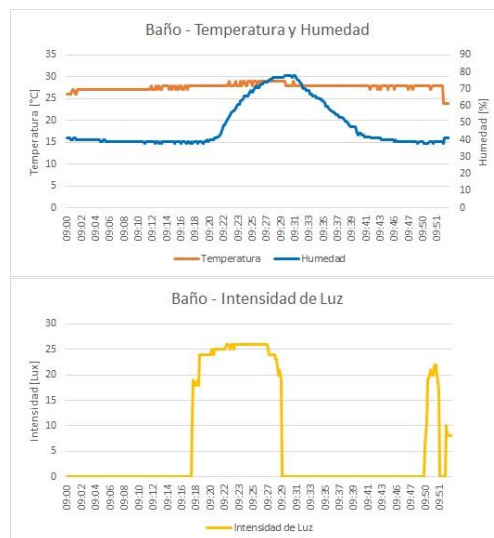


Fig. 3. Datos recolectados en el ambiente baño.

Las redes de sensores inalámbricos fueron ensayadas mediante su instalación en residencias de familiares de autores del proyecto. El objetivo fue evaluar el funcionamiento de las mismos y su utilización para la detección de eventos y actividades domésticas: sensores infrarrojos de movimientos para detectar los desplazamientos; sensores magnéticos de puertas para detectar salida hacia el exterior; sensores de presión y luminosidad para el control de horas de sueño; sensores de presión para el control del uso de computadora y televisor. La Figura 3 muestra como el uso combinado de sensores de temperatura, humedad y luminosidad instalados en un baño permite detectar la realización de la ADV tomar una ducha.

Se utilizaron datos de temperatura y humedad registrados mediante termo-higrómetros digitales y almacenados en data loggers Hobo (Hobo U23-003). Las mediciones se realizaron en el microcentro de la ciudad de Rosario durante el período setiembre de 2013 hasta agosto de 2014 en intervalos de 15 minutos [18].

5 Discusión

La Figura 4 muestra la pantalla principal del modelo de simulación. Se observan las diferentes habitaciones que conforman la residencia, como así también al habitante virtual desarrollando una de las actividades diarias repetitivas (mirar televisión). Los círculos verdes sobre un recuadro azul indican la presencia de uno o más sensores activos en ese ambiente. Un círculo verde intenso indica que el sensor está recolectando el dato correspondiente, mientras que los cuadrados azules corresponden a posibles ubicaciones de sensores que en la corrida de simulación actual no están agregados.

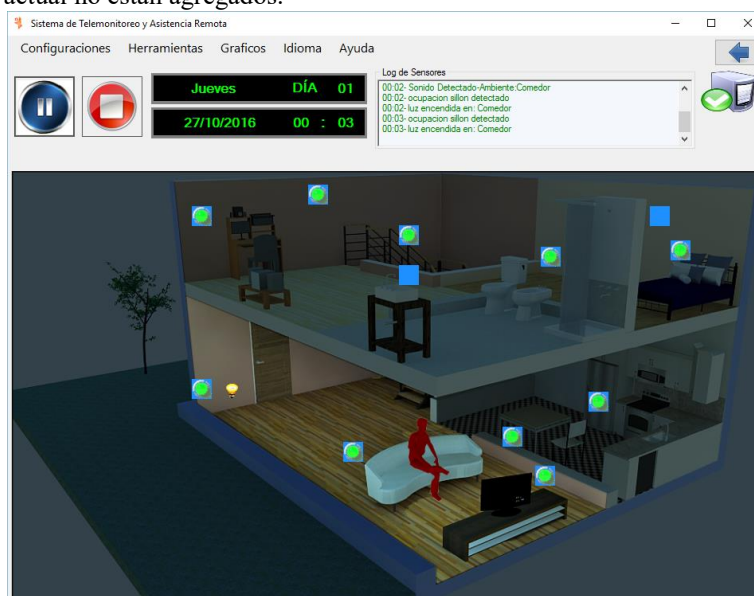


Fig. 4. Vista principal del modelo de simulación.

Antes de iniciar una corrida de simulación, se procede a activar el conjunto de sensores que caracterizan el escenario a evaluar. También se configura la probabilidad de ocurrencia de un siniestro (incendio, pérdida de gas, etc.), y la distribución de esa probabilidad entre las diferentes habitaciones de la residencia. Se define la fecha de inicio de la simulación, la cantidad de días a simular y se elige entre una simulación con animación (para visualización del escenario o validación del modelo) o una simulación sin animación para obtención rápida de resultados. Se evaluaron diferentes escenarios relacionados con la cantidad y despliegue del conjunto de sensores inalámbricos que registraban las AVD del residente. La información recolectada durante las corridas se visualiza en la pantalla en un log de sensores y se almacena en una base de datos. A partir de la misma se obtienen gráficas de producción y reportes de métricas con la intención de detectar cambios en la frecuencia o duración de las AVD y así poder actuar como sistema de alerta temprana para prevenir la ocurrencia de un evento severo de salud.

La capa de aplicación contiene un conjunto de aplicaciones gráficas que les permite a los familiares y proveedores de salud remotos visualizar información puntual e información resumida referida a los AVD del residente monitoreado.

La Figura 5 muestra imágenes que pueden observarse en un smartphone relacionadas con los datos recolectados por los sensores en el dormitorio: temperatura; humedad relativa del ambiente (no agregado); ocupación de la cama; luces encendidas; detección de humo (inactivo). La Figura 6 muestra una gráfica de valores promedios de la ocupación de los diferentes ambientes en periodos de tres horas; se observa una importante ocupación del dormitorio como resultado del hábito de la moradora de leer acostada en la cama. La Figura 7 muestra en un diagrama estilo Gantt la distribución diaria de ambos tipos de AVD. Este tipo de salida gráfica puede ser utilizado como una herramienta para detectar cambios en los hábitos del morador de la residencia.

El sistema permite generar conjuntos de datos sintéticos que en la siguiente etapa del proyecto serán utilizados para entrenar y evaluar algoritmos de reconocimiento de actividades y/o detección de valores anómalos.



Fig. 5. Aplicación móvil del sistema.

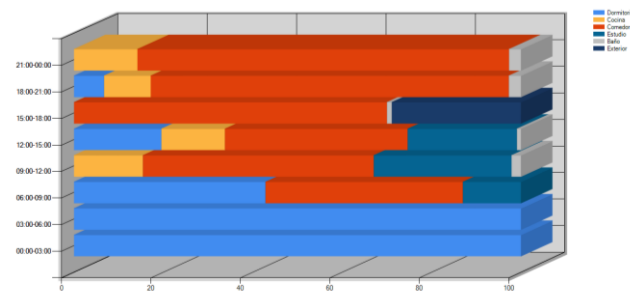


Fig. 6. Grafica de ocupación de ambientes.

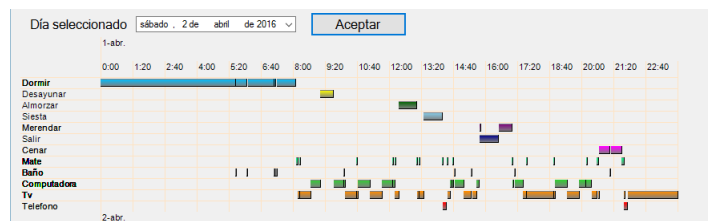


Fig. 7. Resumen diario de actividades.

6 Conclusiones

La población mundial se encuentra en un proceso de envejecimiento debido al efecto combinado de una disminución en las tasas de fertilidad y un notable incremento en las expectativas de vida. Un número creciente de adultos mayores, por preferencia y necesidad, vivirá solo en su residencia habitual a pesar de los riesgos y peligros inherentes a ese modo de vida.

Los hogares inteligentes proveen una alternativa factible, segura, relativamente económica y confiable para la protección y asistencia de personas vulnerables debido a enfermedades

crónicas. El éxito de un hogar inteligente reside en la calidad de información que reciba el sistema de soporte de toma de decisiones, el cual a su vez depende de la cantidad, calidad y despliegue de los sensores involucrados y de la arquitectura general del sistema.

Se ha diseñado una primera propuesta de un sistema de monitoreo y asistencia remota de un adulto mayor con enfermedades crónicas que es configurable, flexible y escalable. Se utilizan tecnologías simples, económicas, de bajo consumo energético y de fácil instalación y mantenimiento. El sistema se evalúa mediante un modelo de simulación de eventos discretos que utiliza el enfoque exploración de actividades. Un conjunto de aplicaciones gráficas permite el monitoreo remoto por parte de familiares y proveedores de salud asociados. El objetivo del sistema es mejorar la calidad de vida de los residentes tratando de predecir el agravamiento o la exacerbación de las condiciones crónicas de los mismos.

Referencias

1. United Nations, Department of Economic and Social Affairs, Population Division: World Population Prospects: The 2012 Revision, Highlights and Advance Tables. Working Paper No. ESA/P/WP.228. (2013)
2. Paré, G, Jaana, M., Sicotte, C.: Systematic Review of Home Telemonitoring for Chronic Diseases: The Evidence Base. *Journal of the American Medical Informatics Association*, 14, 3, May/June 2007.
3. Holland, T.: Healthy at Home: The Economic Benefits of Remote Patient Monitoring, <https://insights.samsung.com/2016/02/05/healthy-at-home-the-economic-benefits-of-remote-patient-monitoring/>
4. European Commission, Joint Research Centre, Institute for Prospective Technological Studies: Strategic Intelligence Monitor on Personal Health Systems (SIMPHS). Market Structure and Innovation Dynamics, EUR 24669 EN.
5. CASAS, Center for Advanced Studies in Adaptive Systems, <http://casas.wsu.edu/research-projects>
6. MIT, Media Lab Project April 2015, <http://www.media.mit.edu/files/projects.pdf>
7. Mobile and Pervasive Computing Research, University of Florida, <http://www.icta.ufl.edu/gt.htm>
8. Phillips Enterprise Telehealth, <http://www.usa.philips.com/healthcare/solutions/enterprise-telehealth>
9. Samsung, Business Home, Telecare, <http://www.samsung.com/us/business/by-industry/healthcare/>
10. Suryawanshi, C.R., Bhute, Y.C.: A WSN based system for enhancing intra mobility solutions for healthcare – A review, *International Journal of Computer Sciences and Engineering*, 2 (9), pp. 33 – 37, (2014)
11. Cardinaux, F., Brownsell, S., Bradley, D. Hawley, M.S.: A home daily activity simulation model for the evaluation of lifestyle monitoring systems, *Computers in Biology and Medicine*, 43, pp. 1428 – 1436, (2013)
12. UCI MLR, <http://archive.ics.uci.edu/ml/datasets.html?sort=nameUp&view=list>
13. Kormanyos, B., Pataki, B.: Multilevel Simulation of Daily Activities: Why and How, 2013 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 15-17 July (2013)
14. Elfaham, A., Hagra, H., Helal, S., Hossain, S., Woong Lee, J., Cook, D.: A Fuzzy Based Verification Agent for the Persim Human Activity Simulator in Ambient Intelligent Environments, 2010 IEEE International Conference on Fuzzy Systems (FUZZ), 18-23 July (2010)
15. Evans, J.R., Olson, D.L.: Introduction to Simulation and Risk Analysis, Prentice Hall, (1998)
16. Ding, D., Cooper, R.A., Pasquina, P.F., Fici-Pasquina, L.: Sensor technology for smart homes, *Maturitas*, 69, pp. 131-136, (2011)
17. Ozkul, T., Sevin, A.: Survey of Popular Networks Used for Biosensors, *Biosensors Journal*, 3, 1, (2014)
18. Coronel A.; Feldman, S.R.; Jozami, E.; Kehoe, F.; Piacentini, R. D. Dubbeling, M.; Escobedo, F. J.: Effect of urban green areas on air temperature in a medium size Argentine city. *Environment Science* 2(803-826), (2015)

Supresión de segundo plano en imágenes de tiempo de vuelo

Javier Giacomantone, Lucía Violini, Luciano Lorenti, Armando De Giusti

Instituto de Investigación en Informática (III-LIDI),
Facultad de Informática - Universidad Nacional de La Plata - Argentina.
La Plata, Buenos Aires, Argentina.
{jog, lviolini, llorenti, degiusti}@lidi.info.unlp.edu.ar

Resumen. En este artículo se presenta un método computacional para detectar y extraer el plano de fondo, segundo plano, a partir de datos obtenidos por cámaras de tiempo de vuelo. Se utiliza una variante de un método de clasificación basado en máquinas de soporte vectorial. Considerando las características particulares del tipo de cámaras utilizadas, se incorpora adecuadamente la información de rango e intensidad, y se utiliza la capacidad para obtener secuencias rápidas de datos en una modalidad particular. El artículo revisa las técnicas específicas de reconocimiento de patrones utilizadas, presenta la solución propuesta y muestra resultados experimentales preliminares del método propuesto.

Palabras clave: Visión por computador, cámaras TOF industriales, Reconocimiento de patrones, Máquinas de soporte vectorial.

1. Introducción

Los métodos de sustracción de plano de fondo, *background*, son utilizados en visión por computador normalmente para detectar objetos de interés, *foreground objects*, en una secuencia de imágenes [1]. La representación del plano de fondo se denomina modelo. En los últimos años han sido propuestos distintos métodos [2][3][4]. Estos varían en cuanto a sus beneficios y limitaciones dependiendo del tipo de imágenes, el ruido, los artefactos y las hipótesis básicas sobre las que el correspondiente método computacional se sustenta. Los principales métodos se pueden clasificar en paramétricos o no paramétricos según establezcan o no suposiciones sobre la distribución de probabilidades de la intensidad del plano de fondo [5]. Otro criterio de clasificación posible es entre los métodos basados en analizar cada *pixel* (*picture element*), o los basados en bloques que dividen la imagen en un número de bloques de pixels sobre los cuales operan. Un método normalmente parte del conocimiento previo de imágenes del plano de fondo, pero no de los objetos de interés. Por lo tanto, desde el punto de vista de reconocimiento de patrones, es un problema de clasificación semisupervisado [6]. El plano de fondo puede ser estacionario o no estacionario. Las cámaras de tiempo de vuelo, *time of flight* (*TOF*), utilizan un arreglo de sensores de tecnología

CMOS y su propia fuente de iluminación en el infrarrojo cercano [7][8]. A partir de la diferencia de fase entre la señal emitida y la reflejada por la escena para cada posición obtienen imágenes de rango, $2\frac{1}{2}D$, e intensidad $2D$. A pesar de ser cámaras de baja resolución, permiten obtener secuencias rápidas de imágenes [9]. La sensibilidad a la iluminación de fondo requiere no sólo consideraciones de diseño particulares sino también la adecuada calibración del tiempo de integración [10][11]. Las características generales anteriores requieren consideraciones y métodos específicos para su posprocesamiento en general, y en particular para la sustracción o también denominada supresión del *background*. La figura 1 muestra cuatro modelos distintos de cámara TOF.

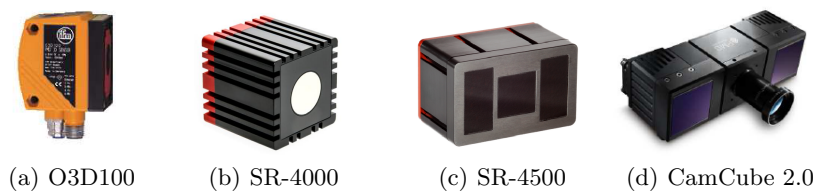


Figura 1. (a) Ifm electronic[©], (b-c) HEPTAGONTM, (d) PMD[Vision][®]

Glazer et. al [12] proponen un método orientado a bloques semisupervisado, utilizando máquinas de soporte vectorial, para sustraer el plano de fondo en imágenes de intensidad pero no en imágenes de rango. Cho et. al [13] proponen, utilizando imágenes de tiempo de vuelo, generar un modelo estadístico del fondo tanto para la imagen de intensidad como para la de rango. Una vez obtenido el modelo, cada pixel de una nueva imagen es comparado con el modelo y se lo clasifica como perteneciente o no al fondo. El método propuesto en este artículo aborda el mismo tipo de problema, es orientado a pixels y asume clases desbalanceadas. Considera los datos obtenidos en una estructura particular y utiliza clasificación basada en máquinas de soporte vectorial de una clase, *one class support vector machines (OCSVM)*. El objetivo de la alternativa propuesta es sustraer el plano de fondo combinando la información de rango e intensidad que permita determinar los objetos de la escena que no pertenecen al *background*.

El artículo ha sido organizado del siguiente modo. En la sección 2 se presenta una breve revisión de los conceptos fundamentales utilizados para construir el modelo. La sección 3 presenta el método propuesto. La sección 4 expone resultados experimentales. Finalmente, en la sección 5 se presentan conclusiones.

2. Máquinas de soporte vectorial

Las máquinas de soporte vectorial tienen como fundamento la teoría de aprendizaje estadístico [14]. En esta sección se presenta en forma breve el método de clasificación y la notación utilizada. Dados l patrones de entrenamiento para dos

clases $\{(x_i, y_i), i = 1, \dots, l\}$, donde $x_i \in \mathbb{R}^N$ es un patrón, un vector de características de dimensión N e $y_i \in \{1, -1\}$ es la etiqueta de la clase a la que pertenece x_i . Decimos que $\mathbf{x}_1, \dots, \mathbf{x}_l \in \mathcal{X}$ donde $l \in \mathbb{N}$ es el número de observaciones y \mathcal{X} es un subconjunto compacto de \mathbb{R}^N . Una SVM determina el hiperplano que maximiza la separación entre ambas clases y es matemáticamente equivalente a maximizar el margen dado por $\frac{2}{\|\mathbf{w}\|}$ sujeto a $y_i(x_i \cdot \mathbf{w}) + b \geq 1$. El hiperplano $(\mathbf{w} \cdot \mathbf{x}) + b = 0$ queda determinado por b y \mathbf{w} . Se puede relajar la condición de muestras de entrenamiento separables incluyendo variables artificiales que permitan mejorar también la generalización. Para obtener el hiperplano minimizamos

$$\begin{aligned} \text{mín} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i \\ \text{sujeto a} \quad & y_i[(\mathbf{w} \cdot \mathbf{x}_i) + b] \geq 1 - \xi_i \end{aligned} \tag{1}$$

donde C controla el compromiso entre complejidad del hiperplano y el error sobre el conjunto de entrenamiento. Resolviendo el Lagrangiano en la forma dual obtenemos la función de decisión como:

$$f(\mathbf{x}) = \text{sgn}\left[\sum_i \alpha_i y_i (x \cdot x_i) + b\right]$$

donde α_i son los multiplicadores de Lagrange. Si las clases no son linealmente separables es posible extender el método proyectando los patrones en un espacio de mayor dimensión donde si sea posible clasificarlos linealmente, para luego obtener la función de decisión no lineal en el espacio original. Pueden ser utilizados distintos tipos de *kernels* que corresponden a estimadores no lineales en el espacio de entrada. En particular en este trabajo es utilizado un kernel de base radial $k(x, x_i) = e^{-\gamma \|x - x_i\|^2}$, γ es el parámetro que determina el ancho del kernel.

Han sido propuestos dos algoritmos basados en SVM para operar con una clase, *support vector data description* [15] y *one-class SVM (OCSVM)* [16]. El algoritmo OCSVM estima una función de decisión f que es positiva para un subconjunto del espacio de características y negativa para el complemento. El mapeo de los datos al espacio de características corresponde al kernel y separa los datos del origen con máximo margen. Sea $\Phi : \mathcal{X} \rightarrow \mathcal{F}$ un mapeo de características en \mathcal{F} de tal modo que el producto interno en la imagen de Φ puede ser calculado evaluando un kernel simple.

$$k(\mathbf{x}, \mathbf{z}) = (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}))$$

Se puede reescribir como un problema de optimización.

$$\begin{aligned} \text{mín}_{\mathbf{w} \in \mathcal{F}, \xi \in \mathbb{R}^l, \rho \in \mathbb{R}} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + \frac{1}{\nu l} \sum_i \xi_i - \rho \\ \text{sujeto a} \quad & (\mathbf{w} \cdot \Phi(\mathbf{x}_i)) \geq \rho - \xi_i, \xi_i \geq 0, \end{aligned} \tag{2}$$

donde $\nu \in (0, 1]$ es un parámetro que controla el término de penalización y ξ_i son variables de holgura. Resolviendo el problema de optimización (1) obtenemos \mathbf{w}

y ρ y la función de decisión es -1 para valores atípicos en el conjunto de datos y +1 para el resto de los patrones en el conjunto de datos.

$$f(\mathbf{x}) = \text{sgn}(\mathbf{w} \cdot \Phi(\mathbf{x})) - \rho \quad (3)$$

Introduciendo multiplicadores de Lagrange $\alpha_i, \beta_i \geq 0$, obtenemos

$$\begin{aligned} L(\mathbf{w}, \xi, \rho, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2} \|\mathbf{w}\|^2 + \frac{1}{\nu l} \sum_i \xi_i - \sum_i \beta_i \xi_i - \rho \\ &\quad - \sum_i \alpha_i (\mathbf{w} \cdot \Phi(\mathbf{x}_i) - \rho + \xi_i) \end{aligned}$$

Fijando las derivadas con respecto a las variables primarias \mathbf{w}, ξ, ρ igual a cero obtenemos

$$\begin{aligned} \mathbf{w} &= \sum_i \alpha_i \Phi(\mathbf{x}_i), \\ \alpha_i &= \frac{1}{\nu l} - \beta_i \leq \frac{1}{\nu l}, \\ \sum_i \alpha_i &= 1. \end{aligned}$$

La función de decisión puede ser escrita como

$$f(\mathbf{x}) = \text{sgn}\left(\sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) - \rho\right)$$

Los multiplicadores α_i se pueden resolver en el problema dual:

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{mín}} \quad & \frac{1}{2} \sum_{ij} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{sujeto a } & 0 \leq \alpha_i \leq \frac{1}{\nu l}, \quad \sum_i \alpha_i = 1. \end{aligned}$$

El parámetro ρ puede ser obtenido notando que para cualquier α_i y el correspondiente patron \mathbf{x}_i se satisface que:

$$\rho = (\mathbf{w} \cdot \Phi(\mathbf{x}_i)) = \sum_j \alpha_j k(\mathbf{x}_i, \mathbf{x}_j). \quad (4)$$

3. Supresión de plano de fondo por medio de OCSVM

El método propuesto consta de dos modos de operación posible y dos etapas cada uno. La primera etapa determina el modelo del fondo que en una segunda etapa permite suprimir el plano de fondo e identificar los *pixels* del *foreground*. El modelo de *background* se determina para cada pixel mediante el cálculo de los parámetros necesarios para definir la máquina de soporte vectorial de una clase como en la sección 2 a partir de un arreglo de imágenes de entrenamiento. El segundo modo de operación es una extensión del primer modo que explota la capacidad de las cámaras TOF de adquirir secuencias rápidas de datos. La figura (2) muestra la estructura de datos adoptada para ambas modalidades.

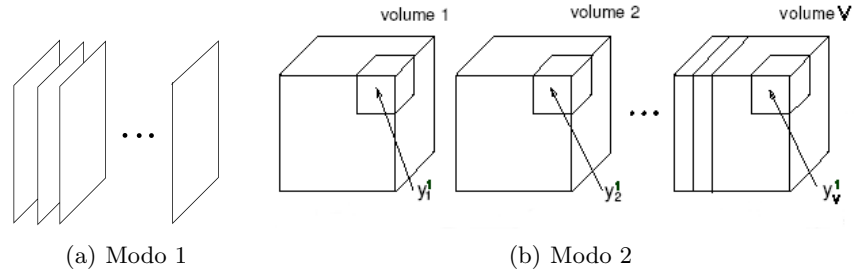


Figura 2. (a) Imágenes de entrenamiento, (b) Voxel correspondiente en V volúmenes

El primer modo de operación propuesto para la etapa de entrenamiento utiliza dos arreglos de imágenes, TOF_r y TOF_g . Cada uno consta de l imágenes $2\frac{1}{2}D$ y $2D$ representando el valor de rango y de intensidad respectivamente. Se calculan los parámetros que definen el modelo aplicando el procedimiento descrito en la sección 2 considerando l vectores de entrenamiento, $k = 1...l$, para cada pixel (i, j) , siendo:

$$\mathbf{x}_{i,j}^k = [x_1, x_2]^T$$

el k -ésimo vector de entrenamiento para el pixel (i, j) donde:

$$x_1 = TOF_r^k(i, j) \quad y \quad x_2 = TOF_g^k(i, j)$$

En la segunda etapa para cada nueva imagen de tiempo de vuelo provista por la cámara TOF se determina para cada pixel su pertenencia al fondo o al objeto de interés mediante la siguiente ecuación:

$$f(\mathbf{x}_{i,j}) = \text{sgn}(\mathbf{w} \cdot \Phi(\mathbf{x}_{i,j})) - \rho \quad (5)$$

El segundo modo de operación propuesto para la etapa de entrenamiento utiliza dos arreglos de l volúmenes ($v = 1...l$) conformados por arreglos de imágenes, TOF_r y TOF_g . Cada uno consta de M imágenes $2\frac{1}{2}D$ y $2D$ representando el valor de rango y de intensidad respectivamente. Se calculan los parámetros correspondiente para definir un clasificador OCSVM a partir de V vectores $v = 1...l$ para cada *voxel (volumen element)* (i, j, k) , siendo:

$$\mathbf{x}_{i,j,k}^v = [x_1, x_2]^T$$

el v -ésimo vector de entrenamiento para el voxel (i, j, k) donde:

$$x_1 = TOF_r^v(i, j, k) \quad y \quad x_2 = TOF_g^v(i, j, k)$$

En la segunda etapa para cada nueva imagen de tiempo de vuelo provista por la cámara TOF se determina para cada voxel su pertenencia al fondo o al objeto de interés mediante la siguiente ecuación:

$$f(\mathbf{x}_{i,j,k}) = \text{sgn}(\mathbf{w} \cdot \Phi(\mathbf{x}_{i,j,k})) - \rho \quad (6)$$

La resolución temporal tiene una cota máxima determinada por la cámara TOF utilizada, pero las características del problema particular a resolver, son las que determinarán el diseño adecuado y el compromiso entre velocidad, memoria disponible y complejidad.

4. Resultados Experimentales

La evaluación del método propuesto fue realizada utilizando secuencias de datos de tiempo de vuelo de rango e intensidad. Se generaron imágenes TOF artificiales simples con diferentes niveles y tipo de ruido. Se utilizaron también imágenes con un objeto sintetizado y con un objeto de *foreground* real. La figura 3 muestra una imagen sintética y una imagen real de rango e intensidad. Las imágenes sintéticas utilizadas fueron de 10x10x3 voxels por volumen y 20 volúmenes. Todos los voxels tienen sumado ruido gaussiano de media 16000 y desvío estándar 4000, figura 3(a). El valor de los voxels simulando *foreground* fue modificado entre 600 y 1500 con incrementos de 100 generando así 10 imágenes sintéticas denominadas TOF1 a TOF10 con ruido gaussiano. Otras 10 imágenes artificiales fueron generadas con ruido Rayleigh denominadas TOF11 a TOF20.

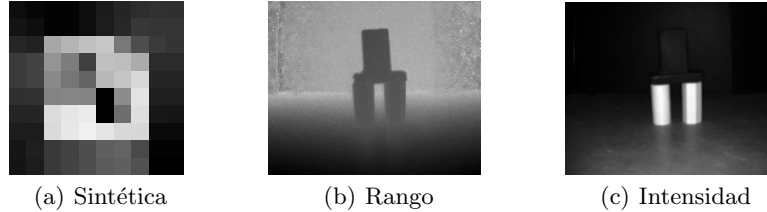


Figura 3. Imágenes de tiempo de vuelo

Las imágenes reales fueron adquiridas con una cámara TOF MESA SwissRanger SR4000 [17][18] que permite obtener imágenes de rango e intensidad de 144x176 pixels, hasta 50fps y rango de trabajo de 0,8 a 8m. Se construyó una escena simple y se adquirieron secuencias de imágenes, conformando un arreglo de imágenes TOF para construir el modelo de *background*, como muestran las figuras 3(b) y 3(c). Sobre una de las imágenes de fondo se sintetizó un objeto y la denominamos imagen de evaluación TOFS. También se adquirieron imágenes para la etapa de evaluación de un objeto real y denominamos al conjunto o a la imagen TOFR, utilizándose la notación de la sección 3, para diferenciar cada imagen o pixel en el arreglo. En la figura 4 podemos observar en color, simplemente para facilitar su identificación en la imagen, los objetos pertenecientes al *foreground* en las imágenes de evaluación. En las figuras 4(a) y 4(b) se muestran las imágenes de

los objetos sintetizados sobre la imagen artificial y sobre la real respectivamente. La imagen de la figura 4(c) se obtiene mediante el desplazamiento de un objeto del *background* y la captura de una nueva imagen TOF. En imágenes TOF reales no sólo existe ruido en la adquisición de las series temporales de cada pixel [19][20] sino también errores en la frontera entre objetos, efecto conocido como *flying pixels* [21][22].

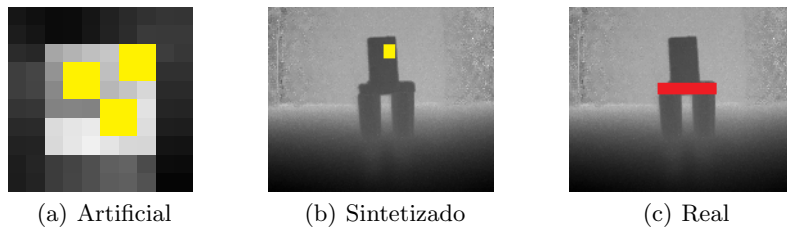


Figura 4. *Foreground* superpuesto en imágenes de rango

La tabla 1 presenta resultados comparativos expresados como la fracción de verdaderos positivos (TPF) y la fracción de falsos negativos (FPF). Estos valores corresponden al punto óptimo de operación en una curva ROC (*Receiving Operating Characteristics*). Se indican también otros datos de interés como la distancia al punto óptimo de operación y el área bajo la curva. Denominamos métodos C1 y C2 al método propuesto en [13] sin filtros morfológicos y utilizando procesamiento morfológico respectivamente. La figura 5 presenta las curvas ROC correspondientes a la evaluación con TOF9, TOF12 y TOFS.

Imagen TOF	Método	Area	d_{po}	TPF_{po}	FPF_{po}
TOF9	$Modo1_{\nu=0,37}$	0.989	0.661	0.979	0.046
TOF12	$Modo2_{\nu=0,4}$	0.977	0.693	0.892	0.049
TOFS	$Modo1_{\nu=0,38}$	0.986	0.645	0.955	0.043
	C_1	0.954	0.573	0.931	0.121
	C_2	0.975	0.618	0.935	0.060

Tabla 1. Métricas de rendimiento

La figura 6 muestra una imagen de rango con *foreground* real y el resultado de aplicar el método propuesto y suprimir el *background* en una imagen real.

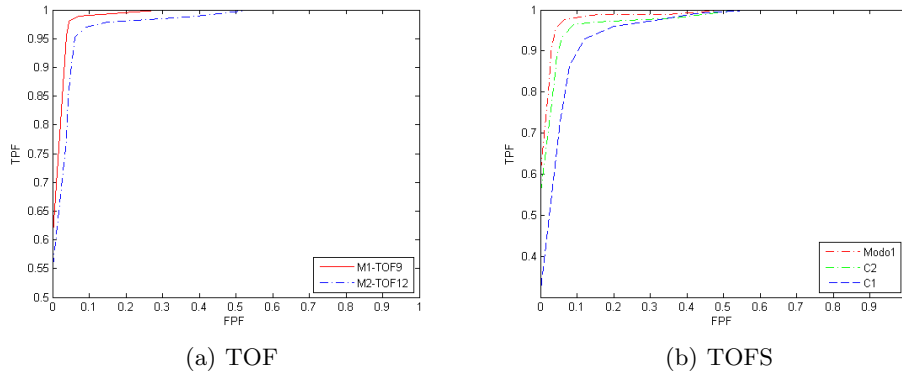


Figura 5. Curvas ROC

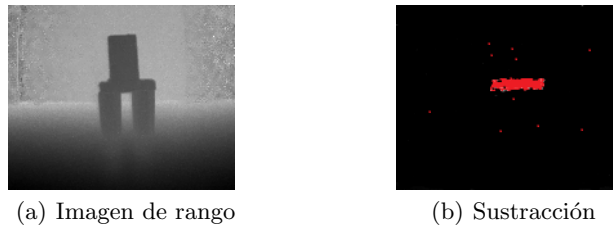


Figura 6. TOFR

5. Conclusiones

En este artículo se ha presentado un método que permite sustraer el plano de fondo en secuencias de imágenes de cámaras de tiempo de vuelo. Los resultados experimentales preliminares muestran que es viable, como resultado de combinar la información de rango e intensidad en un clasificador para clases desbalanceadas y basado en secuencias de pixels correspondientes. El método prevé una tercera etapa, que utilice la propiedad de correlación espacial de pixels o voxels y combine la información de rango e intensidad para filtrar anisotrópicamente o morfológicamente las imágenes y así permitir reducir ruido y *flying objects*. Una extensión futura de la evaluación experimental tiene previsto comparar los resultados obtenidos con métodos basados en bloque y considerar la significancia estadística de los resultados comparativos.

Referencias

1. P. Rajan, S. Prakash. Moving Foreground Object Detection and Background Subtraction Using Adaptive-K GMM: A Survey. International Journal of Advance Research in Computer Science and Management Studies, (2):300-308, 2014.

2. Maasimo Piccardi. Background subtraction techniques: a review. *IEEE International Conference on System. Man and Cybernetics*, 3099-3104, 2004.
3. T. Bouwmans, L. Davis, J. Gonzalez, M. Piccardi, C. Shan. Special Issue on Background Modeling for Foreground Detection in Real-World Dynamic Scenes. *Machine Vision and Applications*, (25):1101-1103, 2014.
4. A. Vacavant, L. Tougne, T. Chateau. Special section on background models comparison. *Computer Vision and Image Understanding*,(122), 2014.
5. Chris Stauffer. W.E.L Grimson. Adaptive background mixture models for real-time tracking. *Computer Vision and Pattern Recognition*, (2), 1999.
6. Rhandley D. Cajote et. al. Framework of Surveillance Video Analysis and Transmission System using Background Modeling and MIMO-OFDM. *IEEE International Conference on Digital Signal Processing*, 1071-1075, 2015.
7. F. Chiabrando, R. Chiabrando, D. Piatti, F. Rinaudo. Sensors for 3D imaging: Metric evaluation and calibration of a CCD/CMOS time-of-flight camera. *Sensors*, (9):10080–10096, 2009.
8. A. Kolb, E. Barth, R. Koch, R. Larsen. Time-of-Flight Sensors on Computer Graphics. *Proceedings of the Eurographics, Munich, Germany*, 2009.
9. S. Foix, G. Alenya, C. Torras. Lock-in Time-of-Flight (ToF) Cameras: A Survey. *IEEE Sensors Journal*, 11(9):1917-1926, 2011.
10. Thierry Oggier et. al. Novel Pixel Architecture with Inherent Background Suppression for 3D Time-of-Flight Imaging. *Proc. SPIE 5665, Videometrics VIII*, 2005.
11. P. Gil et. al . Calibración de cámaras de tiempo de vuelo: Ajuste adaptativo del tiempo de integración y análisis de la frecuencia de modulación. *Revista Iberoamericana de Automática e Informática industrial*, (10):453–464, 2013.
12. Assaf Glazer, Michael Lindenbaum, Shaul Markovitch. One-Class Background Model. *Computer Vision Workshop. Lecture Notes in Computer Science*, (7728):301-307, 2012.
13. Shung Han Cho, Kwanghyuk Bae, Kyu-Min Kyung, Seongyeong Jeong, Tae-Chan Kim. Background Subtraction Based Object Extraction for Time-of-Flight Sensor. *IEEE Global Conference on Consumer Electronics*, 48-49, 2013.
14. V. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988-999, 1999.
15. D. M. J. Tax, R. P. W. Duin. Support vector data description. *Machine Learning*, 54:45-66, 2004.
16. B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Samola, R. C. Williamson. Estimating the support of a high dimensional distribution. *Neural Computation*, 13(7):1443-1471, 2001.
17. F. Chiabrando, D. Piatti, F. Rinaudo. SR-4000 TOF Camera: Further Experimental Tests and First Applications to Metric Surveys. *Proceedings of V Symposium on Remote Sensing and Spatial Information Sciences*, 149-154, 2010.
18. M. Lindner, I. Schiller, A. Kolb, R. Koch. Time-of-flight sensor calibration for accurate range sensing. *Comput. Vis. Image Underst.*, (114):1318–18, 2010.
19. M. Reynolds et. al. Capturing Time-of-Flight data with confidence. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 945-952, 2011.
20. Frank Lenzen et. al. Denoising Strategies for Time-of-Flight Data, 2013. *Lecture Notes in Computer Science (8200):25-45, Springer*, 2013.
21. A. Sabov, J. Kruguer. Identification and Correction of Flying Pixels in Range Camera Data. *Proc. Computer Graphics. Slovakia*, 135-142, 2010.
22. H. Rapp, M. Frank, F.A. Hamprecht, B. Jahne. A theoretical and experimental investigation of the systematic errors and statistical uncertainties of time of flight cameras. *Int. J. Intell. Syst. Technol. Appl.*, (5):402–413, 2008.

Framework para la captura de eventos mediante interfaz natural de usuario

Juan Gabriel Fernandez¹, Gustavo Pereira¹, Jorge Ierache¹

¹Instituto de Sistemas Inteligentes y Enseñanza Experimental de la Robótica (ISIER)
Facultad de Informática Ciencias de la Comunicación y Técnicas Especiales
Universidad de Morón, Cabildo 134, (B1708JPD) Morón, Buenos Aires, Argentina
54 11 5627 2000 int. 189
jgfer@hotmail.com, pereira.gustavo@gmail.com, jierache@unimoron.edu.ar

Abstract: En el siguiente artículo se presenta una propuesta de framework abierto que permite integrar el uso de interfaces naturales para el control de objetos. La creación de este framework surge a raíz de la problemática encontrada en las diferentes soluciones que existen y sus respectivos frameworks asociados; Estas soluciones permiten el empleo de interfaces naturales en diferentes entornos, sin embargo todas ellas son aplicaciones nativas encapsuladas para cumplir un único propósito. Por tal motivo surgió la necesidad de la creación de un framework abierto para el control de objetos y la presentación del concepto de monitores que sustenta la creación del mismo. En particular se desarrollaron casos de prueba para el control de robots.

Keywords: Kinect, Interfaz Natural de Usuario, Framework, Control, Robots.

1 Introducción

Una interfaz natural de usuario es aquella que se utiliza para interactuar con un sistema, aplicación o dispositivo sin utilizar un mando o control de entrada (por ejemplo: ratón, teclado, joystick, etc.) siendo reemplazados directamente por la captura de los movimientos del cuerpo humano, tales como las manos, las piernas, la cabeza, etc. En los últimos años, la popularidad de las interfaces naturales, ha experimentado un crecimiento abrumador. Estas interfaces permiten a un usuario interactuar con una computadora o dispositivo de forma sencilla y sin la necesidad de un aprendizaje complejo y/o especializado. Entre los principales usos que se le ha dado al sensor Kinect se encuentra el de controlar objetos o sistemas mediante gestos. Esto se logra generando aplicaciones dedicadas a tal el fin. Existen varios desarrollos realizados con el Kinect para el control de objetos o sistemas, como por ejemplo el control de mapas virtuales, operaciones aritméticas básicas en Windows o control a distancia de robot con servomotores. Los desarrollos anteriormente mencionados utilizan el sensor Kinect para dar solución a una problemática determinada y se limitan a un único uso. Esto genera la necesidad de la creación de nuevos desarrollos que permitan ampliar las funcionalidades.

En este trabajo se propone y presenta la creación de un framework para el uso genérico de los datos capturados por el sensor Kinect de Microsoft [1], sumando la capacidad de generar una base de monitores abierta para cada punto capturado con independencia del dominio de aplicación; un monitor es un estado que determina si un punto (llamado joint para la nomenclatura que utiliza Kinect) se encuentra dentro de un rango de valores permitido. Estos monitores de puntos capturados facilitan la declaración de métodos que activan eventos locales o remotos en función del estado del monitor, estos se categorizan en estáticos y dinámicos. Se muestra en el siguiente artículo el estado actual de los desarrollos y frameworks realizados para Kinect para el control de objetos, para luego analizar sus problemáticas y plantear una solución. Además se exhibe el desarrollo del framework abierto y sus casos experimentales. Finalmente se presenta la conclusión y las futuras líneas de investigación.

2 Situación actual

El sensor de movimiento o sensor 3D Kinect es un dispositivo de control por movimiento creado originalmente para poder interactuar con la consola de juegos Microsoft Xbox 360[2] sin necesidad de ningún mando o control. La cámara RGB permite generar una imagen tridimensional de lo que tiene por delante. Para ello cuenta con un sensor de profundidad, que permite ver una habitación en tres dimensiones. Este sensor emite rayos infrarrojos que se proyectan sobre la escena con un patrón de 50.000 puntos, invisibles al ojo humano, con el fin de marcar las líneas de profundidad, medir el tiempo que demora en retornar el haz de luz con respecto al dispositivo y calcular la distancia al objeto identificado. Kinect permite la captura de movimiento a través de una serie de algoritmos como son: el **Randomized Decision Forests** [3], que es capaz de aprender ciertas características de un conjunto de imágenes de profundidad con el fin de clasificar los píxeles de una imagen para obtener la pose de un objeto y el **Mean-Shift** [4] que es capaz de estimar la posición correspondiente de las articulaciones del esqueleto. Aplicando dichos algoritmos la tecnología con la que cuenta Kinect es capaz de realizar el reconocimiento esquelético de un individuo en tiempo real [5] [6]. De las diferentes alternativas existentes a Kinect, en el uso de interfaces naturales, se pueden destacar las siguientes: Perceptive Pixel [7], Microsoft PixelSense [8], 3D Immersive Touch [9] y MYO [10]; las cuales permiten captar los gestos y movimientos del cuerpo humano utilizando diferentes tecnologías y técnicas.

2.1 Aplicaciones desarrolladas para Kinect

a) *Touch screen con Kinect* desarrollado por la firma Ubi Interactive [11]; ésta, interactuando con Microsoft, desarrolló un sistema que utiliza un proyector digital combinado con el sensor Kinect para crear una pantalla táctil virtual, que puede ser proyectada en cualquier superficie. Mediante la captura de los movimientos de las manos, se permite el control de las imágenes proyectadas. Teniendo un proyector y el sensor de

movimiento Kinect solo resta una PC con la aplicación que permite convertir cualquier superficie en táctil.

b) *Kinect para estudiar anatomía (Sistema Mirracle)*[12]. Sistema creado en la Universidad Técnica de Múnich, y permite instaurar la ilusión en la cual el usuario tiene “visión de rayos-x” en su propio cuerpo. Utilizando el Kinect de Microsoft para estimar la posición del usuario, y la base de datos del proyecto Visible Korean Human, la persona puede utilizar la interfaz de Kinect para navegar por su propio cuerpo. Existen trabajos de divulgación científica que presentan soluciones para el control de objetos o sistemas mediante la interface natural Kinect, tales como:

c) *Wiimote and Kinect: Gestural User Interfaces add a Natural third dimension to HCI* [13]. Las aplicaciones propuestas en este trabajo adoptan el seguimiento del movimiento del usuario a través de la Nintendo Wii y el sensor Kinect. Estas aplicaciones están específicamente diseñadas para la interacción gestual del usuario en mapas geográficos en 3D, como los utilizados por el motor de búsqueda Bing.

d) *Robust Hand Gesture Recognition with Kinect Sensor*[14]. Presenta el reconocimiento preciso y eficiente de gestos de la mano para aplicaciones básicas de la vida real, tales como: el cálculo aritmético y el juego piedra, papel o tijeras.

e) *Human Machine Interface HMI using Kinect sensor to control a SCARA* [15]. Plantea la posibilidad de manipular el robot SCARA (Selective Compliance Assembly Robot Arm) en tiempo real a través del sensor Kinect. Este desarrollo envía comandos por puerto USB a la controladora del SCARA.

3 Problemática

La problemática que se presenta en los desarrollos que existen, es la falta de generalidad a la hora de utilizar el sensor Kinect. Normalmente las aplicaciones creadas para Kinect determinan la forma en que se realizará la captura, estas definen los parámetros de ingreso y no pueden ser modificados por el usuario. De la misma forma lo hace con la salida o acciones que se realicen. Esto hace que estas aplicaciones tengan un uso bien definido y cerrado. Este acoplamiento no permite utilizar el sensor más que para su uso definido por la aplicación.

4 Solución Planteada

Se desarrolló un framework para uso genérico de los datos capturados por el sensor Kinect. El mismo, permite la creación de monitores para cada punto capturado por el sensor Kinect. Un monitor, como fue definido anteriormente, es un estado que determina si un punto (joint) se encuentra dentro de un rango de valores permitido. Para simplificar el concepto se usara uno de los joints entregados por el sensor Kinect. En un frame¹ determinado (de los treinta

¹ Se denomina frame a un fotograma o cuadro, una imagen particular dentro de una sucesión de imágenes que componen una animación. En esta caso la sucesión de datos que envía Kinect

por segundo que entrega el sensor Kinect) se tienen las coordenadas cartesianas (X, Y, Z) del joint seleccionado. Con esta información se puede hacer un análisis de las tres dimensiones del espacio, y a fin de simplificar el concepto solo se utilizarán las coordenadas X e Y. Con las capturas obtenidas se comienza la etapa de análisis de monitores. Se va a generar un monitor para el joint analizado. El monitor tendrá tres estados posibles: Activo, Inactivo, Nulo. Los estados Activo e Inactivo irán cambiando según se cumplan determinadas condiciones. El estado Nulo se refiere a la inhabilitación del monitor, lo que significa que si no se requiere su utilización los estados activo e inactivo no estarán disponibles. Para determinar que un monitor está activo o inactivo se definirá un rango de valores por cada eje cartesiano. De esta manera se tiene para X e Y un valor inicial (X_i), (Y_i) y un valor final (X_f), (Y_f). Teniendo definidos los rangos de X e Y se puede analizar el estado del monitor. El monitor relacionado con el punto seleccionado estará activo cuando los valores de X e Y estén comprendidos dentro de sus respectivos rangos: $X_i < X_{capturado} < X_f$; $Y_i < Y_{capturado} < Y_f$. Cuando el monitor esté dentro del rango de valores en un determinado frame, se lo considera como activo, en caso contrario su estado será el de inactivo. Se definió una región de valores en donde el monitor puede estar activo. A estos monitores los denominamos Estáticos. De esta manera se pueden generar N cantidad de monitores que estén relacionados a cada joint que entrega el Kinect. Cuando un monitor se active, se disparará la ejecución de uno o más métodos que serán generados para un fin determinado. Estos métodos desencadenarán acciones que podrán ser definidas para cumplir con un fin determinado. Estas acciones podrán ser ejecutadas dentro de un ámbito local o de manera remota. Se puede definir la cantidad de monitores y métodos que se deseen y relacionarlos con un punto en particular del cuerpo.

Continuando con el análisis de los monitores se presenta el caso donde se quiere conocer en qué momento una persona, que se encuentra delante del Sensor Kinect, está quieta, avanzando o retrocediendo. Con el concepto de monitor estático no es posible tener esta información ya que no basta con saber si un monitor se activa o no en un momento determinado. Se debe tener en cuenta cuál era la posición anterior para compararla con la posición actual y allí decidir qué acción se está realizando. Si se analiza la distancia entre el sensor y la persona se tiene que: Si el valor de la posición actual (Z actual) es menor que el valor de la posición anterior (Z anterior) entonces la persona está avanzando (Fig. 1). Si el valor de la posición actual (Z actual) es mayor que el valor de la posición anterior (Z anterior) entonces la persona está retrocediendo:

$Z_{actual} < Z_{anterior} \rightarrow Avanza$; $Z_{actual} > Z_{anterior} \rightarrow Retrocede$.

También se debe analizar si la persona está quieta, entonces se calcula la diferencia entre el Z actual y el Z anterior. Esta diferencia se denomina ΔZ y estará expresada en valor absoluto. Se debe fijar un umbral para la diferencia que aumentará o disminuirá según la necesidad de sensibilidad de la medición. Si ΔZ es menor al umbral definido, entonces la persona se encuentra quieta o detenida. A este monitor lo denominamos Monitor Dinámico porque tiene en cuenta la diferencia entre el estado actual y el anterior. Se puede esquematizar a los monitores y sus características en un diagrama que se muestra en la Fig. 2.

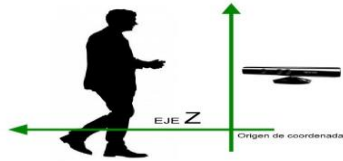


Fig. 1 Eje Z obtenido por Kinect

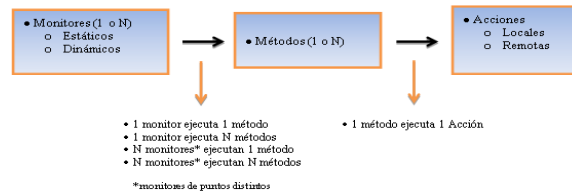


Fig. 2 Esquema de monitores y sus relaciones con otros elementos

El aplicativo internamente generará un string² de conexión [16] para comunicarse con el mundo exterior. La estructura de esta cadena ha sido definida arbitrariamente, tal como se observa en la Fig. 3. Donde **Time**: se presenta en segundos, representa el tiempo exacto cuando se realiza la captura. Se incrementará hasta llegar al tamaño de la captura y se repetirá según la cantidad de frames que se capturen. En este caso se repetirá 30 veces antes de pasar al siguiente, **Frame**: Representa la mínima unidad de captura de Kinect. (30 frames por segundo), **X, Y (01)**: Valores de X, Y en el joint 01 (Joints del 1 al 20), **X, Y (20)**: Valores de X, Y en el joint 20 (Joints del 1 al 20), **M01,..., M02**: Monitores asociados a los joints, **Tamaño Captura**: Tiempo total de captura. (En segundos) A fines prácticos solo se representarán los joints 1 y 20, pero en la captura final todos los joints estarán incluidos.

Time	Frame	X01	Y01	M01	...	X20	Y20	M20	Tamaño Captura
------	-------	-----	-----	-----	-----	-----	-----	-----	----------------

Fig. 3. String de conexión.

5 Framework para interfaz natural de usuario

Se presentan las principales funciones del framework genérico para interfaz humana. En la Fig.4 se muestra un diagrama de clases resumido del desarrollo. Las clases KinectSensor, Joint y Skeleton están relacionadas directamente con las librerías para C# de Kinect. La clase Skeleton contiene una colección de joints y a su vez la clase KinectSensor contiene una colección de Skeletons. La clase Monitor utiliza los métodos de KinectSensor para definir su estado y entregar el mismo. La clase Asignación relaciona al Monitor y al Método y este último ejecuta sus acciones cuando el Monitor este activado. Por último la clase método contiene la secuencia de comandos que se desean realizar. Estos comandos tienen

² Se denomina string a una cadena de caracteres ordenada que contiene datos sobre la captura de movimiento de Kinect para ser enviada y utilizada posteriormente

un formato determinado que permite a la Clase Manejador determinar la manera en que serán ejecutados, en otras palabras, que interfaz de salida se utilizará.

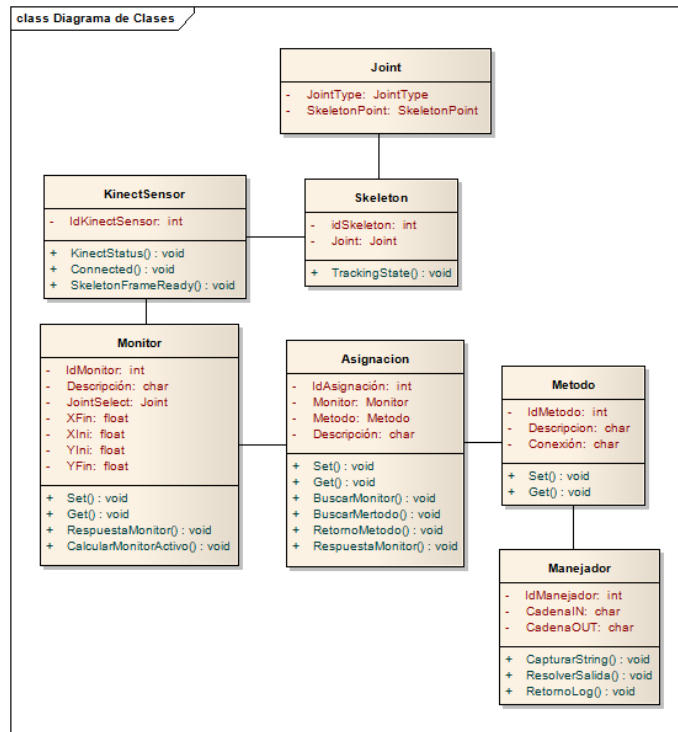


Fig. 4. Diagrama resumido de clases

Se presenta un diagrama de secuencia de la funcionalidad de captura en la Fig. 5. Esta funcionalidad se repetirá continuamente, siempre que el sensor Kinect detecte un skeleton. Primero se obtienen las coordenadas de los puntos del skeleton por medio de la clase KinectSensor. Con estos datos se ejecutan los métodos de la clase Monitor que indican si este se encuentra activo. Cuando se encuentra activo se busca el comando a ser ejecutado en la clase Métodos. Por último la clase Manejador decide por cuál interfaz de salida será enviado el comando.

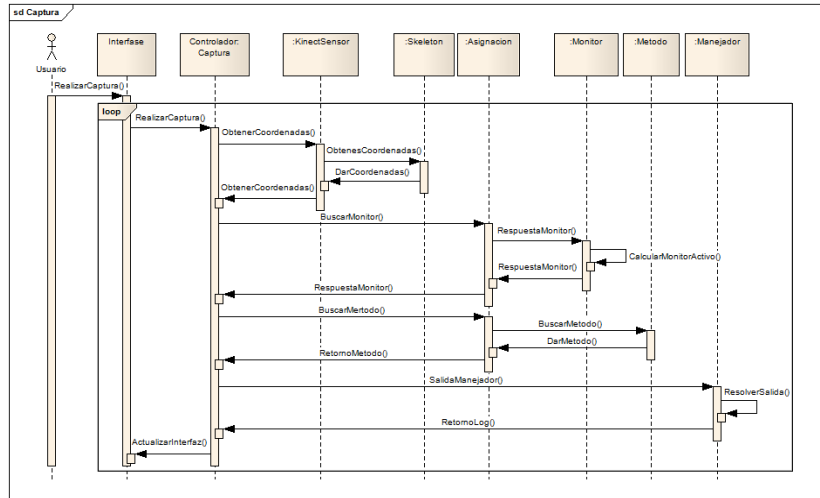


Fig. 5. Diagrama de secuencia de la captura del sensor

La gestión de alta, baja y modificación de monitores, se efectúa a través de una lista desplegable, que permite determinar qué joint será analizado por el monitor. Luego se ingresan los valores iniciales y finales de X e Y por medio de controles deslizantes. En la Fig. 6a, se puede observar una representación del lugar en el plano donde el monitor estará activo, resaltándolo en color rojo. Los valores máximos y mínimos de los controles deslizantes van desde -1 al 1, esto se debe a los datos arrojados por Kinect y a la posición en el que está ubicado; también se tiene en cuenta la posición del usuario frente a él. El sensor ha sido ubicado a una altura de 80 centímetros del suelo y el usuario estará a una distancia de 2 metros del mismo aproximadamente. Se han seleccionado estos valores luego de varias pruebas con el sensor ya que se realizaron muestras de los joints capturados a la distancia anteriormente definida. Para tener una referencia más clara, el valor 1 entregado por Kinect representa 1 metro. Se debe tener en cuenta que el centro de la captura es el origen de coordenadas.

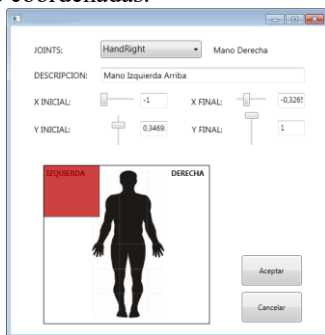


Fig. 6a. Creación monitor

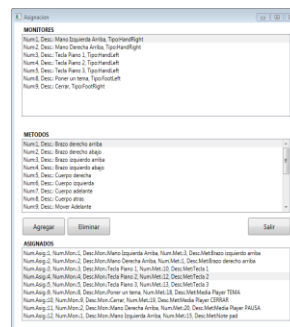


Fig. 6b. Asignación de métodos a monitores

6 Desarrollo de un caso de Aplicación del control de Robots

Se describe un caso de aplicación asociado a la creación de monitores detallados en la sección anterior, aplicando a un robot Robosapien V2 [17], sobre el cual se desarrollaron parte de las pruebas del sistema, en este orden se han agregado métodos predeterminados afines a las acciones del que se pretende controlar. Habiendo generado los monitores, en el siguiente apartado, se relacionará a estos con los métodos que se desean ejecutar, por ejemplo el control a través de la mano derecha arriba. Fig. 6b. Por último se realiza la captura y se indica qué monitores están siendo activados, y por consecuencia qué método se está ejecutando, esto se muestra resaltando en color rojo en la Fig. 7a. Como resultado de esta acción el Robosapien V2 moverá su mano izquierda como se muestra en la Fig. 7b.

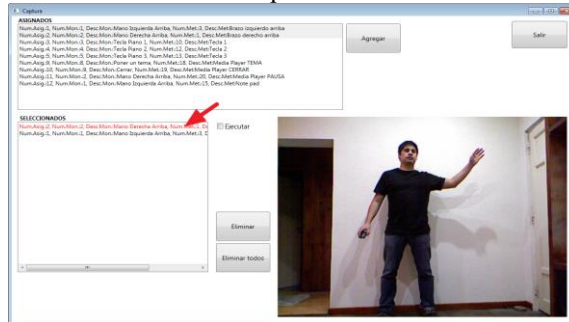


Fig. 7a. Captura de datos



Fig. 7b Robosapien V2 en movimiento.

Habiendo generado el string de conexión detallado en la sección anterior y mediante un proceso de traducción, realizado por un intérprete o interfaz, se envía la orden específica al Robosapien V2 [18] el cual es controlado por medio de comandos IR (infrarrojo) enviados mediante un dispositivo de emisión y recepción de señales infrarrojas, como USB-UIRT³[19]. Previamente se capturan los principales comandos de movimiento del Robosapien V2 y se relacionan a los métodos de salida de la aplicación. En el momento en que el monitor es activado se pasa a ejecutar el método; la interfaz relacionada, por medio del dispositivo USB-UIRT, transmite la señal IR al Robosapien V2 y éste realiza la acción indicada (Fig. 8).



Fig. 8. Esquema de conexión.

³ Permite a cualquier PC equipado con un puerto USB transmitir y recibir señales infrarrojas hacia cualquier dispositivo multimedia compatible con IR.

El desarrollo del framework se encuentra en la etapa de prototipo de pruebas y fue desarrollado en Microsoft Visual Studio 2010 Express [20] junto con el SDK de Kinect para Windows [21], además se utilizó la librería Usb-Uirt-managed-wrapper [22] para poder utilizar USB-UIRT en C#.

Se puede observar un video donde se presenta una demostración de la aplicación controlando el Robosapien V2 [23]. Se desarrollaron otros casos de prueba de control de un robot donde se pretende que el robot (Roboreptile [24]) se pare en dos patas al realizar el gesto de levantar la mano. Se controla el movimiento lateral de la cabeza del robot cuando el gesto del usuario es el mismo. En la Fig. 9 se pueden ver los movimientos del robot y además se provee un video con la demostración [25].



Fig. 9. Roboreptile en movimiento

7 Conclusiones y futuras líneas de trabajo

La utilización de este framework permite hacer uso de una interfaz natural de forma genérica y genera un bajo acoplamiento con los elementos de su entorno, lo que permite el aprovechamiento del software desarrollado, ya que éste puede ser reutilizado para desarrollos posteriores. Además, los datos de movimiento entregados por el sensor Kinect y los comandos enviados al receptor son totalmente independientes entre sí, permitiendo emitir órdenes de control a diversos dispositivos.

Se destaca que la entrega de datos de control podría realizarse de manera remota, por consiguiente, podría controlarse un dispositivo a grandes distancias, esto se hace posible generando y enviando un string estructurado de conexión.

Las futuras líneas de trabajo se centran en el desarrollo de monitores que faciliten su adecuación en forma dinámica y en este orden también la explotación de información de los monitores bajo un motor de inferencias que explote la información de un motor de reglas para los monitores, métodos y eventos.

8 Referencias

1. Página Oficial de Microsoft Kinect para Windows
<http://www.microsoft.com/en-us/kinectforwindows/>
2. Xbox 360 Pagina official, <http://www.xbox.com/es-ES/xbox-360>
3. Randomized Decision Forests
http://research.microsoft.com/pubs/155552/decisionForests_MSR_TR_2011_114.pdf
4. Mean Shift
http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/TUZEL1/MeanShift.pdf
5. Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-Time Human Pose Recognition in Parts from Single Depth Images. 2013, 22 de Junio.
<http://research.microsoft.com/pubs/145347/BodyPartRecognition.pdf>
6. Richard Szeliski. (2013, 23 de Junio). Computer Vision: Algorithms and Applications.
http://szeliski.org/Book/drafts/SzeliskiBook_20100903_draft.pdf
7. Perceptive Pixel, <http://www.microsoft.com/office/perceptivepixel/>
8. Microsoft PixelSense, <http://www.microsoft.com/en-us/pixelsense/default.aspx>
9. 3D Immersive Touch, <http://edusim3d.com/>
10. MYO: <https://www.myo.com/>
11. Ubi Interactive - Kinect touch screen www.ubi-interactive.com
12. Sistema Mirracle <http://campar.in.tum.de/WebHome>
13. Wimote and Kinect: Gestural User Interfaces add a Natural third dimension to HCI
<http://www.infomus.org/Events/proceedings/AVI2012/CHAPTER%203/p116-francese.pdf>
14. Robust Hand Gesture Recognition with Kinect Sensor
<http://eeewebeba.ntu.edu.sg/computervision/Research%20Papers/2011/Robust%20Hand%20Gesture%20Recognition%20with%20Kinect%20Sensor.pdf>
15. Human Machine Interface HMI using Kinect sensor to control a SCARA
http://carlospillajo.info/wp-content/uploads/sites/1369/2014/07/COLCOM2013_Track5-A.pdf
16. MSDN - Sintaxis de cadena de conexión, <http://msdn.microsoft.com/es-es/library/ms254500.aspx>
17. Robosapien V2 <http://wowwee.com/robosapien-x/>
18. Manual Robosapien V2
http://www.theoldrobots.com/images7b/Robosapien_V2_Manual.pdf
19. USB-UIRT, <http://www.usbuirt.com/>
20. Página Oficial de Microsoft Visual Studio 2010 express,
<http://www.microsoft.com/visualstudio/esn/products/visual-studio-2010-express>
21. Kinect SDK, <http://www.microsoft.com/en-us/kinectforwindows/develop/learn.aspx>
22. JordanZaerr/Usb-Uirt-managed-wrapper, <https://github.com/JordanZaerr/Usb-Uirt-managed-wrapper>
23. Video caso de prueba Robosapien V2
<https://1drv.ms/f/s!At7ZJI6Nbue6l19N6HsmQ-Dh1u3L>
24. Manual Roboreptile
http://www.theoldrobots.com/images7b/Roboreptile_Manual.pdf
25. Video caso de prueba Roboreptile
<https://1drv.ms/f/s!At7ZJI6Nbue6lmDuKmZbn4swtTbu>

Comparación de Sistemas Operativos Embebidos sobre la Computadora Industrial Abierta Argentina

Medina Santiago¹, Pi Puig Martín¹, Dell'Oso Matías¹, Romero Fernando¹, De Giusti Armando^{1,2}, Tinetti Fernando G.^{1,3}

¹Instituto de Investigación en Informática LIDI (III-LIDI), Facultad de Informática, Universidad Nacional de La Plata, 50 y 120 2do piso, La Plata, Argentina.

²CONICET – Consejo Nacional de Investigaciones Científicas y Técnicas

³CIC – Comisión de Investigaciones de la Pcia. de Buenos Aires

{smedina, mpipuig, mdelloso, fromero, degiusti, fernando}@lidi.info.unlp.edu.ar

Abstract. En el siguiente trabajo se presenta una comparación de dos sistemas operativos embebidos aptos para uso en Sistemas de Tiempo Real (STR). Los puntos de comparación abarcan aspectos tales como planificadores, manejo de memoria, portabilidad, entre otros; en esta evaluación se hizo hincapié en la diferencia principal entre un Sistema operativo estático y uno dinámico. Se respaldó la comparación para obtener conclusiones concretas, realizándose mediciones sobre los tiempos en el manejo de tareas y el tiempo de respuesta a un estímulo externo.

Keywords: Tiempo Real, Sistemas Operativos, Sistemas Embebidos, Latencia, Planificación.

1 Introducción

Los Sistemas de Tiempo Real (STR) [1] [2] [3] [4] son aquellos que permiten proveer en forma garantizada un servicio dentro de un intervalo de tiempo de respuesta limitado. Esta restricción exige un cuidadoso diseño tanto del hardware como del software.

Dentro de los STR podemos distinguir:

- 1) Los que no usan SO: muchas veces en STR pequeños, el software no incluye un SO.
- 2) Los que usan SOE con MMU: estos tienen capacidad de gestionar multiprogramación, requieren un hardware más complejo. Por lo general, utilizan capacidades gráficas con lo que mejoran la interacción con los operadores. Muchos son derivados de SO convencionales: RT-Linux, RTAI, Linux RT-Preempt.

3) Los Sistemas Operativos Embebidos (SOE): si bien todos los sistemas de tiempo real suelen estar embebidos en un sistema mayor al cual controlan, los que tienen un SOE se caracterizan porque no se les puede agregar otros programas, solo tiene los que se compilan en forma conjunta con dicho SOE.

Los sistemas operativos embebidos son en general de mínimo tamaño. Sus principales características son:

- Se usan en sistemas embebidos, o sea aquellos que forman parte y controlan un sistema mayor.
- Aseguran rápidos cambios de contexto, dirigido a bajar la latencia del planificador y de las interrupciones.
- Bajo intercambio entre almacenamiento secundario y memoria.
- Gestión de archivos orientada a la velocidad de acceso más que a la eficiencia.
- No son eficientes procesando información. Por lo general no utilizan procesadores complejos.
- Se usan procesadores predecibles más que rápidos.
- No se utiliza memoria virtual, para evitar variaciones en los tiempos de acceso a memoria.
- Tiene un tiempo de respuesta acotado tanto a las interrupciones como a eventos que disparen tareas, tales como alarmas. Esto se debe a un cuidadoso diseño en esta área, principalmente del planificador.
- Objetivos muy limitados, por lo general pocas tareas.
- Diseñados para soportar numerosas arquitecturas y montarse sobre muchos dispositivos.
- Funcionan sobre hardware orientado a la entrada salida más que al procesamiento, con poca memoria, tales como los microcontroladores.

Luego, los SOE pueden clasificarse por la manera en que se realiza la asignación de memoria del sistema que controlan. Los mismos, pueden ser estáticos, en los cuales la asignación de recursos se realiza en tiempo de compilación, o dinámicos, donde se realiza en tiempo de ejecución.

En este trabajo se lleva a cabo una comparación de características de dos Sistemas Operativos Embebidos que corren sobre la placa de desarrollo EDU-CIAA-NXP, el primero de ellos es FreeOSEK, estático y el segundo FreeRTOS, dinámico.

Estos SOE se seleccionaron entre varios de este tipo ya que tienen cada uno características que los destacan entre los demás. Ambos han adquirido gran popularidad y una gran cantidad de usuarios en ambos casos; esto llevó a su portabilidad a una gran cantidad de

plataformas hardware. En ambos, la comunidad que presta soporte es numerosa, con lo que puede conseguirse abundante documentación y ejemplos. No es el caso de otros SOE que fueron desarrollados para arquitecturas específicas: MQX para Freescale (NXP), ERIKA para Microchip, etc.

2 Objetivos y metodología

Este trabajo tiene como objetivo comparar dos SOE, FreeOSEK y FreeRTOS corriendo sobre la misma plataforma de hardware a fin de evaluar las características fundamentales y necesarias en los Sistemas de Tiempo Real.

La metodología utilizada consiste en la programación de escenarios de pruebas que simulan la misma situación en ambos SOE, sobre la misma placa de desarrollo.

2.1 Hardware utilizado

Se utilizó una placa EDU-CIAA-NXP [13]. Está basada en la placa LPC4337. Cuenta con un microcontrolador ARM Cortex-M4 que incluye un coprocesador ARM Cortex-M0, 1 MB de memoria flash, 136 kB de SRAM, 16 kB de memoria EEPROM, periféricos como el timer de estado configurable (State Configurable Timer) (SCT) y el Serial General Purpose I/O (SGPIO) interface, dos controladores de alta velocidad USB, Ethernet, LCD, un controlador de memoria externa y múltiples entradas/salidas digitales y entradas analógicas. Opera a una frecuencia de reloj de más de 204 MHz. El ARM Cortex-M4 es la próxima generación de cores de 32 bit que ofrece bajo consumo de energía, mejoras en las características de debug, y alto nivel para soportar integración de bloques. Incorpora un pipeline de 3 etapas, arquitectura Harvard con buses separados para datos e instrucciones y un tercer bus para periféricos. Incluye una unidad interna de prebúsqueda que soporta ejecución especulativa en bifurcaciones. Soporta procesamiento de señales en un ciclo e instrucciones SIMD (Single Instruction Múltiple Data). También, tiene integrado en el core hardware para procesamiento en punto flotante.

El coprocesador ARM Cortex-M0 es de uso eficiente de energía, de 32 bit compatible en código y herramientas con el Cortex-M4. El coprocesador Cortex-M0, está diseñado para reemplazar los microcontroladores existentes de 8/16-bit.

Otras características del ARM Cortex-M4 son:

- Memoria interna
- Soporta protección de memoria (Memory Protection Unit) de ocho regiones
- Nested Vectored Interrupt Controller (NVIC)
- Unidad de Hardware de punto flotante
- Entrada de interrupción no enmascarable (Non-maskable Interrupt)
- JTAG y Serial Wire Debug (SWD), serial trace y ocho breakpoints
- System tick timer

El Cortex-M0 es capaz de liberar la carga del procesador ARM Cortex-M4. En él se destaca:

- Velocidad de reloj de 204 MHz
- JTAG, Serial Wire Debug, NVIC built-in
- Memoria on-chip, hasta 1 MB flash de dos bancos
- 16 kB de memoria de datos on-chip EEPROM
- 136 kB de SRAM para código y uso de datos
- Múltiples bloques SRAM con acceso separado al bus
- 64 kB de ROM con el código de arranque y drivers de software on-chip
- 32 bit de memoria de propósito general One-Time Programmable (OTP)
- Interfaz Serial GPIO (SGPIO)
- Subsistema State Configurable Timer (SCT) en AHB
- Dispone de un Global Input Multiplexer Array (GIMA) para conectar múltiples entradas y salidas hacia periféricos manejados por eventos

2.2 Sistemas Operativos Evaluados

La base de este trabajo de investigación radica en la comparación entre un sistema operativo estático, FreeOSEK, y uno dinámico, FreeRTOS, teniendo en cuenta que estos tienen una estructura de desarrollo diferente.

En los dinámicos, la asignación de memoria, creación y definición de tareas, semáforos y demás características propias del sistema, se llevan a cabo en tiempo ejecución, permitiendo así mayor flexibilidad en el desarrollo con menos límites en el manejo de tareas. Esto lleva a una reducción del control que tiene el programador en la ejecución del programa y por ende genera algunos riesgos que implican no cumplir tiempos límites.

Luego, en los sistemas estáticos, las tareas y todas sus características, los recursos que utilizan, los eventos y las interrupciones que manejan, conjunto con la asignación de memoria, deben ser definidas antes de compilar el código, durante la programación. Esta forma de desarrollo le brinda al programador un mayor control sobre la aplicación, debido a un mayor determinismo en la ejecución de tareas y los tiempos de respuesta en cada caso. Sin embargo, esta ventaja implica un mayor esfuerzo en la programación y limitaciones en tiempo de ejecución, ya que si se define erróneamente algún aspecto de la planificación o de una tarea, se debe programar el código nuevamente. Además, la asignación estática requiere una mayor cantidad de memoria respecto al esquema dinámico, ya que durante la ejecución no es posible liberar el espacio reservado.

2.2.1 FreeRTOS

FreeRTOS [5] [6] [7] [8] [9] es un sistema operativo de tiempo real para sistemas embebidos que puede funcionar sobre 35 arquitecturas de microcontroladores. Se distribuye bajo licencia GPL con una excepción: permite que el código propietario de los usuarios siga siendo código cerrado, manteniendo el núcleo en sí como código abierto, lo que facilita el uso de FreeRTOS en aplicaciones propietarias. Está formado por un conjunto de archivos en lenguaje C. A su vez, en algunas arquitecturas, incluye algunas líneas de código ensamblador, mayormente en las rutinas del planificador.

FreeRTOS provee:

- Poco uso de memoria
- Bajo nivel de sobrecarga
- Ejecución rápida
- Métodos para múltiples hilos o tareas
- Mutexes y semáforos
- Timers de software
- Reducción del tick para aplicaciones de bajo consumo.
- Prioridades para las tareas
- Cuatro esquemas de asignación de memoria
- Es de código abierto: está bajo continuo desarrollo, y no hay costo de implementación.
- Amplia comunidad de usuarios: ayuda a resolver problemas que surgen.

El método de múltiples hilos cambia tareas en función de la prioridad y un esquema de planificación round-robin. El planificador puede ser configurado para operar en forma apropiativa o colaborativa. Las tareas tienen una prioridad de ejecución, donde 0 es la menor prioridad (se recomienda usar referencias a `tskIDLE_PRIORITY +1,+2`, etc).

Cada tarea tiene una prioridad asignada entre 0 y un valor definido en compilación `configMAX_PRIORITIES - 1`. Es decir, si `configMAX_PRIORITIES` es 5, FreeRTOS tiene 5 niveles de prioridad entre 0 y 4.

2.2.2 FreeOSEK

Es un sistema operativo de tiempo real especialmente diseñado para sistemas embebidos críticos donde la ejecución determinística del sistema es un punto esencial [12].

Fue creado por la industria automotriz y es de libre licencia. Su ventaja respecto de otros SOE es que es escalable y está basado en las especificaciones OSEK-VDX, un estándar. Por ende, existen muchas implementaciones (open source y cerradas). Además, existen muchas empresas que proveen soporte.

El planificador OSEK usa una FIFO por prioridad. Existen tareas que pueden ser interrumpidas (SCHEDULE = FULL) y otras que no (SCHEDULE = NON). En la configuración se pueden dar tres opciones: que todas las tareas puedan ser interrumpidas, que ninguna de ellas o una mezcla de ambos tipos de scheduling. En función a esto, el sistema operativo puede contener una parte del código, la otra o ser un scheduling con todas las posibilidades y que se decida en tiempo de ejecución cual tarea puede o no ser interrumpida.

Este SOE ofrece rutinas de servicio de interrupción (ISR) de dos tipos:

- ISR1: Sin llamadas a la API, más simple y más rápido
- ISR2: Con llamadas a la API, puede llamar a algunas primitivas del SOE

Las ISR1 tienen siempre prioridad más alta que las ISR2 y, estas últimas, ejecutan el planificador al final de la rutina.

2.4 Sistema de Pruebas

Para llevar a cabo las pruebas, se plantearon escenarios específicos que son de interés para el desarrollo de los Sistemas de Tiempo Real, evaluando el desempeño y validándolo, en algunos casos, a través del tiempo de respuesta de ambos SOE.

Para obtener los tiempos en ambos SOE, se utilizó la misma estrategia. Esta consiste en la consulta y modificación del registro Cycle Count Register del procesador [11].

2.4.1 Latencia en Interrupciones

En este punto, se procedió a medir el tiempo de respuesta de cada uno de los SOE frente a la aparición de una interrupción externa.

Para ello, se utilizaron dos pines de la placa de desarrollo, un pin de salida que genera la interrupción y un pin de entrada que la captura. El procedimiento correspondiente se detalla en el siguiente esquema:

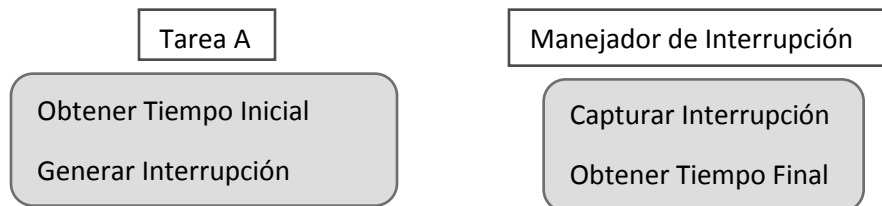


Fig. 1: Esquema de medición de la latencia de interrupción

Luego de medir los tiempos iniciales y finales, se realiza la resta de los mismos para obtener la latencia de interrupción. Se realizó este proceso de manera repetida para obtener un valor promedio de respuesta.

2.4.2 Inversión de prioridades

El caso de inversión de prioridades [10] es un problema muy común en la ejecución de tareas de un sistema. Ocurre cuando dos tareas de distinta prioridad utilizan un mismo recurso y la tarea de menor prioridad lo bloquea para poder utilizarlo de manera consistente. Luego, si una o varias tareas de mayor prioridad adquieren el procesador, el recurso quedara bloqueado hasta que la tarea de menor prioridad lo libere. Por tanto, si durante ese proceso se ejecuta la tarea de mayor prioridad que necesita el recurso, la misma quedará bloqueada a la espera de la liberación del mismo. Si en este lapso una tarea de prioridad intermedia se apropia del procesador, el bloqueo puede extenderse a un tiempo indeterminado.

Para la resolución de este problema, FreeOSEK ofrece la estrategia de Techo de Prioridad (Priority Ceiling), que consiste en darle a la tarea que utiliza el recurso el valor máximo de prioridad para evitar que le quiten el procesador y pueda ejecutarse hasta liberar el recurso. Por otro lado, FreeRTOS implementa Herencia de Prioridad (Priority Inheritance), donde la tarea de baja prioridad hereda el mayor valor de prioridad de las tareas en espera de ese recurso.

2.4.3 Manejo de tareas

Se realizaron pruebas básicas de manejo de tareas, creación, destrucción y suspensión en cada uno de los SOE, evaluando los tiempos de transición entre diferentes estados de las mismas, incluyendo el tiempo de intervención del planificador.

3 Resultados

A modo de comparación, se detallan, en cada una de las siguientes tablas, los tiempos obtenidos en la experimentación.

3.1 Interrupciones

Como muestra la Tabla 1, FreeOSEK permite la utilización de dos tipos de interrupciones. Por un lado, las de categoría 1, sin intervención del SOE, y las de categoría 2 en las cuales el vector de interrupciones llama a un manejador definido dentro del SOE.

En FreeRTOS ninguna función del sistema operativo tiene intervención en el manejo de las interrupciones.

Interrupción	Tiempo
FreeOSEK - Categoría 2	7,65 μ s
FreeOSEK - Categoría 1	0,90 μ s
FreeRTOS	0,50 μ s

Tabla 1: Tiempos latencia de interrupción

Se observa, de los tiempos de respuesta que se muestran en la tabla anterior, que los valores se encuentran en el orden de los microsegundos en todos los casos.

Luego, en el caso de las interrupciones de tipo 2 en FreeOSEK, producen que la intervención del SOE introduzca un overhead adicional por la ejecución del planificador.

Por otra parte, cuando el evento es directamente manejado por el vector de interrupciones del procesador (NVIC), los tiempos disminuyen un orden respecto a cuándo se requiere la intervención del planificador.

3.2 Manejo de Tareas

La Tabla 2 muestra los diferentes tiempos que se obtuvieron en el manejo de tareas en cada SOE.

Cabe aclarar que, la asignación de memoria para cada tarea no se tiene en cuenta en estas mediciones. FreeOSEK, al ser estático, realiza este procedimiento en tiempo de compilación. Por esta razón, este tiempo no influye en el funcionamiento del sistema.

En cambio, en FreeRTOS, la declaración de las tareas se realiza durante la ejecución del sector principal del programa, antes del inicio del planificador y de la ejecución de las tareas.

FreeRTOS, al ser dinámico, proporciona la posibilidad de crear tareas en tiempo de ejecución. Es en este momento cuándo se realiza la asignación de memoria de todos los recursos que se definen para dicha tarea. Para este caso, se realizó una medición de la creación de la tarea y su puesta en marcha por el planificador, resultando en un valor de 22 μ s.

-	Tiempo - FreeOSEK	Tiempo - FreeRTOS
Cambio de contexto	4,55 μ s	5,55 μ s
Suspensión de Tarea	4,25 μ s	4,10 μ s
Suspensión - Activación	5,10 μ s	4,70 μ s

Tabla 2: Tiempos de manejo de tareas

En la tabla anterior se observa que los tiempos de respuesta en el manejo de tareas se encuentran en el orden de los microsegundos, como exigen los Sistemas de Tiempo Real. Además, en ambos SOE, los valores para cada escenario son similares.

Los tiempos relacionados a la suspensión de tareas son mínimamente inferiores en el caso de FreeRTOS. Por otro lado, en el caso del cambio de contexto, donde una de las tareas termina y la otra pasa al estado de ejecución, existe una breve diferencia a favor de FreeOSEK.

4 Conclusión

En este trabajo se presenta una comparación de dos Sistemas Operativos Embebidos fundamentalmente seleccionados por sus características respecto a la manera en que se realiza la alocaación de recursos y tareas.

Los valores obtenidos comprueban que ambos son aptos para controlar un Sistema de Tiempo Real, ya que poseen tiempos de respuesta del orden de los microsegundos, independientemente de que el tipo de SOE sea dinámico o estático.

A la hora de seleccionar un tipo de SOE, los tiempos de respuesta no son determinantes. En cambio, el tamaño de memoria disponible es un factor que influye directamente en la elección. Esta característica juega a favor de los sistemas dinámicos, debido a que la cantidad de tareas que pueden ser creadas durante el ciclo de vida del sistema, no está limitada. En cambio, en los sistemas estáticos, el número de tareas se encuentra restringido por el tamaño de la memoria del sistema embebido. Por otro lado, la fiabilidad del sistema es una ventaja de los sistemas estáticos ya que las fallas por falta de memoria se verifican en tiempo de compilación.

Se plantea como trabajo futuro realizar pruebas sobre la nueva versión de FreeRTOS (9.0.0) la cual da soporte a tareas y recursos estáticos. Además, sería interesante evaluar los mismos parámetros frente a una mayor sobrecarga del sistema, por ejemplo, a través del incremento de la cantidad de tareas.

5 Bibliografía

1. N.C. Audsley , A. Burns , M. F. Richardson , A. J. Wellings: Hard Real-Time Scheduling: The Deadline-Monotonic Approach. Proc. IEEE Workshop on Real-Time Operating Systems and Software (1991).
2. L. Buhr. "An Introduction to Real Time Systems". Prentice Hall (1999).
3. A. Burns, A. J. Wellings: Designing hard real-time systems. Proceedings of the 11th Ada-Europe international conference on Ada. ISBN:0-387-55585-4 (1992).

4. A. Burns & A. Wellings: Real-Time Systems and Programming Languages. Addison Wesley, ISBN 90-201-40365-x.
5. R. Barry. Using the FreeRTOS Real Time Kernel.
6. C. Becker. RTOS en sistemas embebidos, disponible en http://iee.eie.fceia.unr.edu.ar/PDF_RTOS.pdf
7. A. Celery. Sistemas Operativos de Tiempo Real, disponible en http://www.sase.com.ar/2011/files/2010/11/SASE2011Introduccion_RTOS.pdf
8. FreeRTOS, disponible en <http://en.wikipedia.org/wiki/FreeRTOS>
9. FreeRTOS, disponible en <http://www.freertos.org/>
10. L. Sha, R. Rajkumar, and J. P. Lehoczky. Priority inheritance protocols: An approach to real-time synchronization. IEEE Trans. Comput., 39(9):1175{1185, September 1990.
11. Cortex-M4 Technical Reference Manual, disponible en <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi0439b/BABJFFGJ.html>
12. Introducción a OSEK-OS, El Sistema Operativo del CIAA-Firmware, disponible en http://proyecto_ciaa.com.ar/devwiki/lib/exe/fetch.php?media=desarrollo:firmware:rtos:introduccion_a_osek-os.pdf
13. Documentación EDU-CIAA-NXP, disponible en <http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp>

Prototipo funcional de un sistema de detección de caídas basado en la plataforma CIAA

Matías Dell'Oso¹, Laura Lanzarini¹, Pablo Ridolfi²

¹Instituto de Investigación en Informática LIDI (III-LIDI), Facultad de Informática,
Universidad Nacional de La Plata

²Laboratorio de Procesamiento Digital, Departamento de Ingeniería Electrónica, Facultad
Regional Buenos Aires, Universidad Tecnológica Nacional
Argentina
{mdelloso, laural}@lidi.info.unlp.edu.ar
{pridolfi}@frba.utn.edu.ar

Resumen. Las caídas son la principal causa de lesiones en personas mayores de 65 años. En este trabajo se describe el prototipo de un dispositivo capaz de detectar la caída de una persona y dar alerta a un centro de monitoreo y/o familiares. Si bien en la actualidad pueden encontrarse en el mercado soluciones similares, las mismas no son de fabricación nacional. El dispositivo descrito en este artículo permitirá contar con un detector de caídas de bajo costo de suma utilidad tanto para adultos mayores como para personas con movilidad reducida. Con él, se busca dar solución a las consecuencias adversas que se presentan al no detectar una caída rápidamente a través de un aparato totalmente desarrollado en Argentina.

Palabras clave: Detección de caídas, Teleasistencia, Proyecto CIAA, Tiempo Real, Sistemas Embebidos.

1. Introducción

El 30% de las personas mayores de 65 años sufre una caída cada año y dicho número aumenta de manera constante debido a que la población de adultos mayores crece anualmente [1-3]. Un informe de la OMS (Organización Mundial de la Salud) en 2012, señala que anualmente se producen 37,3 millones de caídas que requieren atención médica, de ellas 424 mil derivan en muerte, lo que convierte a las caídas en la segunda causa mundial de muerte por lesiones no intencionales, luego de los traumatismos causados por accidentes de tránsito [4].

Algunos estudios [5-8] demuestran que esta situación se presenta de forma similar en diversas partes del mundo. En España, por ejemplo, un estudio realizado por la Fundación MAPFRE mostró que el 14,7% de los adultos mayores de 65 años que viven solos, sufren por lo menos una caída anualmente [6]. El 63,4%, refieren haberse caído una única vez, un 11,3% dos veces, 6,7% tres veces, 0,8% cuatro veces y un 7,6% más de cuatro. En cuanto a los lugares donde se producen las caídas, el 45% se cayeron fuera de su domicilio mientras que el 39,9% lo hizo en su propio domicilio.

En Estados Unidos, se estima que 1 de cada 3 adultos mayores sufre algún tipo de caída cada año. Esto se traduce a una caída cada 13 segundos y en una muerte cada 20 minutos [7] [8].

En Argentina, un artículo publicado por el CoKiBA (Colegio de Kinesiólogos de la Provincia de Buenos Aires) en 2014 [5], advierte que 1 de cada 5 adultos mayores de 65 años sufre al menos una caída al año, y más del 80% de los episodios ocurren en el ámbito doméstico. Además, da cuenta de que, en hospitales públicos de la Provincia de Buenos Aires, 2 de cada 5 adultos de más de 80 años han sufrido al menos una caída al año. En función de estas estadísticas, resulta imprescindible el desarrollo de sistemas que ayuden a la rápida detección de una caída, enviando una alarma al personal médico o a familiares, minimizando así las posibles consecuencias adversas.

Teniendo en cuenta lo expuesto anteriormente, en este artículo se presenta el diseño e implementación del prototipo de un dispositivo capaz de identificar si una persona ha sufrido una caída enviando una alerta en caso de que corresponda.

Este artículo está organizado de la siguiente forma: la sección 2 brinda una general del sistema, la sección 3 contiene los detalles de diseño del dispositivo, la sección 4 explica las pruebas realizadas y la sección 5 explica las conclusiones y líneas de trabajo futuras.

2. Descripción general del sistema

El detector de caídas ha sido diseñado para que la persona pueda llevarlo sujeto a la altura de la cintura enganchado en su ropa o un cinturón y está capacitado para medir periódicamente la aceleración de sus desplazamientos. Cuenta con un botón de pánico, que el usuario puede presionar en caso de necesitar asistencia e indicadores de estado que informan si la alerta ha sido enviada o no. Dos módulos de comunicación, WiFi y GSM, generan la redundancia necesaria para asegurar el envío de la alerta ante posibles fallos (falta de señal, desconexión de Internet, entre otros). Posee además una memoria EEPROM para guardar información de contacto (números telefónicos de familiares, servidor de la central, etc.), una batería y un acelerómetro con el cual se detectará la ocurrencia de una caída (ver Figura 1).

En cuanto al modo de funcionamiento, el dispositivo medirá periódicamente la aceleración del paciente. Si se detecta una caída (o si el usuario presiona el botón de pánico), se iniciará la secuencia de envío de datos.

En lo que respecta al software de monitoreo, se diseñó una interfaz web, en la cual se puede monitorear el estado del dispositivo que carga cada paciente. A su vez, si se produce una caída, se enviará un mensaje de texto a todos los números telefónicos que el paciente tenga asociado.

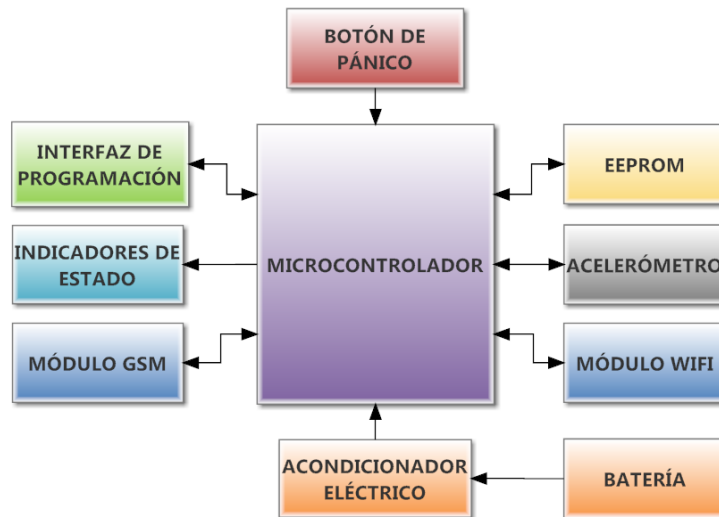


Figura 1: Esquema del dispositivo de detección de caídas

3. Diseño e Implementación

3.1. Diseño de hardware

Para poder detectar una caída es necesario que la persona cargue consigo un dispositivo que le permita enviar una señal de auxilio cuando sea necesario. A la hora de elegir cuál sería este dispositivo, se contemplaron dos opciones: utilizar un teléfono inteligente moderno, el cual cumpla con todas las características necesarias para detectar y alertar una caída (Procesador, conexión GSM y WiFi, y acelerómetro), o diseñar un sistema embebido. Finalmente, se decidió diseñar un sistema embebido por las siguientes razones:

- Los adultos mayores se resisten al uso de teléfonos celulares, por lo que no se podría garantizar que lo fueran a tener con ellos en todo momento.
- Los teléfonos celulares no poseen un sistema operativo de tiempo real, por lo que una tarea ajena a la detección de caídas podría retrasar la activación de una tarea crítica. Además, al contar con aplicaciones en segundo plano, el tiempo de vida de la batería se vería afectado significativamente.
- Un teléfono celular no podría cumplir con las normas y certificaciones necesarias para desarrollar un producto electrónico para el ámbito médico.

No obstante, no se descarta la posibilidad de desarrollar en un futuro una aplicación para dispositivos móviles, utilizando como base el algoritmo implementado en el sistema embebido, con el fin de realizar comparaciones de rendimiento o para ofrecerla como una aplicación complementaria.

A la hora de seleccionar el microcontrolador a utilizar, se buscó uno que perteneciera a la gama de bajo consumo, pero que al mismo tiempo tenga un poder de cómputo

adecuado para en un futuro procesar algoritmos complejos. Bajo dichas condiciones, se seleccionó el microcontrolador multinúcleo asimétrico LPC4337. La principal ventaja de utilizar un microcontrolador asimétrico es que, en principio, se puede ahorrar mucha energía utilizando el núcleo de baja potencia para las operaciones que no requieran una gran capacidad de cómputo, y activando el núcleo de alta potencia cuando el programa demande cálculos más complejos. Otra de las razones por la que se decidió a utilizar el LPC4337, es ubicar el trabajo en el marco del proyecto CIAA (Computadora Industrial Abierta Argentina) [9].

Cabe aclarar que, para el diseño de este prototipo, se utilizó la placa de desarrollo EDU-CIAA-NXP [10]. Además, tanto el acelerómetro como los módulos de comunicación utilizados son módulos independientes que contienen la electrónica básica necesaria para funcionar. Queda como trabajo futuro a la hora de desarrollar el producto final, integrar dichos módulos y el microcontrolador en un único circuito impreso.

Para conectar los módulos, los pulsadores e indicadores de estados y la memoria EEPROM, se diseñó una placa de expansión para la EDU-CIAA-NXP (ver Figura 2) la cual incluye: tres LEDs indicadores de estado, la memoria EEPROM, tres conectores para insertar los dos módulos de comunicación y el acelerómetro, una fuente de switching para alimentar la EDU-CIAA con voltajes superiores a 5 V, y un regulador lineal de 3,3V.

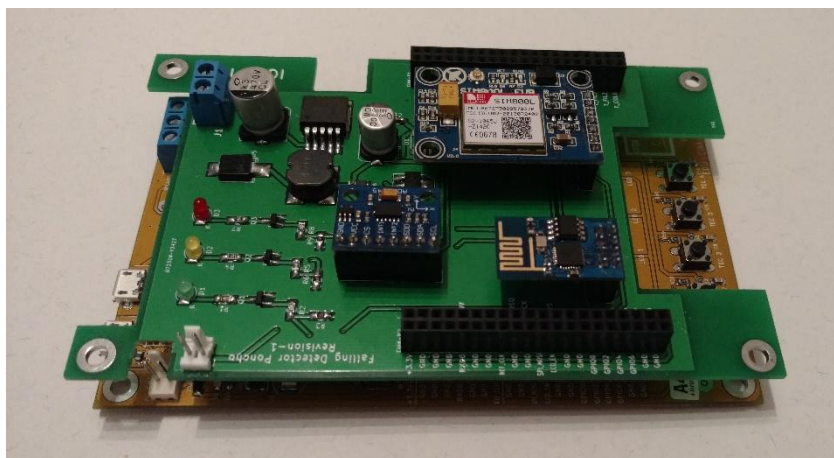


Figura 2: Placa de expansión conectada a la EDU-CIAA-NXP

3.2. Algoritmo de detección de caídas

Se desarrolló un algoritmo basado en umbrales el cual toma muestras del acelerómetro cada 8 ms (125 Hz), las procesa y, si se superan los valores de umbral, se iniciará el proceso de envío de alertas. A continuación, se detallará el algoritmo utilizado:

En primer lugar, se determinó qué umbrales se establecerán para la detección de una caída. Para ello, basándose en [11] [12], se definió el primer umbral de 6g para el vector de magnitud de las aceleraciones en los tres ejes (VMA) calculada según (1.1).

$$VMA = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (1.1)$$

Donde a_x , a_y y a_z son respectivamente las aceleraciones en los ejes X, Y y Z, tomados como se muestra en la Figura 3.

El segundo umbral es de 2g para la suma de las aceleraciones en el plano XZ (VMP) calculada según (1.2). El plano Y no fue incluido debido a que, según pruebas realizadas, se detectaron grandes valores de aceleración en este eje mientras la persona corría o se sentaba abruptamente, las cuales hubiesen dado una señal de falsa alarma.

$$VMP = \sqrt{a_x^2 + a_z^2} \quad (1.2)$$

El tercer umbral es de $1,5 \frac{m}{s}$ para la velocidad (V_0) de la persona en el instante que toca el suelo, calculada según (1.3).

$$V_0 = V_1 + \int_{t_1}^{t_0} VMA(t) dt \quad (1.3)$$

Para realizar esta integración, se integra hacia atrás la aceleración medida en todos los ejes (VMA) desde t_1 (1500 ms luego del contacto inicial) hasta t_0 (momento en el que la persona toca el suelo), teniendo así los tiempos de integración. El valor de 1500 ms fue obtenido empíricamente, debido a que, en todas las caídas realizadas intencionalmente, luego de un segundo y medio, no se observaron cambios significativos en los valores de aceleración en ninguno de los ejes de coordenadas.

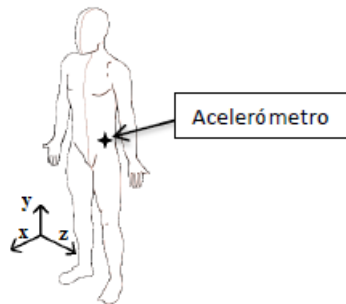


Figura 3: Localización del acelerómetro en el cuerpo de la persona

Para calcular el valor de la integral, se utilizó el método de integración numérica Simpson compuesto [13]. El valor de n (cantidad de sub-intervalos) fue establecido en 94, la mitad de la cantidad de muestras capturadas durante el período de integración (188 muestras). Con este valor se pretende alcanzar un punto medio entre la calidad

de la aproximación del cálculo y el cómputo que debe realizar el microcontrolador. Queda como trabajo futuro realizar pruebas variando el valor de n.

El sistema detectará una caída cuando se supere el primer umbral o cuando se superen el segundo y el tercero conjuntamente, tal como se puede observar en la Figura 3.

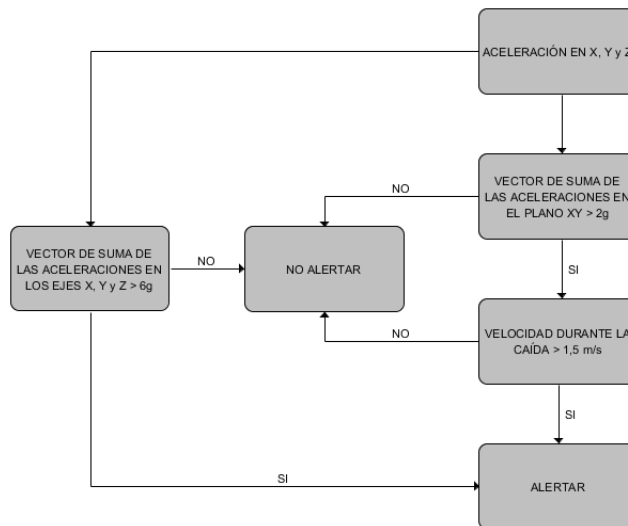


Figura 3: Diagrama de bloques del algoritmo utilizado para detectar una caída

4. Ensayos y Resultados

En las siguientes secciones se presentarán algunos de los ensayos realizados para verificar y validar el correcto funcionamiento de los módulos utilizados y del sistema en conjunto. Tanto las pruebas unitarias como las pruebas del sistema en conjunto, fueron realizadas utilizando la placa de expansión para la EDU-CIAA-NXP mencionada en la sección 3.1.

4.1 Pruebas unitarias sobre los módulos de comunicación y acelerómetro

Con el fin de comprobar que la resolución elegida para el acelerómetro ($\pm 8g$) es adecuada para captar los movimientos de una persona, se desarrolló una función en MATLAB que recibe por puerto serie los datos de la aceleración de los tres ejes y los grafica durante un período determinado. Para realizar esta prueba, se colocó el dispositivo en la cintura y se ejecutaron varias pruebas realizando movimientos bruscos (saltar, correr, sacudir el cuerpo, entre otras.) durante 8 segundos (1000 muestras). Como resultado, se observó que la aceleración de todos los ejes es, en todo momento, menor que 8g. La Figura 4 muestra el gráfico de la aceleración en los tres ejes mientras el sujeto de prueba realizaba saltos cortos y sacudía el cuerpo.

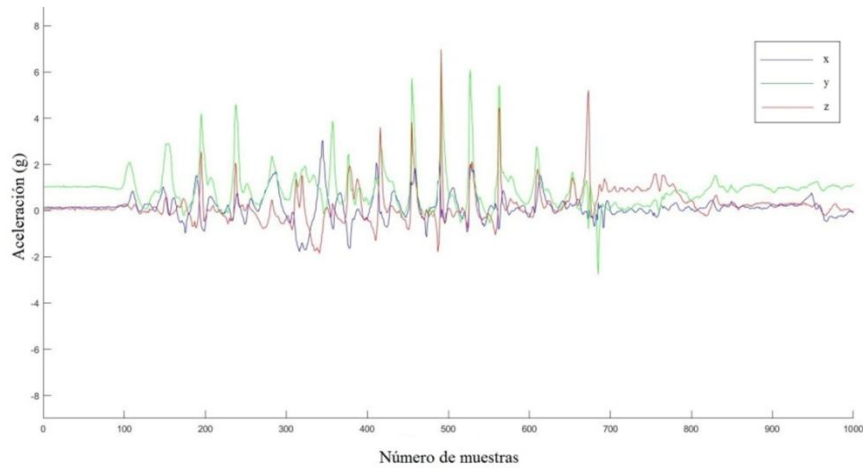


Figura 4: Monitoreo de la aceleración – Prueba de rango. Valores obtenidos por el acelerómetro al saltar y sacudir el cuerpo

Para comprobar que la comunicación mediante los módulos de comunicación se establece de manera correcta, se utilizó la UART para monitorear cada comando enviado a dichos módulos. En primer lugar, se desconectaron los módulos, se pulsó el botón de emergencia y, a continuación, se los volvió a conectar. La Figura 5 muestra los mensajes obtenidos para el módulo WiFi. Se puede observar que, aunque la comunicación falle, el dispositivo se recupera y, luego de un tiempo determinado, intenta establecer la comunicación nuevamente.

```

RealTerm: Serial Capture Program 2.0.0.70
Error - Module is not responding\r\n
Error - Module is not responding\r\n
Error - Module is not responding\r\n
Error - Module is not responding\r\n
Module is responding\r\n
Echo disabled\r\n
Operating mode was changed to: STATION\r\n
Module connected to AP = "Matt\r\n
IP - Station = 192.168.1.101\r\n
MAC - Station = 18:fe:34:fe:5e:1d\r\n
Multiple connections enabled\r\n
Connection Started:\r\n
Send you message now:\r\n
Message was sende\r\n

```

Figura 5: Correcta inicialización del módulo WiFi y envío de alerta exitoso

4.2. Pruebas del sistema en conjunto

Para comprobar que el sistema funciona de manera correcta, se plantearon 4 escenarios de la vida cotidiana y se observó el comportamiento del dispositivo en cada uno de ellos. Para ello, se utilizó la función de MATLAB mencionada anteriormente para monitorear la aceleración medida por el dispositivo en cada

escenario y, en los casos que se detectó una caída, se corroboró que la alerta haya sido recibida tanto en la página web, como en el teléfono celular asociado. En todos los casos, el dispositivo detector de caídas fue colocado en la cintura del sujeto de pruebas. A continuación, se detallan los escenarios planteados:

- i. **Una persona tropieza, cae al piso y permanece tendida en el suelo.** Esta prueba consistió en dejarse caer sobre una colchoneta y observar la aceleración en el transcurso de la caída. En la Figura 6 se puede observar que, en el momento de la caída, el vector magnitud de la aceleración supera el umbral establecido (6g) detectándose así, una caída.

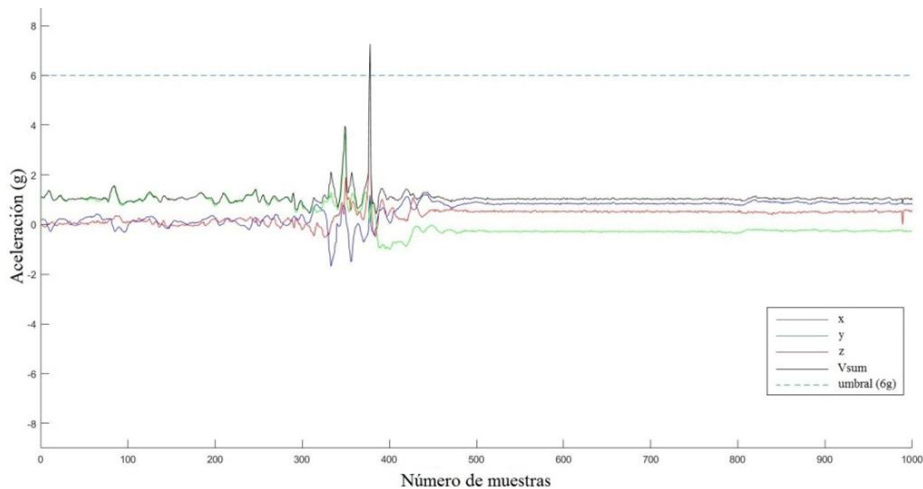


Figura 6: Monitoreo de la aceleración. Valores obtenidos por el acelerómetro al producirse una caída desde una gran altura

- ii. **Una persona sube y baja escalones de una escalera.** Para este escenario, ninguno de los umbrales fue superado, por lo que no se ha detectado ninguna caída.
- iii. **Una persona se descompensa y, mientras está cayendo, se agarra de un objeto para no golpear contra el piso.** En este caso el dispositivo no pudo detectar la caída. Debido a que el sujeto de pruebas se agarró de un objeto mientras caía, su aceleración disminuyó y el impacto contra el piso no fue lo suficientemente fuerte como para superar los umbrales establecidos. En este escenario, el algoritmo basado en umbrales falla. Es por ello que es de suma importancia implementar, en un futuro, un algoritmo inteligente que pueda detectar este tipo de caídas. De todas maneras, si el paciente sigue consciente luego de producirse la caída, será capaz de pulsar el botón de alerta y enviar la señal de auxilio.
- iv. **Una persona camina hacia una silla, se sienta abruptamente y luego cae al piso hacia su derecha.** Para este escenario el umbral de 6g no es alcanzado, pero el dispositivo detecta la caída debido a que supera el umbral de la velocidad con la cual la persona impacta contra el piso.

5. Conclusiones y trabajo futuro

Se ha presentado el diseño e implementación de un prototipo para la detección de caídas basado en la plataforma CIAA. Con base en las pruebas realizadas, se considera que su desempeño en lo referido a identificaciones y alertas ha sido satisfactorio.

Este es el primer paso en una dirección sumamente prometedora tanto en lo que respecta a la formación de recursos humanos como a la generación de mano de obra calificada argentina y a la asistencia social. No se incluyen comparaciones con otras soluciones existentes [14] [15] por considerar que el aporte de este trabajo es llegar a desarrollar el aparato propuesto con tecnología nacional y de bajo costo; aspectos que condicionan la capacidad del dispositivo.

Actualmente se está trabajando en la generación de una base de datos de caídas no sólo para dar mayor fortaleza a la selección de los umbrales utilizados por el reconocedor actual sino para construir un modelo no lineal adaptativo que permita mejorar la tasa de acierto del dispositivo a través del reconocimiento de ciertos patrones de movimiento propios del individuo.

En lo que respecta al hardware aún falta diseñar la carcasa la cual deberá ser resistente al agua para permitir el uso del dispositivo bajo la ducha. Finalmente se buscará separar los módulos de adquisición de datos y procesamiento de los módulos de comunicación GSM y WiFi con el fin de reducir el consumo y prolongar la vida útil de la batería.

Bibliografía

- [1] Deandrea, S., Lucenteforte, E., Bravi, F., Foschi, R., La Vecchia, C., Negri, E.: *Risk factors for falls in community-dwelling older people: a systematic review and meta-analysis*. 2010; 658–668
- [2] Major Injury Hospitalizations Due to Unintentional Falls in Canada 2009–2010. Report.
- [3] Peeters GM, Pluijm SM, van Schoor NM, Elders PJ, Bouter LM, Lips P. *Validation of the LASA fall risk profile for recurrent falling in older recent fallers*. J Clin Epidemiol 2010;63-1242.
- [4] Organización Mundial de la Salud (2016, Apr 4). Centro de prensa [Online]. 2012. Disponible en: <http://www.who.int/mediacentre/factsheets/fs344/es/>
- [5] Colegio de Kinesiólogos de la Provincia de Buenos Aires (2016, Jun 14). Centro de prensa. [Online]. Disponible en: <http://www.cokiba.org.ar/web/?q=node/116>
- [6] Sáinz M. *Estudio de investigación sobre Seguridad en el domicilio de personas mayores* (2016, Jun 27). Fundación MAPFRE [Online]. 2008. Disponible en: <http://www.mapfre.com/documentacion/publico/i18n/consulta/registro.cmd?id=128697>
- [7] Stevens J. A., Mack K. A., Paulozzi L. J., Ballesteros M. F. *Self-Reported Falls and Fall-Related Injuries Among Persons Aged >65 Years*. Morbidity and Mortality Weekly Report. Vol. 57. No. 9. 2008.
- [8] Centers for Disease Control and Prevention (2016, Apr 2). *Important Facts About Falls* [Online]. Disponible en: <http://www.cdc.gov/homeandrecreationalsafety/falls/adultfalls.html>
- [9] Proyecto CIAA (2016, Jul 5). Computadora Industrial Abierta Argentina [Online]. Disponible en: <http://www.proyecto-ciaa.com.ar/devwiki/doku.php>

- [10] Proyecto CIAA (2016, Jul 7). EDU-CIAA-NXP [Online]. Disponible en: <http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp>
- [11] Perry J.T., Kellog S., Vaidva S. M., Jong-Hoon Y., Hesham A. Sharif H. *Survey and evaluation of real-time fall detection approaches*. 6th International Symposium on High-Capacity Optical Networks and Enabling Technologies (HONET), 2009.
- [12] Lindemann U. *Evaluation of a fall detector based on accelerometers: a pilot study*. *Medical & Biological Engineering & Computing*. Vol. 43. No 5.2005.
- [13] Simpson's Rule (2016, Apr 7). Wolfram MathWorld [Online]. Disponible en: <http://mathworld.wolfram.com/SimpsonsRule.html>
- [14] Bagalà, F., Becker, C., Cappello, A., Chiari, L., Aminian, K. *Evaluation of Accelerometer-Based Fall Detection Algorithms on Real-World Falls*. 2012.
- [15] Rodriguez, J., Mercuri, M., Karsmakers, P., Soh, P.J., Leroux, P., Schreurs, D. *Automatic Fall Detector based on Sliding Window Principle*. WIC 2013.

Arquitectura e Implementación de Equipamiento Prototipo de Bajo Costo para la Medición del Diámetro de Fibras de Cabra en Campo

Rafael Zurita, Miriam Lechner, Rodolfo del Castillo, and Eduardo Aisen

Facultad de Informática y Facultad de Agronomía,
Universidad Nacional del Comahue
Buenos Aires 1400, Neuquén, Argentina
{rafa,mtl,rdc,eduardo.aisen}@fi.uncoma.edu.ar
<http://faiweb.uncoma.edu.ar>

Resumen La cría de cabras es una actividad económica habitual en el norte de la provincia de Neuquén. La falta de tecnología adecuada reduce las ganancias que el criancero puede obtener de la misma.

En este artículo se presenta la arquitectura de hardware y software de un prototipo real, diseñado y desarrollado para la clasificación de fibras textiles; con el objetivo de ser utilizado por el criancero directamente en el campo.

Desde la perspectiva del hardware, se evidencia la necesidad de un sistema embebido. Lo cual apunta a bajos costos de producción, robustez y movilidad. Por otro lado, desde el software, se utiliza un algoritmo con orden de tiempo de ejecución lineal, como base del procesamiento de imágenes.

Se ha evaluado el prototipo realizando mediciones de diámetro de fibras animal, y comparando los valores obtenidos contra un instrumento comercial de similares características.

El proceso de validación muestra una alta correlación entre los resultados obtenidos, así como también con los valores obtenidos al procesar imágenes de fibras artificiales. Basado en esta experiencia, y el análisis de costo/beneficio, se muestra que es posible la producción de una solución final para el criancero.

Palabras clave: sistema embebido, procesamiento de imágenes, fibra textil.

1. Introducción

Los pequeños productores en el sistema agroecológico del norte de la Provincia de Neuquén son los crianceros. A nivel local la palabra criancero denomina a un amplio conjunto de productores familiares en el que predominan productores campesinos o con rasgos campesinos y que se dedican fundamentalmente a la cría de caprinos y ovinos [1]. Viven de la producción de fibras textiles de origen animal (principalmente caprino), y no disponen de los medios técnicos y el asesoramiento profesional requerido para: i) orientar la selección animal a la

producción de vellones mas pesados y calidades de fibra de valor superior; ii) establecer objetivamente los parámetros de la calidad de la fibra, de manera de poder clasificarla para optimizar la comercialización; iii) negociar con el comprador el precio de mercado que la calidad de la fibra amerita; y iv) orientar la producción ganadera hacia aquellos biotipos mejor adaptados a la naturaleza de la región, para así poder frenar y eventualmente revertir el proceso de desertificación que las afecta [2].

Por otro lado, existen en el mercado varios equipos profesionales para la determinación de finura (desarrollados principalmente para lana) como el OFDA 2000, WoolView 20/20, LaserScan, air flow, y lanámetro, entre otros [4], [5]. De cualquier manera, la mayoría de estos equipamientos son costosos y destinados a ambientes de laboratorio, por lo que no son indicados para utilización sobre el animal [6]. WoolView 20/20 fue diseñado para trabajo en campo, pero el producto está discontinuado [7]. En este artículo se presenta la arquitectura de hardware y software de un equipamiento prototipo desarrollado en la Universidad Nacional del Comahue (por lo que se lo ha denominado “prototipo UNCOMA”) para la medición de diámetro de fibra sobre el animal en campo. Considerando al criancero como destinatario del producto final, y no un entorno de laboratorio. En tal sentido, se persiguen al menos dos objetivos que resulten en la factibilidad de uso: entrenamiento previo mínimo y, preparación sencilla de la muestra a analizar.

Los resultados muestran que el costo, performance y precisión del prototipo UNCOMA propuesto, es adecuado para la selección objetiva en origen de animales de alta producción de fibra (especialmente cashmere), lo que permitiría aumentar la producción por cabeza e incrementar los ingresos anuales del productor campesino. De este modo se podrán hacer efectivas las propuestas de manejo de la carga animal, contribuyendo así a revertir el grave problema de la desertificación.

El resto de este trabajo se estructura de la siguiente manera. En la sección 2 se describen las arquitecturas de software y hardware del equipamiento propuesto, con un énfasis en el método de medición por análisis de imagen digital. En la sección 3 presentan varios resultados experimentales, para validar el proceso de determinación de finura llevado a cabo por el prototipo UNCOMA. Por último, en la sección 4 se presentan las conclusiones y líneas de trabajo a futuro.

2. Arquitectura

2.1. Arquitectura de Hardware

La arquitectura del hardware presenta dos componentes principales :

- El dispositivo de mano portátil;
- El hardware de procesamiento embebido

El dispositivo de mano portátil contiene un microscopio digital, un led de contraste, una prensa, y un mecanismo de gatillo. El microscopio es de 400

ampliaciones, modificado y calibrado. Funciona con una tensión de 5 voltios, y mecánicamente se encuentra empotrado en el dispositivo, como se observa en Figura 1.

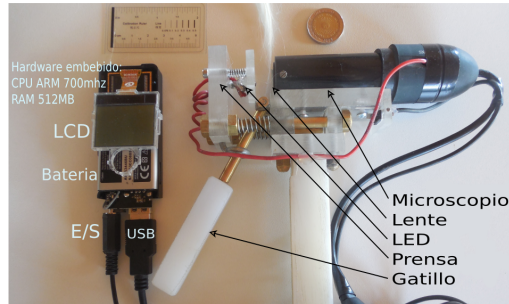


Figura 1. Equipamiento prototipo UNCOMA

El mecanismo de gatillo realiza varias tareas en un único paso. Al gatillar, el operador prensa el acrílico opuesto a la óptica del microscopio, lo que produce que las fibras sean enfocadas correctamente y queden firmes. Además, al final del accionar del gatillo, se activa un interruptor digital. Este interruptor envía una señal al hardware embebido, a través de un puerto de E/S digital que dispara un evento en el sistema y activa el software de obtención de imágenes. En ese momento, el software en el hardware embebido solicita una captura de imagen digital de las fibras que el microscopio tiene enfocadas y estancadas, para luego ser procesada. La imagen digital es obtenida desde el dispositivo portátil a través de una interfaz USB 2.0, la cual interconecta ambos componentes (el dispositivo de mano y el hardware embebido).

El hardware embebido es el sistema en donde se ejecuta el software que analiza la imagen digital y reporta los resultados. Contiene un System On Chip (SOC) principal Allwinner A10, con una CPU ARM de 700Mhz. La memoria principal (RAM) es de 512MB, y también cuenta con una interfaz de memoria SD en donde se ha incorporado una tarjeta microSD de 2GB. El bootloader del SOC carga, al momento de reset, el sistema operativo desde la memoria microSD, y luego el sistema operativo ejecuta la aplicación prototipo final. El SOC Allwinner también cuenta, internamente, con dispositivos de E/S (GPIO), un puerto USB 2.0, y una interfaz Wireless WiFi 802.11b/g/n.

El tamaño del hardware embebido es de 10x4x3cm, y el del dispositivo portátil es de 20x20x5cm. El peso total de todo el equipamiento es de 900gr. En cuanto al costo del equipamiento es necesario enumerar los componentes críticos de mayor precio:

1. Microscopio de 400x incorporado en el dispositivo portátil : USD 60.
2. Placa electrónica embebida, que contenga un core CPU ARM de 700Mhz+, y 256MB de RAM: USD 30.
3. Piezas mecánicas elaboradas a medida: USD 20.

2.2. Arquitectura de Software

A bajo nivel, el hardware embebido está controlado por el sistema operativo Debian. Particularmente, los principales drivers que se utilizan del kernel Linux son el universal video class (UVC) y la interfaz de E/S de puertos generales (GPIO) del núcleo. A través de uno de los puertos de entrada digitales (GPIO) se recibe la señal digital disparada por el dispositivo portátil. En este punto, la aplicación en espacio de usuario recibe una notificación del evento y realiza el proceso de captura y análisis digital de una imagen de fibras. El proceso general puede ser observado en la Figura 2.

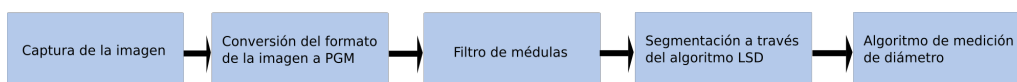


Figura 2. Etapas de todo el proceso, llevadas a cabo por el prototipo UNCOMA.

Para la captura de la imagen, la aplicación embebida utiliza las facilidades que el kernel Linux exporta al espacio de usuario a través del driver UVC. El driver UVC que controla la captura del microscopio incorporado en el dispositivo portátil obtiene una imagen completa de las fibras en la prensa, y se la presenta a la aplicación en espacio de usuario. El formato de la imagen digital recibida es JPEG, por lo que se procede, en primera instancia, a un proceso de conversión al formato de mapa de grises portable (PGM). Esto es llevado a cabo por el conjunto de herramientas de software netpbm, y es necesario debido a que la aplicación embebida requiere este formato como entrada del algoritmo *Line Segment Detector (LSD)*. La resolución de la imagen es de 640x480 píxeles, en donde 1 píxel corresponde a 0.9743 micrones de acuerdo a la óptica y microscopio seleccionados para el prototipo.

Cada fibra contiene una médula, eventualmente la imagen capturada puede hacerla visible lo cual afecta a las mediciones. Por tal motivo se ejecuta un filtro de médula para quitarlas parcialmente. El filtro se realiza calculando la transformada de distancia para píxeles que se encuentren dentro de un rango de grises. Dicho rango fue previamente establecido y representa los valores esperables para la médula. Si la detección es positiva, se reemplaza la zona medular por píxeles que contengan un color dentro del rango de grises coincidente con fibra, que también fue previamente establecido.

Posteriormente, la aplicación utiliza el algoritmo *Line Segment Detector (LSD)* para obtener un conjunto de segmentos de línea que representan a la imagen original [8]. A partir de este conjunto se realiza la medición de diámetros de fibras según el algoritmo a continuación:

```

# Pseudocódigo del algoritmo de cálculo de diámetro medio de fibras
# ENTRADA: Conjunto de segmentos de línea (lsd_set)
# SALIDA: Mediciones y diámetro medio (mediciones_set, diametro)
  
```

```

mediciones = 0;
diametro = 0;
[1] FOR each segmento IN lsd_set DO
    diametro_tentativo = 0;
    mediciones_t = 0;
[2]  FOR each punto (x,y) IN inicio_medio_final(segmento) DO
        f() = perpendicular(segmento, punto);
[3]    FOR each seg2 IN lsd_set DO
        IF paralelos(segmento, seg2) AND
            ( f() == perpendicular(seg2) ) AND
            ( f(DESDE segmento HASTA seg2) == pixels_de_fibra() ) THEN
[4]        diametro_tentativo = distancia( f(DESDE segmento HASTA seg2) );
            mediciones_t++;
        END IF
    END FOR
  END FOR
[5]  IF (mediciones_t >= 2) THEN
        diametro = diametro + diametro_tentativo;
        mediciones++;
        mediciones_set = add(diametro_tentativo);
    END FI
END FOR

diametro = diametro / mediciones;
RETURN mediciones_set, diametro

```

De manera general, el algoritmo busca segmentos paralelos intersectados por dos a tres perpendiculares. Interpretando esto como un segmento de fibra. Si los píxeles contenidos entre los segmentos, a lo largo de las perpendiculares, coinciden con el rango de color esperable para fibra, entonces se consideran dichas distancias como diámetros tentativos. Calculando por último el promedio de los valores obtenidos para cada segmento. Específicamente:

1. Por cada segmento de línea en el conjunto se calcula la ubicación de 3 puntos (x,y): principio medio y final.
2. Por cada punto se calcula la función de la recta perpendicular al segmento original.
3. Se recorren todos los segmentos restantes en búsqueda de aquellos que sean paralelos al segmento original, e intersectados a su vez por la perpendicular calculada en el paso anterior. Si el segmento perpendicular desde el punto origen hasta la intersección contiene puntos que no pertenecen al rango de color esperable para fibras, se descarta.
4. Si no se descarta, el segmento de línea paralelo encontrado es el borde opuesto al original de la fibra, y se procede a realizar el cálculo de distancia entre el punto original y la intersección. Esta medida es catalogada como diámetro tentativo.

5. Si al menos dos puntos obtuvieron un diámetro tentativo se establece la medición tentativa como válida, y pasa a formar parte del cálculo estadístico de media de diámetro general. Sino, el diámetro tentativo es catalogado como ruido y se descarta.

Finalmente, luego del procesamiento de segmentos de línea, se realiza el cálculo de media de diámetro general, la desviación estándar y la varianza. Los resultados son emitidos por la salida estándar, y de acuerdo a los valores, se muestra en el display LCD, del hardware embebido, la siguiente información :

- [1] diámetro medio;
- [2] porcentaje de medidas menores a 17um;
- [3] porcentaje de medidas entre 17um y 30um;
- [4] porcentaje de medidas mayores a 30um;

Toda la estadística es almacenada en el dispositivo embebido, y puede ser extraída y visualizada, a través de su conexión wireless. Adicionalmente, se ha desarrollado una interfaz gráfica (GUI) para visualizar el proceso de análisis en una PC, y corroborar fehacientemente (visualmente) que el análisis automático es correcto (que las mediciones realizada por el software se encuentren dentro de las fibras, y sean perpendiculares a sus extremos). Esta interfaz gráfica permite también utilizar el dispositivo portátil (su microscopio) desde una notebook o PC.

Un ejemplo de análisis utilizando la interfaz gráfica se muestra en la Figura 3. A la izquierda se observa la imagen original obtenida desde el microscopio. A la derecha una representación gráfica del resultado del algoritmo de medición. Los segmentos en color rojo son mediciones finales obtenidas por el algoritmo. Esta representación visual es útil para corroborar si hubieron mediciones erróneas del software (por ejemplo, fuera de las fibras).

3. Resultados

Para determinar la validez del prototipo UNCOMA se realizaron mediciones de tiempo y precisión.

El tiempo de ejecución total del prototipo es importante porque nos permite evaluar si el hardware y software son adecuados para uso cotidiano. Un dispositivo que tome un tiempo considerable en medir la finura de fibra, puede no ser indicado para un campo con mucho ganado, ya que se requeriría un mayor número de operarios y dispositivos, para realizar mediciones en paralelo. Lo cual sería costoso o incluso impracticable.

La precisión del resultado del análisis es la valuación más importante, ya que un dispositivo que realiza mediciones incorrectas o poco precisas no es confiable. Y por ende carente de valor frente al ojo experto del criancero.

3.1. Orden del Tiempo de Ejecución

El costo en tiempo de ejecución se ha observado al analizar 96 muestras de fibras reales (sin contar el tiempo del operario en preparar las fibras a medir

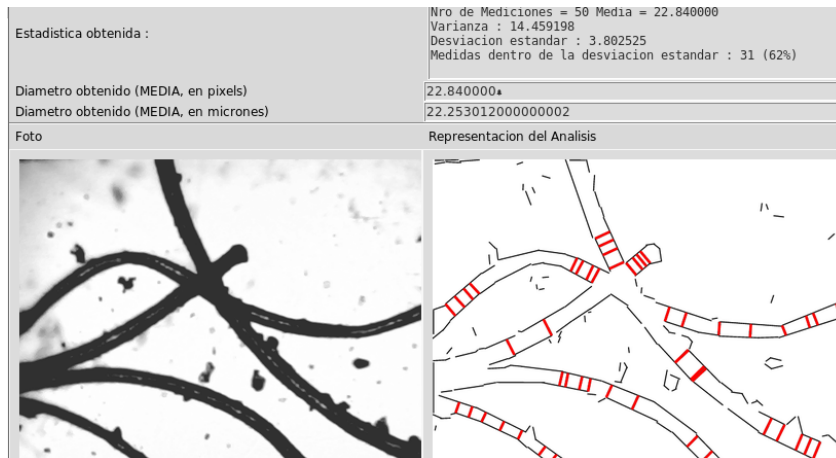


Figura 3. Resultado de la medición de diámetro presentado por la interfaz gráfica.

sobre el animal). El tiempo fue medido mediante la herramienta GNU time, el cual ejecuta cada etapa y evalúa los recursos utilizados. Se observa que tanto el proceso de obtención de la imagen, como la conversión de la misma es constante. No siendo así para el análisis de la imagen mediante el software propuesto, el cual varía en el rango de (0seg..3seg]. Durante dicho análisis, el mayor costo en tiempo de ejecución es el del algoritmo LSD, que es de orden de tiempo lineal [8]. En base a esto, y al hecho de que las fotos poseen la misma resolución y, que el número de de fibras analizadas en cada medición es menor a 15 (un mayor numero de fibras distorsionaría los resultados), el tiempo de ejecución de cada medición ha sido determinado, luego de analizar 96 muestras, en un rango (0seg..3seg].

3.2. Precisión del proceso de análisis

Para la obtención de resultados de precisión se analizan dos clases de mediciones realizadas:

- mediciones con imágenes de fibras artificiales que contienen medidas conocidas a priori, y
- mediciones con 96 imágenes de muestras de fibras reales, de un mismo mechón, utilizando el prototipo UNCOMA y el producto comercial Wool-View 20/20. Las muestras se obtuvieron en ambos equipamientos independientemente.

Medición de imágenes fibra artificiales Se han confeccionado 74 imágenes de fibras artificiales mediante el software de manipulación de imágenes GIMP. Las imágenes contienen entre 1 y 7 líneas negras (fibras artificiales) de orientación y sentidos aleatorios. Cada línea tiene un grosor conocido a priori y documentado

en toda su extensión. Por ejemplo, en la Figura 4 se observa, a la izquierda, una imagen con 5 fibras artificiales, cada una de un grosor definido en toda su extensión. Los diámetros seleccionados para las líneas se encuentran entre los 10 píxeles de grosor y los 40 píxeles. La media observada manualmente para cada imagen es calculada como la sumatoria de las medidas de cada línea en la imagen sobre la cantidad de líneas. En la Figura 4 se observan, a la derecha, los resultados de 35 de las 74 mediciones realizadas por el prototipo UNCOMA a las imágenes preparadas. Junto a cada barra de la media calculada por el software se muestra la media calculada a priori, en base a las medidas de las líneas dibujadas con el programa GIMP.

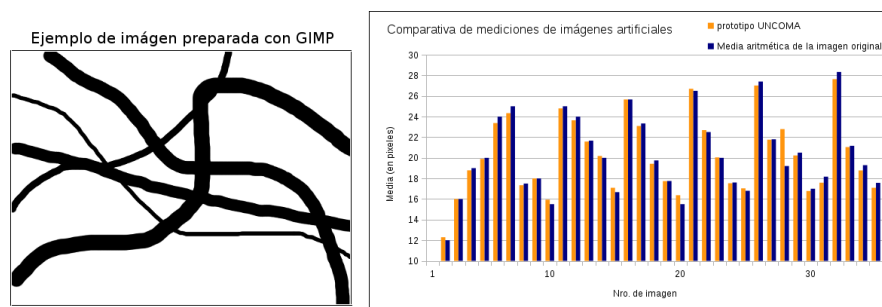


Figura 4. Izq: Una imagen artificial. Der: Mediciones realizadas a las imágenes.

Medición de imágenes de fibra reales Para el análisis del software propuesto se realizó una comparativa de medición entre el prototipo UNCOMA y el equipamiento WoolView 20/20. WoolView 20/20 es un dispositivo comercial para la medición de fibras de lana con precisión menor a un micrón, el cual puede analizar las muestras sin quitar las fibras del animal.

Con el prototipo UNCOMA se analizaron 96 muestras obtenidas de un mismo mechón de animal. Estas muestras fueron almacenadas para ser publicadas en conjunto con este trabajo. Se desarrolló un script por lotes que secuencialmente analiza cada una de estas muestras almacenadas, utilizando el software prototipo. La estadística de cada foto digital analizada es almacenada en un archivo separado. Tanto las muestras originales como el resultado de cada medición de diámetro a través del software del prototipo puede obtenerse de [3].

La medición utilizando el equipamiento WoolView 20/20 fue realizada por un operario diferente al que realizó la medición con el prototipo UNCOMA. Utilizando el mismo mechón de animal se realizaron 96 mediciones con WoolView 20/20. Este equipamiento acumula la estadística, por lo que se obtuvieron los detalles de las estadísticas intermedias cada 10 mediciones.

En la Figura 5 se observan los resultados de las mediciones intermedias de ambos equipamientos. El diámetro medio final obtenido por el prototipo propuesto

es de 20,6263 micrones (en la Figura 6 se observa el histograma normalizado de todas las mediciones). El diámetro medio obtenido con el equipamiento WoolView 20/20 es de 20,9 micrones. En la tabla (1) se observan las estadísticas finales de ambos equipamientos.

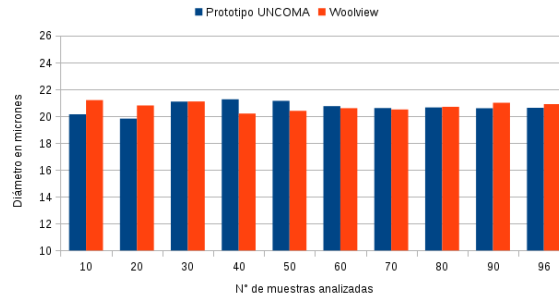


Figura 5. Comparativa acumulativa de 96 mediciones utilizando dos equipamientos : prototipo UNCOMA y WoolView 20/20. La diferencia de la media de diámetro entre ambos equipos va disminuyendo a medida que se realizan mayor cantidad de mediciones sobre el mismo mechón de fibras. Luego de 20 mediciones la diferencia es menor a un micrón. Luego de 50 mediciones la diferencia es menor a 0.5 micrones.

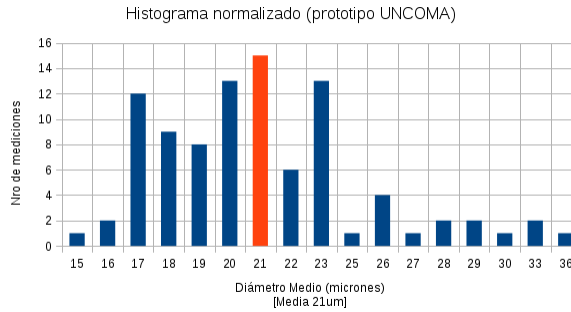


Figura 6. Histograma de las 96 mediciones realizadas con el prototipo UNCOMA.

4. Conclusiones y Trabajo Futuro

En este trabajo se presenta la arquitectura de un sistema prototipo funcional de bajo costo, para la determinación de finura de la fibra de cabra. Se ha validado la precisión del instrumento a partir de la medición del diámetro de imágenes de

	prototipo UNCOMA	WoolView 20/20
Nro. de imágenes analizadas	96	96
Diámetro Medio	20,6263	20,9
Coefficiente de confort	91,6 %	91 %
Desvío estándar	4,03um	5,5um
Coefficiente de variación	19,5 %	27 %
Muestras dentro del DE	82 %	No reportado

Tabla 1. Estadística global de ambos equipamientos luego de finalizar las mediciones.

fibras artificiales; y también de muestras de fibras reales. Comparando las últimas con las mediciones obtenidas por otro equipamiento profesional de similares características. Los resultados muestran que la diferencia entre ambos equipamientos es menor a un micrón cuando se realizan, al menos, diez mediciones. El peso y tamaño del equipamiento permiten al operador utilizar el equipamiento a mano alzada, tomando capturas de las muestras de fibra directamente sobre el animal, en el campo. La preparación de las fibras sobre el animal o en laboratorio es un proceso que se encuentra en el orden de los minutos, por lo que el tiempo observado para la medición de diámetro (en el orden de los segundos) no es significativo.

En adelante se pretende realizar la validación del prototipo UNCOMA, con respecto a resultados obtenidos por instrumentos de laboratorio como OFDA 2000. Por otra parte se investigan nuevas técnicas de procesamiento y filtrado de las imágenes, que reduzcan la cantidad de muestras descartadas debido a la falta de foco.

Referencias

1. Bendini, M.: Chapter 8. Transhumant Communities and Agroecosystems in Patagonia. Interactions Between Agroecosystems and Rural Communities. Cornelia Flora (2001). ISBN: 978-0-8493-0917-5
2. Bendini, M., Tsakoumagkos, P., Nogues, C.: Los crianceros trashumantes en Neuquén. Grupo de Estudios Sociales Agrarios (GESA), UNCOMA (2005) http://investigadores.uncoma.edu.ar/cehepyc/publicaciones/Los_trashumantes_en_Neuquen.pdf
3. Banco de imágenes de fibras experimentales UNCOMA, <http://se.fi.uncoma.edu.ar/fibraimg/>
4. OFDA 4000 instrument, http://www.ofda.com/Natural_fibres/Ofda4000.html
5. Sirolan Laserscan, <http://www.itecinnovation.com/productDetails.php?id=52>
6. Baxter, P.: Comparisons between OFDA, Airflow and Laserscan on raw merino wool. International Wool Textile Organization. 2002.
7. WoolView 20/20 portable device, https://web.archive.org/web/20041102185907/http://www.woolview.com/Product1_2.htm
8. Grompone, R., Jakubowicz, J., Morel, J., Randall, G.: LSD: a Line Segment Detector, Image Processing On Line, 2 (2012), pp. 35–55. <http://dx.doi.org/10.5201/ipol.2012.gjmr-lsd>



V Workshop de Seguridad Informática (WSI)

Comité Científico:

Ana Lucía Sandoval Orozco (España)
Antonio Ricardo Castro Lechtaler (Argentina)
Ariel Maiorano (Argentina)
Carlos Acquistapace (Argentina).
Carlos Benítez (Argentina)
Carlos García Garino (Argentina)
Edith García (Argentina)
Javier Díaz (Argentina)
Javier Echaiz (Argentina)
Javier García Villalba (España)
Jorge Ramió Aguirre (España)
Julio César Liporace (Argentina)
Lia Molinari (Argentina)
Liane Tarouco (Brasil)
Marcelo Cipriano (Argentina)
Paula Venosa (Argentina)
Raúl Saroka (Argentina)



V Workshop de Seguridad Informática (WSI)

ID	Trabajo	Autores	
9152	Improving a Compact Cypher Based on Non Commutative Rings of Quaternions	Jorge Kamlofsky (UAI)	1115
9162	Dispositivos móviles y el fenómeno del BYOD. Su impacto en la seguridad de las organizaciones	Nicolás Macia (UNLP), Carlos Damián Piazza Orlando (UNLP), Sebastian Exequiel Pacheco Veliz (UNLP)	1125
9215	Loss of Votes in NIDC Applying Storage in Parallel Channels	Germán Montejano (PUC Rio), Pablo García (UNLPAM), Silvia Bast (UNLPAM), Estela Fritz (UNLPAM)	1135
9236	Securizando la comunicación de un repositorio digital de contenido académico con Moodle	F. Javier Díaz (UNLP), Alejandra Schiavoni (UNLP), Ana Paola Amadeo (UNLP), Duilio Ray (UNLP)	1145
9273	Procedimiento para la detección empírica de Infraestructuras de Clave Pública anómalas o inseguras	Antonio Castro Letchtaler (UNDEF), Marcelo Cipriano (UNDEF), Eduardo Malvacio (UNDEF)	1156
9299	Un enfoque para la detección de anomalías en el tráfico de red usando imágenes y técnicas de Computación de Alto Desempeño	Mercedes Barrionuevo (UNSL), Mariela Lopresti (UNSL), Natalia Miranda (UNSL), María Fabiana Piccoli (UNSL)	1166
9326	Modelo de Integración de estándares para la Gestión de identidad y Privacidad	María del Carmen Becerra(UNSJ), Pedro Zarate(UNSJ), María Claudia Gomez(UNSJ)	1176

Improving a Compact Cipher Based on Non Commutative Rings of Quaternions

Jorge Alejandro Kamlofsky

¹ CAETI - Universidad Abierta Interamericana
Av. Montes de Oca 725 – Buenos Aires – Argentina
Jorge.Kamlofsky@uai.edu.ar

Abstract. Asymmetric cryptography is required to start encrypted communications. Most protocols are based on modular operations over integer's rings. Many are vulnerable to sub-exponential attacks or by using a quantum computer. Cryptography based on non-commutative algebra is a growing trend arising as a solid choice that strengthens these protocols. In particular, Hecht (2009) has presented a key exchange model based on the Diffie-Hellman protocol using matrices of order four with elements in Z_{256} , that provides 128-bits keys also to devices with low computing power. Quaternions are four-component's vectors. These also form non-commutative rings structures, with compact notation and lower run-times in many comparable operations. Kamlofsky et al (2015) presented a model using quaternions with elements in Z_{256} . To provide a 128-bit key is required 4 rounds of 32-bits. However, a gain of 42% was obtained. This paper presents an improvement of this cipher that reduces even more the run-times.

Keywords: Asymmetric cryptography, quaternion's cipher, non-commutative cryptography, post-quantum cryptography.

1 Introducción

1.1 Trabajos Relacionados

La Criptografía es una rama de la Matemática. Trata el problema de enviar información confidencial por un medio inseguro. Para ello, se cifra la información de manera que, aun cuando se encuentre disponible para cualquiera, no pueda ser utilizada, a menos que alguien autorizado la descifre. En una comunicación cifrada, entonces, pueden presentarse dos instancias diferentes: el intercambio seguro de claves y luego, con ello, el cifrado y descifrado del mensaje [1]. La Criptografía se

divide en dos grandes ramas: de clave privada o simétrica, que cifra y descifra los mensajes y de clave pública o asimétrica, que logra el intercambio seguro de claves.

Diffie y Hellman fueron los pioneros de la criptografía asimétrica: en 1976, en [2] presentaron el revolucionario concepto de criptografía de clave pública cuya seguridad radica en el problema de la intratabilidad del logaritmo discreto [3] (*DLP: Discrete Logarithm Problem*). Sin embargo, por facilidad de implementación práctica, el esquema criptográfico de clave pública hoy más usado es RSA [4]: su seguridad radica en el problema de la intratabilidad de la factorización de grandes números enteros (*IFP: Integer Factorization Problem*).

En 1993 Peter Shor presentó en [5] un algoritmo que reduce la complejidad computacional del problema IFP mediante una computadora cuántica. A pesar que este dispositivo aún no se había inventado, solo la existencia del algoritmo, debilitó a esta rama de la criptografía. Hoy su existencia es un hecho: la empresa D-Wave Systems ya vendió computadoras cuánticas a Lockheed Martin, al laboratorio Los Alamos, a Google y a la NASA, entre otros [6]. Además, IBM por su lado, ofrece servicios en la nube con su computadora cuántica [7].

Desde inicios de este siglo ha crecido el interés por el desarrollo de cripto-sistemas asimétricos alternativos que sean resistentes a ataques de complejidad sub-exponencial y ataques vía computadora cuántica [8 – 9]. A la mayoría de estos esquemas se los denomina colectivamente como criptografía post-cuántica [10] o bien, por su naturaleza algebraica, se los denomina criptografía no conmutativa [11]. Sobre esta línea, no se conocen ataques que hayan logrado resultados concretos. Dentro de esta línea, en [12] se presentó un esquema de distribución de claves Diffie-Hellman basado en un anillo de polinomios matriciales. Al sistema se lo denominó compacto debido a que no se requieren librerías de precisión extendida, lo que hace posible su uso en procesadores de menor porte. En [13] se implementó dicho esquema en un anillo de polinomios de cuaterniones, lo cual permitió la obtención de claves de la misma longitud (128 bits) con una mejora de 42,59% en los tiempos de ejecución.

En este trabajo, se presentan resultados que muestran mejoras aún mucho mayores en los tiempos de ejecución usando cuaterniones con elementos de otros conjuntos numéricos, obteniendo claves de la misma longitud que las presentadas en [12, 13].

1.2 Motivación y Alcance

En [13] se logra el intercambio de claves de 128 bits a partir de cuaterniones con elementos de 8 bits en 42,59% menos tiempo de ejecución. Para ello, en cada instancia se obtiene un cuaternión que trae consigo $4 \times 8 \text{ bits} = 32 \text{ bits}$. El algoritmo se repite 4 veces para obtener los 128 bits. Sin embargo, en otro trabajo [14], donde se compara el desempeño de aplicaciones que utilizan matrices cuadradas y cuaterniones, se observa que la ganancia en tiempos de ejecución obtenida con

cuaterniones es notoriamente superior, lo cual permitió intuir la existencia de un amplio margen para mejorar aún más los tiempos de ejecución de este cifrador.

El destino del protocolo presentado en [12] es para procesadores de pequeño porte: 16 bits. Sin embargo, se trabaja con elementos de 8 bits. De aquí se puede obtener parte de ese margen disponible solo trabajando con elementos de 16 bits.

Como los primeros ensayos obtenidos fueron favorables, se permite pensar en ampliar la idea a elementos de 32 bits, considerando que hoy también pueden llamarse a los procesadores de 32 bits como de menor porte.

1.3 Objetivo del Trabajo

La finalidad de este trabajo es mostrar que mediante el uso de cuaterniones en el conjunto numérico adecuado pueden obtenerse importantes mejoras en los tiempos de ejecución en el esquema de intercambio de claves Diffie Hellman Compacto.

1.4 Relevancia del tema

La existencia de la computadora cuántica es un hecho. Con ello, la criptografía asimétrica clásica, se encuentra muy debilitada, ya que RSA, el algoritmo más usado ha sido quebrado. Los nuevos desarrollos en criptografía post-cuántica permiten mitigar esta debilidad. Y esquemas más veloces permiten que en la práctica, éstos puedan ser implementados de manera extendida.

1.5 Estructura del Trabajo

En la Sección 2 se presenta el marco teórico. En la Sección 3 se presenta la solución propuesta con datos experimentales. En la Sección 4 se presentan las conclusiones.

2 Marco Teórico

2.1 La Importancia de la Criptografía en la Seguridad de las Comunicaciones

Nociones Básicas de Criptografía Simétrica. La Criptografía se ocupa de asegurar la integridad y confidencialidad en las comunicaciones a través de un canal inseguro. Para ello, el mensaje se transforma en el punto de emisión mediante operaciones matemáticas de manera que sea imposible de interpretar mientras viaja en el canal

inseguro, o bien su costo en tiempo y/o recursos sean tan altos que su descubrimiento carezca de sentido.

Se usan algoritmos criptográficos altamente robustos que permiten además que la información se encripte bit a bit (cifradores de flujo) o en grupos de n-bits (cifradores de bloque) permitiendo que puedan cifrarse comunicaciones en tiempo real [3]. Estos cifradores usan la misma clave para el cifrado y descifrado del mensaje. A estos cifradores se los clasifica como Criptografía Simétrica. Muchos cripto-sistemas simétricos seguros (AES, DES, Trivium) pueden iniciarse con claves de 128 bits.

Nociones Básicas de Criptografía Asimétrica. La criptografía asimétrica o de clave pública, usa elementos públicos que se comparten, y elementos privados que se mantienen en secreto. Generalmente usan propiedades y operaciones de aritmética modular en estructuras algebraicas de anillos de números enteros.

Ésta brindó soluciones al problema de presentar en forma segura claves para su uso en cifradores simétricos: mientras RSA [4] permite que se pueda enviar la clave simétrica cifrada a otro usuario usando su clave pública, con Diffie-Hellman [2] y ElGamal [15] ambas partes pueden generar la misma clave intercambiando elementos.

Amenaza a la Criptografía: El Algoritmo de Shor y la Computación Cuántica.

En 1995 Peter Shor presentó un algoritmo para computación cuántica basado en la transformada rápida de Fourier (FFT) que logra resolver en tiempo polinómico el problema IFP [5]. Es decir, permite reducir drásticamente la complejidad del problema (considerado de clase NP) a niveles atacables [16].

Una computadora cuántica usa qubits en lugar de bit. Un qubit posee los estados 0, 1 y la superposición de ambos: 0 y 1 a la vez. Por ello, se puede realizar una cantidad exponencial de operaciones en paralelo en relación con la cantidad de qubits del computador cuántico.

En 2001 se implementó el algoritmo de Shor en la primer computadora cuántica. La computación cuántica prácticamente arrasa con todo lo conocido en la criptología actual: con ello desaparecen de escena todos los criptosistemas de clave pública: RSA y todas las variantes de ElGamal y Diffie-Hellman [16].

Criptografía Post-Cuántica Basada en Anillos no Conmutativos. Se utilizan estructuras de anillos de matrices cuadradas o de cuaterniones, entre otros, con elementos finitos, por lo tanto, su seguridad radica en la complejidad del tratamiento del problema DLP. Algunos esquemas como el presentado en [17] se basan en la dificultad de resolver el problema SDP (*Simple Decomposition Problem*) en un anillo no conmutativo de polinomios matriciales.

Desde el punto de vista criptográfico, solo se necesita estar seguro que no existe fórmula que permita reducir la complejidad del problema DLP (incluso con

computadora cuántica). Y esto está garantizado ya que en los anillos no conmutativos no existe forma de relacionar el determinante de una matriz o bien sus eigenvalores con la potencia de la matriz [18], parte de la clave privada, independientemente de la cantidad de qubits que pudiera tener una computadora cuántica que ejecute el ataque.

En [12] se muestran más consideraciones de la seguridad de estos esquemas.

2.2 El Anillo no Conmutativo de Cuaterniones

Anillos no Conmutativos. Un anillo $(A ; + ; \cdot)$ es una estructura algebraica (un conjunto A con las operaciones suma y producto) donde $(A ; +)$ forman estructura de grupo, y $(A ; \cdot)$ de semigrupo. Un anillo será no conmutativo si no se verifica la propiedad conmutativa entre todos los elementos de A para la operación producto.

El primer anillo de división (cuyos elementos no nulos son inversibles) no conmutativo fue el anillo de los cuaterniones. Otro ejemplo es el conjunto de matrices cuadradas de orden n con coeficientes en A (simbolizado por $M_n(A)$), es un anillo con respecto a las operaciones usuales de suma y producto de matrices. Si $n > 1$, entonces $M_n(A)$ no es conmutativo. Otros ejemplos de anillos de división no conmutativos son: los Octoniones, Sedeniones, Tessarines, cocuaterniones o bicuaterniones.

Definición: Cuaternión. Sea $(A ; + ; \cdot)$ un anillo conmutativo con unidad. Un cuaternión con coeficientes en A es un número hiper-complejo q de la forma: $q = a + b.i + c.j + d.k$ donde $a, b, c, d \in A$; i, j, k son unidades imaginarias que verifican que: $i^2 = j^2 = k^2 = -1$; $i \cdot j = -j \cdot i = k$; $j \cdot k = -k \cdot j = i$; $i \cdot k = -k \cdot i = j$.

Fueron creados en 1843 por William Hamilton [19]. Forman una estructura de Anillo de división no conmutativo. Tienen una notación compacta y resultan muy sencillos para trabajar. Son muy eficientes: requieren menor cantidad de operaciones básicas y menor espacio de almacenaje en comparación con la operación de matrices.

Operaciones Básicas con Cuaterniones. Sean los cuaterniones $q = (w, x, y, z)$, $q_1 = (w_1, x_1, y_1, z_1)$ y $q_2 = (w_2, x_2, y_2, z_2)$

$$\text{Norma del Cuaternión } q: |q| = \sqrt{q \cdot \bar{q}} = \sqrt{\bar{q} \cdot q} = \sqrt{w^2 + x^2 + y^2 + z^2}$$

Suma, Resta y Producto de un Escalar por un Cuaternión: Se realiza de la misma forma que con cualquier vector de 4 dimensiones.

Producto: $q_1 \cdot q_2 = (w_1 \cdot w_2 - x_1 \cdot x_2 - y_1 \cdot y_2 - z_1 \cdot z_2, w_1 \cdot x_2 + x_1 \cdot w_2 + y_1 \cdot z_2 - z_1 \cdot y_2, w_1 \cdot y_2 - x_1 \cdot z_2 + y_1 \cdot w_2 + z_1 \cdot x_2, w_1 \cdot z_2 + x_1 \cdot y_2 - y_1 \cdot x_2 + z_1 \cdot w_2)$. Notar que el producto entre cuaterniones no es conmutativo.

$$\text{Cociente: } \frac{q_1}{q_2} = q_1 \cdot (q_2)^{-1} = q_1 \cdot \left(\frac{q_2}{|q|^2} \right) \text{ con } q_2 \neq (0, 0, 0, 0) .$$

$$\text{Potencia: } q_1^n = |q_1|^n \cdot (\cos(n \frac{\alpha}{2}) + \check{v} \cdot \text{sen}(n \frac{\alpha}{2})) .$$

2.3 Sistemas Compactos de Intercambio de Claves Diffie-Hellman Basados en Álgebra No Conmutativa

Esquema Diffie - Hellman Compacto con Matrices (DHCM). En [12] se presentó un sistema de intercambio de claves Diffie Hellman [2] sobre anillos de matrices de enteros con elementos en Z_{256} , que utiliza como clave privada un polinomio con coeficientes y exponentes en Z_{16} . Un par de ventajas surgen de ello. Primero: no se requiere el uso de librerías de precisión extendida, por lo tanto, puede ser usado en procesadores de pequeño porte. De allí la calificación de compacto. La otra ventaja se relaciona con el hecho que las matrices conforman estructura de anillo no conmutativo. Gracias a ello, el esquema es inmune a ataques cuánticos y de complejidad sub-exponencial. La clave resultante es una matriz de orden 4 con elementos en Z_{256} conformando así una clave de 128 bits, adecuada para su uso en varios cifradores simétricos seguros.

Resumen del Protocolo. Alice envía a Bob (a través de un canal público e inseguro) dos números enteros aleatorios m y n en Z_{16} , y dos elementos aleatorios A y B , matrices de orden 4 con elementos en Z_{256} . Elige como clave privada un polinomio entero $f(x)$ con coeficientes y exponentes en Z_{16} tal que $f(A) \neq 0$. Bob elige como su clave privada un polinomio entero $h(x)$ con coeficientes y exponentes en Z_{16} tal que $h(A) \neq 0$. Luego Alice y Bob calculan sus tokens: $r_A = f(A)^m \cdot B \cdot f(A)^n$ (Alice) y $r_B = h(A)^m \cdot B \cdot h(A)^n$ (Bob); y se los intercambian para el cálculo de las claves: $K_A = f(A)^m \cdot r_B \cdot f(A)^n$ (Alice), $K_B = h(A)^m \cdot r_A \cdot h(A)^n$ (Bob) con $K_A = K_B$.

Esquema Diffie-Hellman Compacto con Cuaterniones (DHCQ8). Diversas aplicaciones pueden implementarse con cuaterniones en lugar de matrices cuadradas, en menores tiempos de ejecución [14]. Ello inspiró el desarrollo de este esquema [13], que resultó en un ahorro de tiempo cercano al 50% bajo condiciones similares.

Consideraciones del Protocolo. Este protocolo es muy similar al de matrices [12]: En su lugar, los elementos A y B son cuaterniones en forma cartesiana con elementos en Z_{256} . Cada cuaternión, entonces, tiene 32 bits. Se incorporan dos instancias de normalización del cuaternión, y una modularización en Z_{256} . Gracias a esto se puede

aprovechar la notable simpleza de potenciar cuaterniones normalizados, en comparación con la complejidad de potenciar matrices.

3 La Mejora Propuesta

3.1 Resumen de la Mejora Propuesta

El protocolo presentado en [13] trabaja con elementos de Z_{256} . En esta propuesta se trabaja con elementos en $Z_{2^{k*16}}$ para $k = 1$ (DHCQ16) y $k = 2$ (DHCQ32).

3.2 El Protocolo Propuesto

Alice elige dos números enteros aleatorios m y n en Z_{16} , y dos cuaterniones aleatorios A y B , con elementos de $Z_{2^{k*16}}$ (con $k = 1$ o bien $k = 2$) y calcula sus normalizaciones: q_A y q_B . Luego elige como clave privada un polinomio entero $f(x)$ con coeficientes y exponentes en Z_{16} tal que $f(q_A) \neq 0$ y envía a Bob por el canal inseguro los elementos m , n , q_A y q_B . Bob elige como clave privada un polinomio entero $h(x)$ con coeficientes y exponentes en Z_{16} tal que $h(q_A) \neq 0$. Alice y Bob realizan las normalizaciones de $f(q_A)$ y $h(q_A)$: $f'(q_A)$ y $h'(q_A)$. Alice calcula su token: $r_A = f'(q_A)^m \cdot B \cdot f(q_A)^n$. Bob calcula el suyo: $r_B = h'(q_A)^m \cdot B \cdot h(q_A)^n$; y se los intercambian para el cálculo de las claves: $k_A = f'(q_A)^m \cdot r_B \cdot f(q_A)^n$ (Alice), $k_B = h'(q_A)^m \cdot r_A \cdot h(q_A)^n$ (Bob), las cuales se modularizan: $K_A = k_A \cdot 2^{k*16} \pmod{2^{k*16}}$, $K_B = k_B \cdot 2^{k*16} \pmod{2^{k*16}}$ con $K_A = K_B$.

La clave obtenida para $k = 1$ posee 4×16 bits = 64 bits. Para lograr una clave de 128 bits, el proceso debe repetirse. Para $k = 2$, la clave posee 4×32 bits = 128 bits.

3.3 Un Ejemplo Numérico

El ejemplo numérico se realiza para $k = 1$ (DHCQ16), y a los fines de esta presentación, la cantidad de decimales se limita a 2.

Alice elige dos números enteros aleatorios $m=7$ y $n=10$, y dos cuaterniones aleatorios $A = (45606, 11140, 21549, 43028)$ y $B = (42679, 56493, 45062, 43484)$, con elementos de Z_{65536} y calcula sus normalizaciones: $q_A = (0.68, 0.17, 0.32, 0.64)$ y $q_B = (0.45, 0.60, 0.48, 0.46)$. Luego elige como clave privada un polinomio entero

$f(x) = 14x^{15} + 12x^{14} + 13x^{13} + 8x^{12} + 5x^{11} + 4x^{10} + 9x^9 + 13x^8 + 7x^7 + 11x^6 + 4x^5 + 9x^4 + 14x^3 + 7x^2 + 3x + 4$ y envía a Bob por el canal inseguro los elementos m, n, q_A y q_B . Bob elige como su clave privada un polinomio entero $h(x) = 5x^{15} + 6x^{14} + 3x^{13} + 11x^{12} + 14x^{11} + 3x^{10} + 12x^8 + 2x^7 + 3x^6 + x^5 + 14x^4 + 12x^2 + 9x$. Luego Alice y Bob realizan las normalizaciones de $f(q_A)$ y $h(q_A)$: $f(q_A) = (0.78, -0.14, -0.27, -0.54)$ y $h(q_A) = (-0.88, 0.11, 0.21, 0.41)$. Alice calcula su token: $r_A = (-0.37, -0.12, 0.36, 0.26)$, Bob calcula el suyo: $r_B = (-0.17, 0.26, 0.00, 0.39)$; y se los intercambian: $k_A = (-0.25, 0.06, 0.09, -0.11)$ (Alice), $k_B = (-0.25, 0.06, 0.09, -0.11)$ (Bob), las cuales se modularizan: $K_A = k_A \cdot 65536 \pmod{65536} = (49061, 3662, 5778, 58342)$, $K_B = k_B \cdot 65536 \pmod{65536} = (49061, 3662, 5778, 58342)$ con $K_A = K_B = K$: la clave.

3.4 Equipamiento Usado

El computador usado contiene un procesador AMD A10-5745M \times 4 núcleos de 64 bits y 12Gb de memoria RAM. Se instaló una distribución Ubuntu 15.10, el cual tiene un núcleo Linux Debian. Los algoritmos fueron programados en Python 2.7.10.

3.5 Resultados Experimentales

Se presenta una comparación de los tiempos de ejecución del algoritmo DHCM [12], DHCQ8 [13] y la solución aquí propuesta: para 16 bits y 32 bits ($k = 1$ y $K = 2$) para la obtención de 10.000 claves de 128 bits con polinomios aleatorios con coeficientes y exponentes en Z_{16} . La tabla 1 muestra los resultados experimentales.

Tabla 1. Tiempos de ejecución para la obtención de 10.000 claves de 128 bits mediante las diferentes opciones analizadas.

N° Test	CPU Time (s)				N° Test	CPU Time (s)			
	DHCM 8-bits	DHCQ8 8-bits	DHCQ16 16-bits	DHCQ32 32-bits		DHCM 8-bits	DHCQ8 8-bits	DHCQ16 16-bits	DHCQ32 32-bits
1	37,8177	20,0350	11,3851	5,5569	14	37,8728	19,9016	11,6061	5,7219
2	36,4591	19,5770	11,1946	5,7522	15	36,7675	19,3920	11,1752	5,6757
3	37,6444	19,0540	11,3141	5,6158	16	37,7545	19,1195	11,6449	3,9712
4	36,6029	19,4511	11,3369	5,7398	17	37,0721	19,1790	11,3357	5,5914
5	37,3720	20,0530	11,2472	5,6139	18	36,4384	19,6326	11,4020	5,5797
6	37,2393	19,8001	11,0701	5,5895	19	37,0333	19,6599	11,4161	5,5830
7	37,0366	18,9289	11,2918	5,6067	20	37,5093	19,1302	11,1700	5,5174
8	37,3745	19,7008	11,1029	5,7065	21	36,5904	19,6875	11,3885	5,6230
9	37,3369	19,2519	11,2191	5,6430	22	36,8067	19,8909	11,6555	5,7127
10	36,4455	19,7068	11,6477	5,5709	23	37,9800	19,2684	11,4140	5,5620
11	37,6164	19,3665	11,4154	5,5396	24	37,2734	19,4936	11,1356	5,6469
12	37,6196	20,0801	11,1886	5,6568	25	36,9313	19,3517	11,2654	5,5139
13	37,3872	19,7883	11,4547	5,8716					

El tiempo promedio (en segundos) para la obtención de 10000 claves con DHCM fue 37,2, con DHCQ8 fue 19,54, con DHCQ16 fue 11,34 y con DHCQ32 fue 5,57. Como los coeficientes de variación (CV) son $CV < 0,1$ se acepta al promedio como indicador de tendencia central adecuado. Este experimento generó 1 millón de claves sin error.

3.6 Ventajas de la Solución

Velocidad del Cifrador. Menores tiempos de ejecución evidencian mayor velocidad del esquema de cifrado propuesto en comparación con los esquemas presentados en [12] y [13]: El esquema DHCQ8 logra la generación de 128 bits de clave en el 52% del tiempo en comparación con DHCM lo que implica 1,9 veces su velocidad; DHCQ16 lo hace en el 30% implicando más tres veces su velocidad, y DHCQ32 lo hace en el 15% lo que significa más de 6 veces su velocidad. Si la comparación se hace entre los esquemas propuestos y DHCQ8: DHCQ16 lo hace en el 58% del tiempo haciéndolo 1,72 veces más rápido. Sin embargo, DHCQ32 lo hace en el 29% del tiempo, es decir con 3,5 veces más rapidez.

Solución Apta para Procesadores de Pequeño Porte. La implementación de la solución aquí propuesta puede realizarse sin necesidad de uso de librerías de precisión extendida, lo cual lo hace apto para procesadores de pequeño porte.

Inmunidad Frente a Ataques de Complejidad Sub-Exponencial o de Computadora Cuántica. Los anillos de cuaterniones conforman estructuras algebraicas no conmutativas, y sobre este tipo de estructuras no se conocen aún ataques de estos tipos que hayan sido efectivos y que debiliten su seguridad.

4 Conclusiones

Ya se ha visto que usando cuaterniones se puede obtener aplicaciones más rápidas que con matrices. Y esto también es válido en criptografía. Puede aprovecharse la simpleza de la potencia cuaterniones normalizados para lograr mayor velocidad. Trabajar con los conjuntos numéricos adecuados permite hacer uso más eficiente de esta ventaja, logrando implementaciones criptográficas más veloces.

Al usarse estructuras de anillos no conmutativos, hace a este esquema inmune a ataques de complejidad sub-exponencial o de computadora cuántica.

El esquema propuesto es útil para procesadores de menor porte.

Referencias

1. Marrero Travieso, Yran: La Criptografía como elemento de la seguridad informática. ACIMED 11.6 (2003).
2. Diffie W., Hellman M.E: New directions in cryptography. IEEE Transactions on information theory, 22, 644-654, (1976).
3. Menezes, Alfred J., Paul C. Van Oorschot, and Scott A. Vanstone: Handbook of applied cryptography. CRC press (1996).
4. Rivest, Ronald L., Adi Shamir, and Len Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM 21.2, 120-126. (1978)
5. Shor P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput., 5, 1484-1509 (1997)
6. D-Wave-Systems Press Releases [en línea], (2016). Disponible en: <<http://www.dwavesys.com/news/press-releases>>. Fecha de consulta: 05/06/2016.
7. IBM: IBM Makes Quantum Computing Available on IBM Cloud to Accelerate Innovation [En Línea], (2016). Disponible en: <<https://www-03.ibm.com/press/us/en/pressrelease/49661.wss>>. Fecha de consulta: 05/06/2016.
8. Magliveras S.S., Stinson D.R., van Trung T.: New approaches to designing public key cryptosystems using one-way functions and trapdoors in finite groups, Technical Report CORR, 2000-2049 (2000)
9. Shpilrain V., Zapata G.: Combinatorial group theory and public-key cryptography, Preprint arXiv/math.gr, 0410068 (2004)
10. Barreto, P. et al: Introdução à criptografia pós-quântica, Minicursos do XIII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais, SBSeg, (2013).
11. Gerritzen L. et al (Editors): Algebraic Methods in Cryptography, Contemporary Mathematics, AMS, Vol. 418, (2006)
12. Hecht J.: Un modelo compacto de criptografía asimétrica empleando anillos no conmutativos. V Congreso Iberoamericano de Seguridad Informática CIBSI, Montevideo (2009).
13. Kamlofsky J.A., Hecht J.P, Abdel Masih S., and Hidalgo Izzi, O.: A Diffie-Hellman Compact Model Over Non-Commutative Rings Using Quaternions. VIII Congreso Iberoamericano de Seguridad Informática CIBSI, Quito (2015).
14. Kamlofsky J., Bergamini L.: Cuaterniones en Visión Robótica. V Congreso de Matemática Aplicada, Computacional e Industrial MACI, Tandil (2015).
15. Elgamal, Taher. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. En Advances in cryptology. Springer Berlin Heidelberg, pp. 10–18 (1984).
16. Hecht, JP.: Fundamentos de Computación Cuántica. Editorial Académica Española. ISBN 978-3-8484-7529-2 (2005).
17. Cao Z., Xiaolei D., Wang L.: New public-key cryptosystems using polynomials over non-commutative rings, Preprint arXiv/cr, eprint.iacr.org/2007/009.pdf (2007).
18. Eftekhari, M.: A Diffie–Hellman key exchange protocol using matrices over noncommutative rings. Groups-Complexity-Cryptology, 4(1), pp. 167–176 (2012).
19. Hamilton, W. R.: Lectures on Quaternions: Containing a Systematic Statement of a New Mathematical Method, Hodges and Smith, (1853)

Dispositivos móviles y el fenómeno del BYOD. Su impacto en la seguridad de las organizaciones

Paula Venosa, Nicolás Macia, Carlos Damián Piazza Orlando y Sebastián
Exequiel Pacheco Veliz

Facultad de Informática
Universidad Nacional de La Plata
Argentina
{pvenosa,nmacia}@info.unlp.edu.ar
{cpiazza,spacheco}@cespi.unlp.edu.ar

Resumen Las amenazas que atentan contra la seguridad de los dispositivos móviles: malware, robo, fuga de información, vulnerabilidades de software y phishing entre otras ponen en riesgo la seguridad personal de los usuarios y de los activos de información de las organizaciones en las que estos trabajan.

Este artículo trata sobre el impacto del uso de los dispositivos móviles personales en las organizaciones lo cual se conoce como: Bring Your Own Device (BYOD). Este fenómeno constituye un campo de interés para quienes estudian problemáticas relacionadas a la seguridad de la información en dispositivos móviles, puedan expandir sus análisis a ambientes organizacionales.

Se presentan las líneas de trabajo abordadas, herramientas probadas, guías y buenas prácticas generadas para usar de manera segura dispositivos móviles en una organización.

fsd

Palabras Claves: Dispositivos móviles, Seguridad, Amenazas, BYOD, (IT) Tecnologías de información, (MDM) Mobile Device Management.

1. Introducción

El uso de dispositivos móviles ha aumentado considerablemente y la diversidad de tecnologías y plataformas conlleva a que aparezca un nuevo desafío en materia de investigación en lo que hace a aspectos relacionados con la seguridad y privacidad de la información.

El fenómeno del BYOD es una tendencia que se expande cada día en las organizaciones cualquiera sea su tamaño: hoy en día es muy común que los integrantes de las mismas hagan uso de sus dispositivos móviles personales para realizar sus labores en la organización. Lo que era antes un dispositivo de uso

personal se convierte en parte de la red de la organización lo cual, es cómodo para el usuario, pero también implica riesgos a la seguridad de la organización. Esta tendencia causa problemas a los administradores de los departamentos de TI (Tecnologías de Información) a la hora de implementar políticas de seguridad. Ahora hay distintos tipos de dispositivos personales sobre los que no se posee ningún control que solicitan acceso a la red de las organizaciones. Por ello se requiere la implementación de políticas MDM (Mobile Device Management), que conlleven a una gestión exitosa de este abanico amplio de dispositivos móviles, en pos de implementar la movilidad organizacional de manera segura. Por lo mencionado, este fenómeno constituye un campo de investigación de interés para aquéllos que estudian problemáticas relacionadas a la seguridad de la información y su aplicación en tecnologías actuales, análisis de protocolos de seguridad y mecanismos de protección disponibles y en desarrollo.

1.1. Amenazas latentes en el uso de dispositivos móviles

Los problemas de seguridad a los que los dispositivos móviles están expuestos son similares a los que está expuesta una computadora[3], pero se ven agravados ya que cuentan con una mayor exposición al ser su comunidad de usuarios más amplia, además de que se los utiliza tanto en el ámbito laboral como en el personal.

La problemática se potencia tanto debido al desconocimiento general sobre los problemas de seguridad a los que los dispositivos están expuestos, como a la falta de información en relación a las contramedidas que se pueden adoptar. Si eso lo sumamos a con la cantidad de elementos incluidos en los smartphones surgen otros problemas, como por ejemplo los relacionados con el espionaje, puesto que un dispositivo comprometido podría permitir consultar su localización vía GPS, transmitir la información captada por su micrófono o incluso su cámara.

A la hora de determinar las distintas amenazas que podrían atacar la integridad del dispositivo y de los datos en él contenidos, podemos armar la siguiente clasificación general:

Malware. Es una clasificación general de software malintencionado en la que se incluyen los virus, rootkits, troyanos, etc., y es más comúnmente relacionado con la destrucción o robo de datos. La definición de malware ha cambiado durante los últimos años y se puede dividir en dos categorías: tradicional o moderno. El malware tradicional se refiere a la forma clásica de malware en la que el objetivo es infectar y propagarse al mayor número de dispositivos posible, al tiempo que maximiza el daño al sistema, sin ningún motivo específico detrás de un ataque. Los malware actuales son más sofisticados y estratégicamente planificados. Una de las principales diferencias del malware moderno en comparación con el tradicional, es que el malware moderno está dirigido y es sigiloso. El objetivo es elegido específicamente ya que tiene algo de valor para el atacante y evitar la detección es esencial con el fin de llevar a cabo un ataque prolongado. Los autores de malware modernos han comenzado a focalizarse en el desarrollo de malware para dispositivos móviles y Android es una de las plataformas a las que más provecho le sacan.

Robo y Fuga de Datos. La seguridad de la información es esencial para cualquier persona u organización que quiere proteger su información confidencial. La fuga de datos y el robo en el contexto de los dispositivos móviles se refieren a datos que han sido almacenados en el dispositivo móvil, y la existencia de varias maneras para que estos datos sean filtrados o robados por un tercero. El malware es una de las maneras de lograr el robo y la fuga de información, aunque también puede ser robada por ataques que implican acceso físico al dispositivo.

Vulnerabilidades de Software. Las vulnerabilidades de software son un riesgo de seguridad constante para cualquier aplicación o plataforma. En el tiempo entre el descubrimiento de una vulnerabilidad hasta que sea arreglada y parcheada, una aplicación corre el riesgo de ser explotada. Los navegadores Web son un ejemplo típico y que se puede encontrar básicamente en cualquier PC o dispositivo móvil. Si los autores de malware descubren una falla, entonces pueden explotarla con el fin de inyectar código malicioso que puede afectar a millones de dispositivos hasta el momento en que el defecto sea parchado en el sistema del usuario.

Ingeniería Social. En términos técnicos, la ingeniería social se refiere a un ataque en el que se utilizan las habilidades sociales para obtener información, ya sea personal o relacionada a la organización. Las personas que practican la ingeniería social (estafadores) utilizan la interacción social para engañar a sus víctimas logrando que confíen en él. El engaño podría ser cara a cara, por teléfono, mensajería, e-mail, etc. Por lo general, el atacante fingirá ser otra persona, por ejemplo, un nuevo empleado en la misma organización como la víctima, un conocido de otro conocido, amigo de un amigo, personalidad pública, etc.

Phishing. Es una forma de ingeniería social, pero por sus características merece ser mencionado de manera particular. Un ataque de phishing es un intento de obtener una parte de la información sensible de los usuarios haciéndoles creer que es por una causa legítima. El ataque se inicia normalmente utilizando el correo electrónico o sitios web maliciosos como herramienta primaria. Un correo electrónico de phishing se disfraza por ejemplo, del banco de la víctima o del departamento de IT de los lugares de trabajo de las víctimas, y pueden contener una investigación respecto a la información personal o financiera, tales como información de tarjetas de crédito y contraseñas.

Ransomware. Una amenaza con múltiples ataques que se están registrando en estos últimos años es el ransomware. Es un tipo de malware que restringe el acceso a determinadas partes o archivos del sistema infectado, y pide un rescate a cambio de quitar esas restricciones. Algunos tipos de ransomware cifran los archivos del sistema operativo utilizando el dispositivo y coaccionando al usuario a pagar el rescate. Al igual que otros tipos de malware para Android[4], a medida que las amenazas de ransomware fueron evolucionando en los últimos años, sus creadores adoptaron muchas de las técnicas que les resultaron eficaces para atacar equipos de escritorio y las aplicaron a la plataforma móvil. El ransomware, como lo indica su nombre, es cualquier tipo

de malware que le exige al usuario infectado el pago de una suma de dinero a cambio de la promesa de "liberar" o "rescatar" un recurso secuestrado.

Jailbreak IOS. Los dispositivos iOS parecen ser bastante seguros hasta ahora. Sin embargo, esta declaración sólo se aplica a los dispositivos que no han sido jailbroken. El jailbreak es el término genérico que se le ha puesto a los métodos que hay para saltarse las medidas de seguridad impuestas por Apple en su sistema iOS y poder instalar, modificar y cambiar cualquier aspecto del sistema. Un dispositivo con jailbreak[5] ofrece a los usuarios y desarrolladores el acceso a los recursos que se le fueron prohibidos. Sus principales usos son modificación del sistema, personalizar más el sistema con fuentes de letra diferentes, animaciones, nuevas funciones que no están en iOS. Descarga de aplicaciones que no han sido aceptadas en la App Store o aplicaciones de pago gratuitas. Los usuarios ya no se limitan a sólo usar App Store para obtener más aplicaciones y por lo tanto podrían utilizar copias de contrabando de aplicaciones legítimas (de sitios de terceros) que podrían ser maliciosas.

Android Rooting. Root, Rooting o Rutear(Rootear) se le conoce al método utilizado para darle al usuario los privilegios de Administrador o Super Usuario(SuperUser) del sistema operativo Android. Este concepto proviene del OS Linux y hace referencia al mismo comportamiento en Android ya que este también es un tipo de Linux. El proceso de Root es necesario cuando se quieren ejecutar ciertas aplicaciones que necesitan privilegios especiales para realizar tareas que entran en conflicto con directivas de seguridad impuestas por el desarrollador del SO (En este caso, Google). Siendo root se obtiene el control total sobre el sistema Android: se puede manipular cualquier archivo a nivel de sistema, e incluso se puede ser capaz de cambiar de sistema operativo manualmente o modificarlo a niveles avanzados, como cambiar la velocidad del microprocesador. En este punto es donde nos detendremos y prestaremos especial atención. Rootear un dispositivo abre la puerta a un número mayor de peligros que los que tiene un dispositivo que se encuentra, digamos, "de fabrica"[6]. Las restricciones impuestas por Android a nivel de permisos de aplicación están pensados para que las zonas de memoria donde puede acceder una aplicación estén delimitadas dentro de su sandbox según los permisos que le fueron otorgados en su instalación, evitando accesos no deseados a información que la aplicación no tiene que conocer ni modificar. Siguiendo esta línea, en un móvil Rootead, una aplicación maliciosa podría solicitar adquirir privilegios de Root lo que le permite ignorar el esquema de permisos (como dijimos, ser root es tener permisos de administrador, lo que hace que las restricciones desaparezcan) y acceder a cualquier parte del sistema aunque no haya declarado y adquirido los permisos necesarios.

Como se puede ver, existe un amplio espectro de amenazas que pueden atacar contra la seguridad de un dispositivo móvil. Además, gracias al fenómeno del BYOD estas amenazas también atentarán contra la seguridad de la organización. Debido a esto, resulta necesario para las organizaciones que acepten el BYOD tener en cuenta los problemas mencionados con la finalidad de poder mantener un ambiente organizacional seguro. Para ello, será necesario que las

organizaciones desarrollen y apliquen políticas de uso, uso de software de gestión para dispositivos móviles (MDM) y capaciten a sus usuarios, entre otras cosas.

2. El fenómeno BYOD y la seguridad

2.1. ¿Qué es BYOD?

Bring Your Own Device[7] es un fenómeno cultural y tecnológico que incentiva a los miembros de una organización a utilizar sus propios dispositivos móviles personales en las actividades laborales de la organización donde trabajan.

El fenómeno del BYOD es una tendencia que se expande cada día en las organizaciones de todos los tamaños gracias a la reducción de precios de dispositivos móviles tales como smartphones, tablets y laptops. Hoy en día es muy común que los miembros de una organización hagan uso de sus dispositivos móviles personales para realizar sus labores en la organización. La razón por la que esto ocurre es la premisa de que quien trabaja cómodo trabaja mejor. Además, algo que ha contribuido a esta tendencia es la explosión de smartphones y tablets en los hogares y que la tecnología usada en casa y en la oficina puede estar interconectada.

Debido a todo esto, a partir del BYOD, lo que era un dispositivo personal se convierte en parte de la red organizacional con los problemas que esto podría acarrear en lo que a seguridad se refiere.

2.2. BYOD y buenas prácticas de seguridad en la organización

A partir del fenómeno del BYOD anteriormente descrito, surgen líneas claras en las que las organizaciones deben trabajar para mantener la seguridad de la red de la organización. Entre estas líneas podemos mencionar:

- Definición de política de seguridad en el uso y la gestión de dispositivos móviles.
- Uso de aplicaciones para el Mobile Device Management (MDM).
- Concientización y capacitación del usuario.

Definición de políticas de seguridad en el uso y la gestión de dispositivos móviles

Una política BYOD es un conjunto de reglas que gobiernan los aspectos relacionados con el uso de dispositivos personales para acceder y utilizar recursos de la organización. Una política es algo específico para cada organización puesto que debe basarse en los requerimientos de la organización, su perfil de riesgo y su situación.

A la hora de definir la política, es conveniente tener en mente que, lo que impulsa los programas de movilidad organizacional, es la necesidad de proporcionar acceso seguro y transparente (es decir, con la mejor experiencia de usuario) a los recursos de la organización en cualquier momento, en cualquier lugar y desde

cualquier dispositivo. Las iniciativas de movilidad deben contribuir a facilitar la continuidad del trabajo, mejorar la colaboración, simplificar el teletrabajo y mejorar la satisfacción de los miembros. La política debe ser clara, concisa, realista, sostenible y adaptada a los usuarios que la van a utilizar. Una política que pretende influenciar el comportamiento de los empleados no puede estar escrita en un complicado lenguaje técnico ni alejarse de la cultura y estilo de la organización.

Aunque en un principio la elaboración de la política pueda parecer una tarea abrumadora, hay que tener en cuenta que la peor política es la ausencia de la misma. Cuando los miembros de la organización saben qué se espera de ellos, cuáles son los comportamientos aceptables y las consecuencias del incumplimiento, es menos probable que rompan las reglas.

Pero las políticas por sí mismas no son suficientes para evitar las consecuencias del incumplimiento de las normas, ya sea por error, accidente o decisión voluntaria del usuario. Necesitaremos herramientas que nos ayuden a hacer cumplir las políticas, como pueden ser los sistemas de gestión de dispositivos móviles (Mobile Device Management, MDM), sistemas de gestión de aplicaciones móviles, etc.

Mobile Device Management

Las herramientas Mobile Device Management (MDM) permiten la gestión de los dispositivos móviles de una organización. Las herramientas tipo MDM sirven para asegurar, monitorear y administrar dispositivos móviles de forma centralizada. La gestión se realiza desde un servidor centralizado en el que se pueden definir políticas, actuaciones o incluso localizar o realizar un borrado remoto de la información contenida en un dispositivo móvil perdido evitando así por ejemplo, la extracción de datos de su memoria.

Como ejemplo, con un sistema MDM, la organización podría establecer la política de que un dispositivo que tiene determinada aplicación instalada no pueda acceder a la red. En función de quién sea el propietario del dispositivo se podrán aplicar ciertas funcionalidades o no. Por ejemplo, en caso de que el propietario del dispositivo sea el propio usuario, no se podrá negar al usuario la instalación de aplicaciones no productivas para la organización. En cambio, el departamento IT sí podrá controlar qué aplicación del dispositivo particular puede acceder a datos corporativos y cual no.

Concientización del usuario

Para evitar que los usuarios de una organización sean el destino de las distintas amenazas mencionadas, las organizaciones deberían implementar acciones de concientización y capacitación de sus usuarios. Estas acciones se realizan en el marco de la política de seguridad intenta reducir, por ejemplo, la posibilidad de que un usuario introduzca algún malware y exponga datos sensibles de la organización a terceros.

Es bueno que una organización implemente un plan de concientización con el objetivo de concientizar y fomentar hábitos cotidianos de seguridad en sus

empleado. La comunicación es una pieza fundamental del programa BYOD. De nada sirve establecer políticas de seguridad, herramientas de seguridad y gestión, así como programas de soporte, si los usuarios no son conscientes de ello. Los usuarios deben conocer qué pretenden las políticas BYOD y de seguridad establecidas, los usos aceptados, por qué es importante el cumplimiento de las políticas y qué herramientas va a utilizar el área de TI para implementarlas. Además deben recibir formación sobre las medidas de seguridad que deben implementar en sus dispositivos y el porqué de ello (por ejemplo, de nada sirve que la organización implemente el más seguro de los sistemas de autenticación, si los usuarios dejan sus credenciales al descubierto). Los usuarios deben conocer también las consecuencias del incumplimiento de las políticas.

3. Experiencias

A partir de la experiencia de nuestro grupo en el área de seguridad tanto en el marco de CERTUNLP[8][10][11][12][13][14][15] como en el de distintos proyectos de extensión relacionados a Concientización en Seguridad de la Información[1][2][9], los autores de este artículo venimos trabajando hace un tiempo en cuestiones relacionadas a seguridad en dispositivos móviles. En el ámbito de la UNLP, teniendo en cuenta las líneas de trabajo descritas en la sección anterior, se han desarrollado diversas tareas que se describen a continuación.

En primer lugar, se investigó sobre la temática y se realizaron recomendaciones de seguridad respecto a cuestiones de BYOD. Estas recomendaciones se realizaron teniendo en cuenta estándares de referencia como lo es la ISO 27000[16], y fueron aplicadas en nuestro ámbito de trabajo. La metodología utilizada ha sido la de construir una matriz de criticidad, que permita identificar fácilmente cuáles son los dispositivos más críticos para la organización. Esto puede estimarse en base a la criticidad de la información que dicho dispositivo almacena y además teniendo en cuenta el grado de exposición que el mismo tiene.

Luego, en base a la criticidad de la información y al grado de exposición, se ha armado una matriz que determina la forma de aplicar las recomendaciones de seguridad, definiendo básicamente prioridad en cuanto a aplicar controles a los dispositivos móviles. También como parte de este marco, se han definido recomendaciones, que incluyen buenas prácticas, para los distintos tipos de dispositivos móviles con los que cuenta la organización, así como un procedimiento a seguir en caso de pérdida o robo.

Por otro lado, dado que se hace imprescindible contar con alguna herramienta de MDM nos introdujimos en el tema a través del testeo de la herramienta SOTI Mobicontrol[17], ya que tuvimos la oportunidad de contar con una licencia por 30 días que se nos suministró para poder conocer la herramienta, ver sus características y entender mejor el funcionamiento de una herramienta de MDM desde una visión práctica.

SOTI Mobicontrol es una herramienta comercial para el manejo de dispositivos organizacionales (Enterprise Mobility Management - EMM)[18]. Entre las funcionalidades que provee se encuentra el manejo de aplicaciones, contenido,

información, servicios de localización, filtros web, filtros de llamada, control de malware y hasta control del plan de tarifas de los dispositivos que se registren en la herramienta. Actualmente posee soporte para dispositivos Android, iPhone, Windows Phone permitiendo definir un repositorio de aplicaciones propio para que los dispositivos puedan seleccionar las aplicaciones a instalar de un listado brindado por el administrador del sistema y luego descargarlas ya sea desde el store de aplicaciones o desde el servidor donde Mobicontrol este corriendo, permitiendo tener un mayor control de las aplicaciones que los dispositivos de la organización pueden o deben tener instaladas.

La herramienta funciona sobre la arquitectura Windows, tanto en versiones Desktop (Windows 7 en adelante, versiones de 64 bits) como Server. Para el almacenamiento de la información utiliza una base de datos de tipo SQLServer (2008 en adelante) que puede instalarse por separado.

La infraestructura utilizada en las pruebas para el servidor de Mobicontrol fue: Una máquina virtual en VirtualBox con Windows 7 y SQLServer 2008 R2 instalado de manera independiente ya que el instalador de SQLServer que provee la herramienta presentaba problemas. Las pruebas se realizaron principalmente para la tecnología Android. Los dispositivos utilizados para las pruebas fueron:

- Movil Sony Xperia U con Android Ice Cream Sandwich 4.0.4
- Movil Motorola MotoG con Android KitKat 4.4

La herramienta provee un panel de control desde donde se pueden administrar aspectos relacionados a certificados para la conexión https, hostname del servidor, estado de los servicios, etc; y el acceso al sistema de administración de dispositivos se realiza mediante un navegador.

Soti Mobicontrol resulta una herramienta de MDM muy completa que abarca muchos de los conceptos necesarios para la administración de dispositivos organizacionales de manera centralizada. Las pruebas realizadas nos han permitido entender el funcionamiento de este tipo de servicios. Una desventaja que presenta es que sólo existe una versión para sistemas operativos Windows y que no pertenece a la familia de software Open Source, decompilando el .apk de la aplicación agente que se instala en los dispositivos el código se encuentra ofuscado para quedar ilegible.

Por último, en sintonía con las líneas descritas, en el marco de la semana de la seguridad de la información de noviembre pasado, se ha realizado una jornada denominada "Concientización sobre seguridad de la información. Redes sociales y smartphones: principios, riesgos y cuidados"[19], destinada a usuarios de la UNLP de direcciones de enseñanza de Facultades, Colegios (SIPECU) y postgrados. En la misma se abordaron riesgos en el uso de dispositivos móviles personales en el ámbito de nuestra organización: la Universidad Nacional de La Plata.

4. Conclusiones

De la misma manera que con las PC, los dispositivos móviles son un blanco de los ataques contra la seguridad de la información. Los problemas de seguridad

a los que están expuestos son similares. Al contar los dispositivos móviles con GPS, cámara y micrófono hacen que estos sean un objetivo valioso.

La seguridad de los dispositivos móviles constituye una problemática actual de gran interés para las organizaciones cualquiera sea su tamaño, y en particular también para las del ámbito académico como la nuestra, donde el fenómeno del BYOD se presenta como una realidad difícil de restringir.

A la hora de diseñar e implementar soluciones y gestionar la seguridad de los dispositivos de la organización, se debe abordar la problemática en forma integral, teniendo en cuenta como se describe en este trabajo: la implementación de políticas, normas y procedimientos (establecimiento un marco de seguridad aplicable a la organización), herramientas que faciliten la administración de la seguridad en los activos de la organización y concientizando a los empleados de la organización a través de actividades específicas para fomentar buenas prácticas en el marco del BYOD.

Referencias

1. Lic. Nicolás Macia - Lic. Einar Lanfranco - Lic. Paula Venosa - Lic. Alejandro Sabolansky A.P.U. Carlos Damián Piazza Orlando - A.P.U. Sebastian Exequiel Pacheco Veliz: Uso de dispositivos móviles y BYOD: Su impacto en la seguridad. XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015).
2. Lic. Nicolás Macia - Lic. Einar Lanfranco - Lic. Paula Venosa - Lic. Alejandro Sabolansky A.P.U. Carlos Damián Piazza Orlando - A.P.U. Sebastian Exequiel Pacheco Veliz: Seguridad en dispositivos móviles: un enfoque práctico. XVI Workshop de Investigadores en Ciencias de la Computación (Usuahia, 2014).
3. Reporte de malware por F-secure - https://www.f-secure.com/documents/996508/1030743/Threat_Report_H1_2014.pdf
4. El auge del ransomware para Android: criptográfico y de bloqueo de pantalla http://www.welivesecurity.com/la-es/2016/02/18/auge-ransomware-para-android/?utm_source=newsletter&utm_medium=email&utm_campaign=wls-newsletter-190216
5. Malware attacks Jaibroken IOS devices <http://researchcenter.paloaltonetworks.com/2015/08/keyraider-ios-malware-steals-over-225000-apple-accounts-to-create-free-app-utopia/>
6. The risks of rooting your Android phone <http://www.bullguard.com/bullguard-security-center/mobile-security/mobile-threats/android-rooting-risks.aspx>
7. BYOD Retos de la seguridad - http://www.welivesecurity.com/wp-content/uploads/2014/01/documento_guia_byod_W.pdf
8. Díaz, Francisco Javier; Venosa, Paula; Macia, Nicolás; Lanfranco, Einar Felipe; Sabolansky, Alejandro Javier; Rubio, Damián: Análisis digital forense utilizando herramientas de software libre. XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina)
9. Compromiso social y calidad educativa: desafíos de la extensión Proyecto Caperucita y el lobo en el ciberespacio. Concientización en seguridad informática para jóvenes y tercera edad. IV Congreso Nacional de Extensión Universitaria y a las IX Jornadas Nacionales

10. Lanfranco, Einar Felipe; Benencia, Raúl; Macia, Nicolás; Venosa, Paula: Automatizando la gestión de configuraciones en pos de la seguridad. XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015).
11. Javier Díaz, Alejandra Osorio, Paola Amadeo, Nicolás Macia, Paula Venosa: Seguridad Proactiva en los Sistemas de Gestión e Información académica: el caso de la UNLP. TICAL (Chile, 2015)
12. Traberg, Gastón; Molinari, Lía; Venosa, Paula; Macia, Nicolás; Lanfranco, Einar Felipe. Automatizando el descubrimiento de portales de autenticación y evaluación de la seguridad mediante ataques de fuerza bruta en el marco de una auditoría de seguridad. XXI Congreso Argentino de Ciencias de la Computación (Junín, 2015).
13. Santiago Alessandri, Matias Fontanini, Nicolás Macia. Service-Knocking Communication. 41JAIIO. Año 2012.
14. Lanfranco, Einar Felipe; Macia, Nicolás; Venosa, Paula; Molinari, Lía; Díaz, Francisco Javier: Tendencias en incidentes de seguridad atendidos por el CERT académico Cert-UNLP. XII Workshop de Investigadores en Ciencias de la Computación (2010)
15. Díaz, Francisco Javier; Foster, Pablo Mauricio; Lanfranco, Einar Felipe; Macia, Nicolás; Molinari, Lía; Venosa, Paula: Definición e Implementación de un Centro de Atención de Incidentes (CERT) para un ámbito Universitario. XV Congreso Argentino de Ciencias de la Computación (2009)
16. ISO 27000 - http://www.iso27000.es/download/doc_iso27000_all.pdf
17. Mobicontrol - <https://www.soti.net/products/mobicontrol/overview/>
18. Enterprise mobility management (EMM) - <http://searchmobilecomputing.techtarget.com/definition/enterprise-mobility-management-EMM>
19. Concientización sobre seguridad de la información. Redes sociales y smartphones: principios, riesgos y cuidados - http://www.cespi.unlp.edu.ar/articulo/2015/11/25/el_cespi_realizara_el_evento__equipo_certunlp__jornada_de_seguridad_de_la_informacion_

Loss of Votes in NIDC

Applying Storage in Parallel Channels

Pablo García¹, Germán Montejano^{1,2}, Silvia Bast¹, and Estela Fritz¹

¹ FCEyN - Universidad Nacional de La Pampa - Argentina
pablogarcia@exactas.unlpam.edu.ar,

WWW home page: <http://www.exactas.unlpam.edu.ar>

² FCFMyN - Universidad Nacional de San Luis - Argentina
gmonte@unsl.edu.ar,

WWW home page: <http://www.unsl.edu.ar>

Abstract. Birthday Paradox states that in a group of 23 people, the probability that there are at least two who share the same birthday is very close to $\frac{1}{2}$. This assertion is unacceptable for any scheme that proposes a vote storage method based on a vector of slots whose position is chosen at random. In this situation, it may produce collisions. A collision occurs when two or more votes are stored in the same slot. It produces the loss of the coincident votes. This is the original model of the Non - Interactive Dining Cryptographers (NIDC) protocol. The actual paper shows new achieved results obtained by analyzing the behaviour of a storage technique based on parallel channels. This scheme consists of replicating each vote in Q parallel channels, keeping the total number of slots (T) without variation.

Keywords: Parallel Channels, Storage Birthday Paradox, Non - Interactive Dining Cryptographers, Collisions.

1 Introduction

Within the scope of a research line that began at 2013 and which was formally presented in [1], the exact security level requested for anonymity in an electronic voting scheme was analyzed. Many of the proposed schemes (Mix Net based) give unconditional security to the votes' information and computational assurance to voter's privacy. However, it is easy to see that it is an erroneous proposal . In [2] it is concluded that it is necessary to give unconditional security for the privacy, because it must be protected indefinitely. Otherwise, votes must be kept for a finite period of time.

Consequently, those schemes, that include unconditional security as the main feature, acquire maximum interest. In this sense, one of the most interesting is Dining Cryptographers (DC), which is described in detail in [3]. This protocol is resourceful and gives unconditional privacy.

The analysis is focused on a derivative of DC, called Non Interactive Dining Cryptographers (NIDC [4]), that relaxes the condition of concurrency online

for all participants. This protocol is suitable to be applied to electronic voting scheme.

The original version of NIDC, stores data in a vector of slots. This is observed in figure 1.

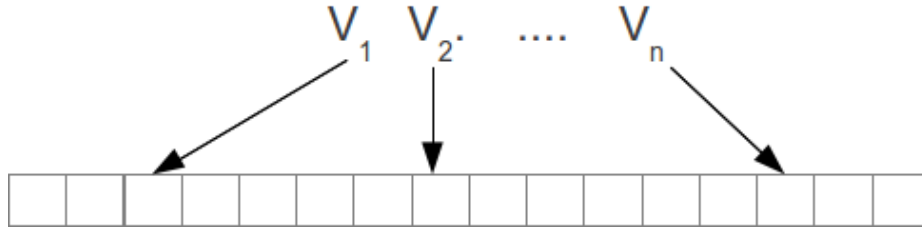


Fig. 1. Original NIDC Storage

If two or more votes are stored in the same slot, a collision occurs. That results in the loss of coincident votes. Simultaneously, the proposition of true randomness for the choice of position indicates that collisions may happen. It then seeks to ensure that the proportion of lost data is kept below a desired value with a certain probability. The proposed model in Figure 1 may be explained by Birthday Paradox ([5]). In those conditions, it is required a very significant number of slots to obtain suitable security levels.

Two interesting alternatives, aimed at improving the Birthday Paradox effect are presented in [6]. In that document an optimization for NIDC, applying multiple networks in serie and in parallel, is proposed. In this case, however, it seeks to generalize the approach to storage in parallel channels, a matter that may be generalized to multiple real-world problems, including NIDC. The alternative proposal consists on implementing N parallel channels, replicating each vote in all channels, in potentially different random positions in each case, as outlined in Figure 2.

It begins by describing the parameters involved:

T : # Total slots to implement. $T \in \mathbb{Z}^+$.

S : # Parallel slots on each channel. $S \in \mathbb{Z}^+ \wedge S \leq T$.

N : # Voters. $N \in \mathbb{Z}^+$.

Q : # Parallel channels to implement. $Q \in \mathbb{Z}^+$.

Q_{to} : # Parallel channels to implement (Theoretically Optimal). $Q_{to} \in \mathbb{R}^+$.

Q_{po} : # Parallel channels to implement (Practically Optimal). $Q_{po} \in \mathbb{Z}^+$.

R : # Replicas of a vote on the same channel. $R \in \mathbb{Z}^+$.

PLV : Percentage of Lost Votes.

Throughout previous papers ([7], [8] and [9]) the following relevant findings have been set forth (in addition, function CEIL will be used; it computes the nearest higher integer. That is necessary because Q_{to} could be non integer):

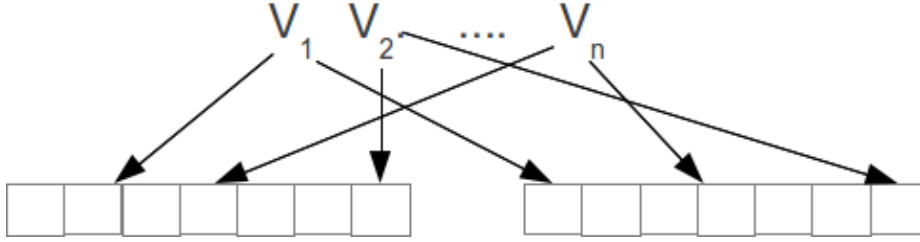


Fig. 2. Scheme based on Parallel Channels

- For a fixed number of voters N , the recommended number of slots for each parallel channel (S) is given by the formula:

$$S = CEIL\left(\frac{N}{\ln 2}\right) \quad (1)$$

- For given values of T and N , there exists an optimal number of parallel channels. Such value is expressed:

$$Q_{to} = \ln 2 \frac{T}{N} \quad (2)$$

That formula should be taken to the next integer.

$$Q_{po} = CEIL(Q_{to}) \quad (3)$$

- An appropriate lower bound for the probability of $X =$ "no vote is lost" is obtained by applying equation:

$$Pr(X) > \left(1 - \left(\frac{1}{s}(n-1)\right)^q\right)^n \quad (4)$$

Besides, concrete methods were published to obtain optimal values for all parameters using a spreadsheet ([10]) and the pseudocode algorithm that must be applied for the same purpose was shown in [11].

In this document the variable PLV is analyzed. One equation is described to get an approximation of the expected value of that variable. The following section describes the deduction of such formula .

2 Expected Value of Percentage of Lost Votes (PLV)

By applying a parallel channels scheme, a question that quickly arises is: for a situation with N voters, and Q parallel channels of S slots (such that $T = SQ$), which is the expected Percentage of Lost Votes (PLV)?

At the beginnig, it is considered the original Birthday Paradox proposal. The first thing we see is that even in the best case (the 23 people Birthdays on different dates), the number of slots that will not be used is 342, then we have approximately 6,3 % of occupied slots and 93,7 % of empty slots. Consequently, for each slot containing a vote, more than 15 receive no ballots.

The proposal is to divide all the slots in $Q > 1$ parallel channels and to deposit an occurrence of each vote in each of the channels. In addition to what appeared in the simulations, the idea is related to the fact that a vote is lost on a given channel is independent of what happens in the other $Q - 1$ channels.

Independent events verify that:

$$Pr(A \cap B) = Pr(A)Pr(B) \quad (5)$$

Clearly a vote will be lost only if it collides on all the channels. The number of local collisions increases, since each channel will have a measure smaller than the single vector. However, an optimization based on replicas is obtained.

Let ε be:

$$\varepsilon = \frac{N}{S} \quad (6)$$

Initially the situation is analyzed if a single vector is implemented, therefore, $S = T$.

Several strategies based on analyzing the probability distribution are presented in [12]. Also, the approaches proposed by Feller [13], were mentioned. These approaches improve their behaviour when $N \rightarrow \infty$ y $T \rightarrow \infty$. A tool that may be useful is Stirling's approximation for calculating factorials:

$$N! = \sqrt{2\pi N} \left(\frac{N}{e}\right)^N \quad (7)$$

It is proposed another approach, which is simpler than the previous one because it only calculates expected values rather than probability distribution.

Considering the first vote, the probability that it falls into the slot 1 is:

$$p = \frac{1}{S} \quad (8)$$

Consequently, the probability that it does not fall into the slot 1 is:

$$q = 1 - p = \left(1 - \frac{1}{S}\right) \quad (9)$$

Generalizing to N votes, we get a binomial distribution with parameters N and p .

Let X_k be: "Exactly k votes are stored in slot 1" with $k \in Z^+$

$$Pr(X_k) = \binom{N}{k} p^k q^{N-k} \quad (10)$$

$$Pr(X_k) = \binom{N}{k} \left(\frac{1}{S}\right)^k \left(1 - \frac{1}{S}\right)^{N-k} \quad (11)$$

Given that:

$$\lim_{x \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n = e \quad (12)$$

We can assure:

$$\begin{aligned} Pr(X_0) &= \left(1 - \frac{1}{S}\right)^N \approx e^{-\varepsilon} \\ Pr(X_1) &= N \frac{1}{S} \left(1 - \frac{1}{S}\right)^{N-1} \approx \varepsilon e^{-\varepsilon} \\ Pr(X_2) &= \frac{N(N-1)}{2} \left(\frac{1}{S}\right)^2 \left(1 - \frac{1}{S}\right)^{N-2} \approx \frac{1}{2} \varepsilon^2 e^{-\varepsilon} \end{aligned}$$

These probabilities also represent the expected number of votes in slot 1. It is obvious that the same reasoning can be applied to any slot. Therefore, it is possible to find the expected frequency.

Given that $\lim_{x \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n = e$, for $N = S = 1000$, $\varepsilon = 1$. Therefore:

$$Pr(X_0) = Pr(X_1) = e^{-1} \approx 0.3678 \quad (13)$$

Similarly, for $N = 500$, $S = 1000$, $\varepsilon = \frac{1}{2}$, in which case:

$$\begin{aligned} Pr(X_0) &= e^{-\frac{1}{2}} \approx 0.6065 \\ Pr(X_1) &= \frac{1}{2} e^{-\frac{1}{2}} \approx 0.3032 \end{aligned}$$

Let $E(k)$ be: #expected slots containing k votes. Its value is obtained as follows:

$$E(k) = Sp(X_k) \quad (14)$$

This fits together with the Poisson approximation stated in [15] :

$$E(\text{Poisson}(\lambda)) = \lambda \quad (15)$$

$$\lambda[X_0] = Se^{-\frac{N}{S}} \quad (16)$$

6

For $k = 1$:

$$\lambda[X_1] = \frac{(ne^{\frac{-r}{n}})}{k!} \left(\frac{r}{n}\right)^k = \frac{Se^{(-\varepsilon)}}{k!} \varepsilon = S\varepsilon e^{-\varepsilon} \quad (17)$$

For $k = 2$:

$$\lambda[X_2] = \frac{ne^{(\frac{-r}{n})}}{k!} \left(\frac{r}{n}\right)^k = \frac{Se^{-\varepsilon}}{2} \varepsilon^2 = S\varepsilon^2 e^{-\varepsilon} \quad (18)$$

The Poisson approximation improves its quality when $S \rightarrow \infty$ and $N \rightarrow \infty$. The previous formula is related to S . It is more interesting yet, to obtain a connection with the number of successful votes, ie for $k = 1$, $E(1)$ is divided into N and it is obtained:

$$\frac{S\varepsilon e^{-\varepsilon}}{N} = e^{-\varepsilon} \quad (19)$$

It is possible to generalize the approach to Q channels, with $Q > 1$. For example, for $S = N = 1000$:

$$s[1] = 1000e^{-1} \approx 368 \quad (20)$$

Consequently, for $Q = 1$:

$$Pr(\text{successfulvote}) \approx 0.36 \quad (21)$$

$$Pr(\text{lostvote}) \approx 1 - 0.36 = 0.64 \quad (22)$$

For $Q = 2$, one vote is lost if collides in the two channels:

$$Pr(\text{successfulvote}) = 1 - 0.39 \approx 0.61 \quad (23)$$

$$Pr(\text{lostvote}) = 0.64^2 \approx 0.39 \quad (24)$$

The same scheme is generalized $\forall Q > 2$. Thus it is obtained a formula to calculate the expected value of the variable *Percentage of Lost Votes*.

$$| PLV | = (1 - e^{-\frac{r}{s}})^q \quad (25)$$

2.1 Practical Verification of the Proposed Formula

Given formulas above, a simulator has been implemented which two main aims:

1. To verify the correctness of formulas.
2. To bear out that the approach of storing in parallel channels optimizes the results in terms of several variables which may be considered.

The simulator is implemented allowing the following inputs:

1. Total number of slots to implement (T).
2. Number of voters (N).
3. Quantity of parallel channels to implement (Q).
4. Quantity of election acts that will be simulated by session (R).

The simulator verifies that the total number of slots (T) is a multiple of quantity of parallel channel, because the quantity of slots in each channel (S) must be an integer number.

When the simulation is complete, the following information may be obtained:

1. Total of successful votes (SV).
2. Total of lost votes (LV).
3. Quantity of runs where at least one vote is lost (R)
4. Quantity of runs (Votings) without lost votes (RWL)
5. Quantity of runs (Votings) with lost votes (RLV)
6. Best case, that is to say, how many votes were lost in the most successful run (BC).
7. Worst case, that is to say, how many votes were lost in the less successful run (WC).

Therefore, we will observe the behaviour of the formula (25) based on the next ratio:

$$SPLV = \frac{LV}{SV + LV} \quad (26)$$

Table 1 shows the values that were obtained in different simulations and the difference between those and the analytical results obtained by application of equation (25). With this purpose, the following variables are introduced:

- FV : Values obtained by application of formula (25).
- SV : Values obtained by simulation.

Watching the values of Table 1, the difference between FV and SV remains at very low values. Specifically:

- The maximum one is 0,002452561.
- The minimum one is 2,80437E-07.

N	T	S	Q	FV	SV	$DIFFERENCE$
15	150	15	10	0,010185894	0,008466667	0,001719227
15	300	30	10	8,89424E-05	0	8,89424E-05
15	450	45	10	3,35005E-06	0	3,35005E-06
15	600	60	10	2,80437E-07	0	2,80437E-07
30	300	30	10	0,010185894	0,007733333	0,002452561
30	600	60	10	8,89424E-05	0	8,89424E-05
30	900	90	10	3,35005E-06	0	3,35005E-06
30	1200	120	10	2,80437E-07	0	2,80437E-07
60	600	60	10	0,010185894	0,0098	0,000385894
60	1200	120	10	8,89424E-05	0,00015	-6,10576E-05
60	1800	180	10	3,35005E-06	0	3,35005E-06
60	2400	240	10	2,80437E-07	0	2,80437E-07
120	1200	120	10	0,010185894	0,012025	-0,001839106
120	2400	240	10	8,89424E-05	0	8,89424E-05
120	3600	360	10	3,35005E-06	0	3,35005E-06
120	4800	480	10	2,80437E-07	0	2,80437E-07
240	2400	240	10	0,010185894	0,009129167	0,001056727
240	4800	480	10	8,89424E-05	0	8,89424E-05
240	7200	720	10	3,35005E-06	0	3,35005E-06
240	9600	960	10	2,80437E-07	0	2,80437E-07
360	3600	360	10	0,010185894	0,008391667	0,001794227
360	7200	720	10	8,89424E-05	1,11E-05	7,78313E-05
360	10800	1080	10	3,35005E-06	0	3,35005E-06
360	14400	1440	10	2,80437E-07	0	2,80437E-07
480	4800	480	10	0,010185894	0,00988125	0,000304644
480	9600	960	10	8,89424E-05	0	8,89424E-05
480	14400	1440	10	3,35005E-06	0	3,35005E-06
480	19200	1920	10	2,80437E-07	0	2,80437E-07

Table 1. Difference between FV and SV

- The average value is: 0,000227181

Another aspect which should be highlighted is that the formula (25) works better when $N < S$. As both values approach, the behaviour is worse. For example, Figure 3 shows the values of FV and SV with the following values for the parameters:

- $N = (7..15)$
- $T = 150$
- $Q = 10$
- $S = 15$

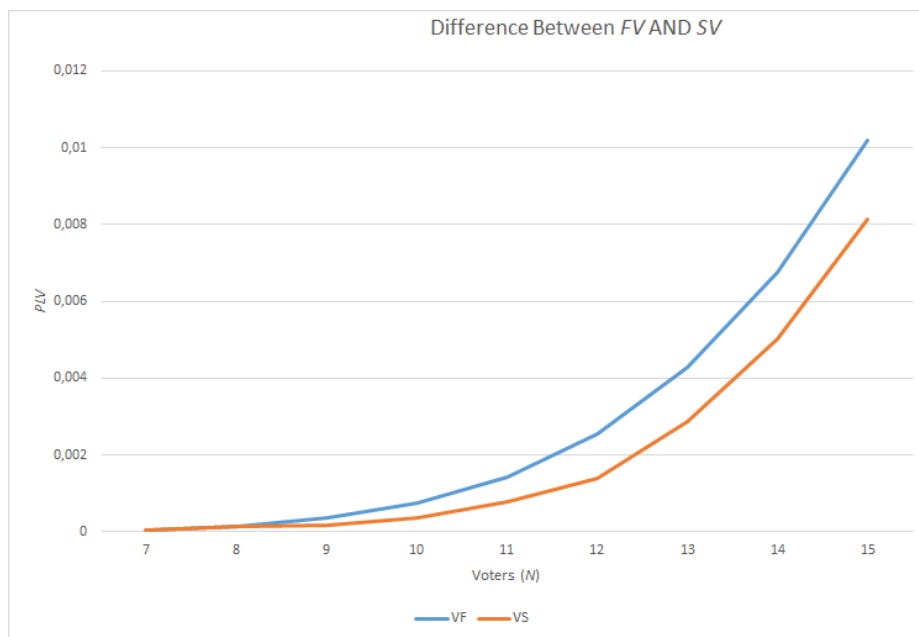


Fig. 3. PLV : Difference Between FV AND SV

3 Conclusions

The approach based on parallel channels optimizes the use of storage space intended to store data whose location is truly random. The formulas (1), (2), (3), (4) y (25) accurately describe the behaviour of the model.

Specifically, the results obtained by the formula (25) are very close to the values obtained in the simulations, though the difference increases when N is close to S . Even in that case, the behaviour of the formula is acceptable.

References

1. Uzal R., van de Graaf J., Montejano G., Riesco D., García P.: Inicio de la Línea de Investigación: Ingeniería de Software y Defensa Cibernética. WICC 2013. Ps. 769 - 773. ISBN: 9789872817961. (2013).
2. van de Graaf J., Montejano G., García P.: Manejo de Colisiones en un Protocolo Non Interactive Dining Cryptographers. JAIIO 2013. ISSN: 18502776. WSegI 2013. ISSN: 23139110. Ps. 29 a 43. (2013).
3. Chaum D.: The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability Journal of Cryptology. (1988).
4. van de Graaf J.: Anonymous One Time Broadcast Using Non Interactive Dining Cryptographer Nets with Applications to Voting Towards Trustworthy Elections. Ps 231-241. Springer-Verlag Berlin, Heidelberg. ISBN:9783642129797. (2010).
5. Flajolet P., Gardy D., Thimonier L.: Birthday paradox, coupon collectors, caching algorithms and self-organizing search. Discrete Applied Mathematics 39. Ps. 207-223. North-Holland. (1992).
6. García P., van de Graaf J., Hevia A., Viola A.: Beating the Birthday Paradox in Dining Cryptographer Networks. The third International Conference on Cryptology and Information Security in Latin America, Latincrypt 2014. September 17-19, 2014. Florianopolis, Brasil. Lecture Notes in Computer Science, Springer (2014).
7. García P., van de Graaf J., Montejano G., Bast S., Testa O.: Implementacin de Canales Paralelos en un Protocolo Non - Interactive Dining Cryptographers. JAIIO 2014. ISSN 18502776. WSegI 2014. ISSN: 23139110. (2014)
8. García P., Montejano G., Bast S.: Aspectos optimizables en un Protocolo Non-Interactive Dining Cryptographers. CONAIIISI 2014. ISSN: 23469927.(2014).
9. García P., Montejano G., Bast S, Fritz E.: Anonimato en Sistemas de Voto Electrónico: Últimos Avances. WICC 2016. ISBN: 9789506983772. (2016)
10. García P., van de Graaf J. Montejano G., Riesco D., Debnath N., Bast S.: Storage Optimization for Non - Interactive Dining Cryptographers (NIDC) Information Technology - New Generations (ITNG). Ps. 55 - 60. ISBN: 978-1-4799-8827-3. DOI: 10.1109/ITNG.2015.15. IEEE. (2015)
11. García P., Bast S., Fritz E., Montejano G., Riesco D., Debnath N.: A systematic method for choosing optimal parameters for storage in parallel channels of slots International Conference on Industrial Technology (ICIT)- Ps. 1700 - 1705. DOI:10.1109/ICIT.2016.7475019 IEEE - 2016
12. van de Graaf J., Montejano G., García P.: Optimización de un esquema Occupancy Problem orientado a E Voting. WICC 2013. Ps. 749 - 753. ISBN: 9789872817961. (2013).
13. Feller W.: An Introduction to Probability Theory and its Applications. Volmen I. Third Edition. John Wiley and Sons. New York. (1957).

Securizando la comunicación de un repositorio digital de contenido académico con Moodle

Francisco Javier Díaz¹, Alejandra Schiavoni¹, Ana Paola Amadeo¹, Duilio Ray¹

¹ Laboratorio de Investigación en Nuevas Tecnologías Informáticas - LINTI
Facultad de Informática – Universidad Nacional de La Plata
La Plata, Buenos Aires, Argentina

jdiaz@unlp.edu.ar, {ales,pamadeo}@info.unlp.edu.ar, dray@linti.unlp.edu.ar

Resumen. En la Facultad de Informática de la Universidad Nacional de La Plata se vienen usando un conjunto de plataformas que permiten sistematizar los procesos inherentes a la gestión académica. En el LINTI, Laboratorio de Investigación en Nuevas Tecnologías Informáticas, se está desarrollando un proyecto que consiste en la integración de un repositorio con diferentes herramientas y plataformas, como el LMS Moodle, un sistema de gestión de bibliotecas, sistemas de almacenamiento en la nube y redes sociales. La flexibilidad de uso de plataformas basadas en software libre permite la personalización, adaptación e incorporación de nuevos módulos en forma totalmente abierta. Sin embargo, al realizar una modificación es imprescindible tener en cuenta los aspectos de seguridad que pueden poner en riesgo la integridad de la plataforma. El presente artículo describe el análisis realizado a un módulo implementado para la comunicación entre Moodle y DSpace, con el objetivo de detectar posibles vulnerabilidades. El trabajo se realizó con el asesoramiento de expertos en seguridad informática del CERT-UNLP y se tuvieron en cuenta los aspectos de seguridad propios del LMS y del repositorio. Luego de analizar la implementación, se corrigieron los errores encontrados siguiendo las normas prescriptas.

1 Introducción

Las principales universidades de varios países han incorporado a su oferta educativa programas apoyados por e-learning para la formación profesional, haciendo uso de las TIC, que permiten flexibilidad en tiempo y espacio, para integrar a más gente en los procesos de enseñanza-aprendizaje. Estos programas consumen y generan contenidos en formato digital que pueden ser aprovechados por otros programas, sistemas u organizaciones con objetivos comunes. La incorporación de nuevas plataformas sirve para complementar la clase presencial, ampliando los límites y espacios temporales, promoviendo el aprendizaje ubicuo, que representa un nuevo paradigma educativo [1]. Con este fin, el sector educativo apuesta a la difusión y reutilización de objetos abiertos como elemento clave para la interoperabilidad y la concentración de recursos de forma estándar, compartida y organizada. Pensar en recursos educativos y en las licencias CC permite introducir el concepto de Recursos Educativos Abiertos (REA),

término que fue utilizado por primera vez por la UNESCO en el año 2002 y desde entonces, varias instituciones educativas de todo el mundo acuerdan con esta iniciativa. Por esta razón, el hecho de integrar los distintos sistemas de información intervinientes en el campo de la educación a distancia resulta de suma importancia y todo un desafío. Siguiendo esta línea, desde hace varios años se está desarrollando un proyecto en el LINTI que apunta a integrar distintas plataformas y herramientas que se utilizan dentro del ámbito académico, así como también la comunicación con las redes sociales ampliamente usadas hoy en día. El proyecto incluye no sólo la creación de un repositorio para albergar el contenido como recursos abiertos, sino también la implementación de módulos que extienden su funcionalidad básica permitiendo la comunicación con otras herramientas, la interacción con servicios externos y la integración con las redes sociales.

Para de la integración con el LMS Moodle, se implementó un módulo, cuyo objetivo fue facilitar la publicación del material generado por los alumnos a través de las entregas de los trabajos prácticos y trabajos finales. La posibilidad de contar con este método de publicación alienta a docentes a publicar dichos trabajos en uno o varios repositorios externos, ya que no necesitan conocer la interfaz y forma de acceso a ellos [2].

La adopción de plataformas basadas en software libre como DSpace y Moodle permite la personalización, adaptación de la interfaz y de las funciones provistas y la incorporación de nuevos módulos, en forma flexible y totalmente abierta. Sin embargo, al modificar una herramienta es imprescindible tener en cuenta en forma estricta aspectos de seguridad, que garanticen que no aparezcan vulnerabilidades que puedan poner en riesgo la integridad de la información.

En este artículo se describen en primera instancia las características y organización de las plataformas involucradas: LMS Moodle y el repositorio basado en DSpace. A continuación, se menciona el trabajo del CERT-UNLP, (Computer Emergency Response Team) que funciona en el ámbito de la Universidad Nacional de La Plata y está integrado por un grupo de investigadores que trabaja en temas de seguridad desde hace más de 5 años. Luego, se describe el módulo desarrollado, se detallan los errores detectados y la forma en que fueron solucionados para garantizar la seguridad integral de la plataforma.

2 Descripción general de las plataformas involucradas

La interconexión de sistemas es algo habitual en los servicios ofrecidos actualmente, sobre todo en aplicaciones basadas en Internet y ha tenido amplia repercusión en el ámbito académico.

Una de las consecuencias más evidentes de la influencia de las TIC en la educación superior está en la forma en la que se generan y en los medios por los que se transmiten los contenidos educativos, que en su mayor parte son en formato digital. Esta evolución en la forma de manipular la información por el público en general y por los alumnos durante el proceso de aprendizaje, motivó el estudio de la vinculación con el repositorio de las distintas herramientas que contienen material educativo.

2.1 Repositorio digital

La construcción de un repositorio digital, con material que sirve de complemento a la Biblioteca de la Facultad, incluye el análisis de la información a almacenar junto con los detalles de catalogación de la misma.

El repositorio se está implementando con la plataforma DSpace, que es de código abierto que provee herramientas para la administración de colecciones digitales, y comúnmente es usada para gestionar repositorios institucionales. En un principio, para la implementación del repositorio, los recursos a almacenar comprenderán proyectos realizados por los alumnos, durante el dictado de asignaturas de las diferentes carreras. Estos trabajos inicialmente se encuentran almacenados en la plataforma virtual de aprendizaje, a través de la cual son entregados y evaluados por el docente.

El proyecto comenzó con la integración del repositorio con la plataforma virtual de aprendizaje Moodle, de manera de poder establecer una comunicación bidireccional entre ambas plataformas. En primera instancia se llevó a cabo la comunicación para poder consultar y transferir elementos desde DSpace e incluirlos dentro del contenido de un curso [3]. En una segunda etapa, se implementó un módulo específico que permite publicar en forma semi-automática material generado por los alumnos. En las siguientes secciones se describe en forma general las características funcionales del módulo implementado y se detallan los ajustes que se debieron realizar para garantizar los aspectos de seguridad.

En etapas siguientes, se incorporaron las funciones de comunicación con servicios y herramientas existentes en Internet con el fin de potenciar las prestaciones propias del repositorio. Se realizó la comunicación con los servicios de gestión de archivos en la Nube, DropBox y Google Drive, que permiten guardar archivos en forma directa y con la red social Facebook, para compartir y recomendar un recurso determinado. La integración de esta funcionalidad dentro de DSpace intenta aprovechar estas prácticas ya habituales para que el contenido logre una mayor difusión [4].

2.2 Plataforma virtual de aprendizaje

La Facultad viene trabajando con la plataforma virtual Moodle para la gestión de cursos a través de la Web, desde hace aproximadamente 12 años como complemento de las clases presenciales de los cursos de las carreras de grado, para cursos de postgrado y de la Secretaría de Extensión [5]. La plataforma incluye más de 10000 usuarios registrados en alrededor de 170 cursos. La Fig. 1 muestra la evolución del número de cursos ofrecidos año a año y por categoría. Es importante mencionar que los cursos quedan activos de un año al otro, quedando vigentes durante 4 o 5 años sobre todo para los cursos relacionados con las materias de grado de la Facultad.

Un curso básico en Moodle está formado por recursos y actividades. A través de los recursos se incorporan contenidos como archivos en distintos formatos, páginas Web, un directorio, objetos de aprendizaje, entre otros. Por su parte las actividades crean una interacción entre alumnos y docentes, como cuestionarios, foros, entrega de tareas, etc. La conexión con las redes sociales también es un módulo desarrollado por

este grupo de trabajo, módulo Twitter [6], que se está utilizando cada vez más para comunicar alumnos y docentes.

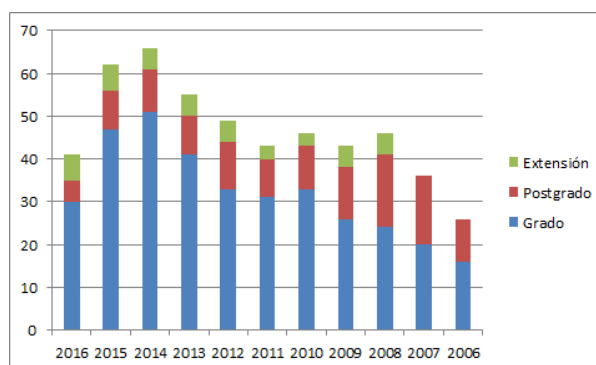


Fig. 1. Evolución del número de cursos por año

Dentro de las actividades, las Tareas es uno de los módulos más utilizados por la naturaleza práctica que caracterizan a las materias que integran el plan de estudios de la carrera. A través de las tareas los estudiantes suben sus trabajos que luego son evaluados por los docentes, quienes califican y plasman sus devoluciones a través de Moodle. Las devoluciones y la calificación son enviadas a cada alumno a través de su correo electrónico y sólo pueden ser vistas por él y los docentes a cargo de la materia. Esta forma de entrega resulta muy útil tanto para los alumnos como para los docentes, ya que los trabajos pueden ser subidos en forma on-line desde su hogar y en cualquier momento dentro del plazo establecido.

Como se mencionó anteriormente, algunos trabajos se desarrollan con un objetivo concreto y para ser utilizados en determinadas instituciones que los requieren. Analizando la cantidad de entregas totales realizadas por la plataforma en los últimos años, más de 2700, si fueran descartadas sería un desperdicio significativo, ya que podrían tener un muy alto potencial de reuso y re significación por la cátedra o por docentes de la misma u otras instituciones educativas.

3 Normas de seguridad y funcionamiento del CERT-UNLP

En el CeSPI funciona CERT-UNLP, el centro académico cuyo ámbito de aplicación es la Red de la Universidad Nacional de La Plata [7]. El mismo fue creado en el marco de la política de calidad del CeSPI, con el propósito de prevenir, detectar, analizar, investigar, registrar los incidentes de seguridad que son reportados. A partir de la creación de CERT-UNLP [8], los incidentes de seguridad se gestionan de una forma ordenada y sistemática acelerando la rapidez y eficiencia de las respuestas con el fin de minimizar la pérdida de la información y la calidad e interrupción del servicio. Dentro de los servicios que un CERT puede brindar, además del tratamiento de incidentes de seguridad, podemos detallar servicios reactivos: avisos y alertas; tratamiento

y análisis forenses y servicios proactivos: auditoría de red, auditoría de sistemas/aplicaciones, monitoreo, detección de intrusiones, evaluación de seguridad de redes y Pentest de aplicaciones Web (proceso certificado ISO9001:2008 desde el año 2012). Asimismo se prestan servicios de calidad en la seguridad como son la consultoría y la concientización en Seguridad de la Información.

A fin de cumplir con el objetivo mencionado, los miembros de CERT-UNLP participan de la comunidad de LAC-CSIRTs, la cual reúne CERTs de América Latina y el Caribe y de las capacitaciones y encuentros que organiza OWASP. En lo que respecta a la Seguridad Informática, el CeSPI desde el año 2007 brinda certificados digitales para e-ciencia a través de una PKI registrada en TACAR, reconocida por TAGPMA e IGTF [9]. Respecto a los Pentests de aplicaciones web, el CERT-UNLP realiza pruebas de penetración sobre aplicaciones Web desarrolladas en la organización. Estos chequeos, que permiten identificar problemas de seguridad, están basados en recomendaciones de OWASP y se realizan en la fase de prueba [10].

En los chequeos habituales, es posible agrupar las vulnerabilidades que se estudian y el Pentest analiza básicamente: *protección de sesiones de usuario*: problemas en el manejo de sesiones, fallas de inyección en formularios de autenticación, problemas en el manejo del cambio de contraseñas, etc.; *protección de información en Base de datos*: inyecciones SQL, problemas en el almacenamiento criptográfico, problemas de encriptación de las comunicaciones; *protección de los usuarios y sus transacciones*: CSRF, fallas de inyección XSS, problemas de encriptación de las comunicaciones, problemas de Phishing, problemas de DOS, etc.; *protección de la Aplicación*: referencia insegura de objetos, configuración de seguridad inadecuada, revelación innecesaria de información, inclusión de archivos local y remota, etc.; *protección del Servidor*: puertos abiertos y protocolos habilitados innecesarios, análisis de servicios y versiones utilizadas, existencia de política de firewall, problemas de DOS.

4 Características del módulo de comunicación con Moodle

La interacción entre DSpace y Moodle se llevó a cabo utilizando el protocolo SWORD, que permite el depósito de forma remota hacia un repositorio o sistema de información. DSpace incorpora el protocolo SWORD de dos formas: como un servidor compatible, disponible desde la versión 1.8 en adelante; o como cliente, para hacer que DSpace deposite ítems en otros sistemas que acepten este protocolo. En el lado de Moodle, se implementó un módulo que utiliza una librería cliente provista por SWORD. El módulo creado se encarga de preparar el recurso a transferir al repositorio incorporándole los metadatos en un formato estándar según información de contexto dentro de la plataforma. El diseño de la funcionalidad del módulo se basó en una metodología que permitió definir la estructura del recurso a publicar y los criterios de clasificación automática.

La posibilidad de contar con este método de publicación alienta a los docentes que gestionan sus cursos en la plataforma educativa, a publicar los trabajos entregados por sus alumnos en uno o varios repositorios externos, ya que no necesitan conocer la interfaz y forma de acceso a ellos. Para realizar esta tarea primero se le presenta al docente un formulario donde se listan la estructura del repositorio, de forma tal de

permitirle seleccionar en qué unidad se depositará el envío. Para obtener estas colecciones se usa el protocolo REST. Luego el docente selecciona los trabajos enviar, introduce su nombre de usuario y contraseña propios del repositorio, y realiza el envío, para esta comunicación se utiliza el protocolo de interoperabilidad SWORD.

5 Testeos y corrección de vulnerabilidades del módulo

5.1 Aspectos de seguridad de Moodle

Moodle provee herramientas para manejar la seguridad del módulo, esto es, asociar permisos a los diferentes usuarios del módulo (docentes, docentes del curso, estudiantes, etc.), y luego controlar los mismos. Si un usuario pertenece a un grupo va a poseer todos los permisos asociados a ese grupo. Para configurar los permisos se deben definir las *capabilities*. Estos permisos nos permiten definir acciones y los niveles de seguridad requeridos para cada acción. Para validar que un usuario tenga los permisos asociados se debe utilizar el método *has_capability()*, que recibe como segundo parámetro el contexto del módulo del curso, esto se debe a que los permisos definidos corresponden justamente a la instancia del módulo en un curso determinado, como se observa en la Fig. 2.

```
29
30 require_once(dirname(dirname(dirname(__FILE__))).'/config.php');
31 require_once(dirname(__FILE__).'/lib.php');
32
33 require_login();
34 $cmid = required_param('id', PARAM_INT);
35 if (!$cm = get_coursemodule_from_id('sword', $cmid)) {
36     error("Course module ID was incorrect");
37 }
38 $context = context_module::instance($cm->id);
39
40 if(has_capability('mod/sword:view',$context)){
41
```

Fig. 2 Se envía información de contexto para garantizar los permisos adecuados

5.2 Testeos y correcciones realizados en el módulo

Una vez finalizado el desarrollo del módulo, se solicitó una evaluación de seguridad al CERT-UNLP. El Pentest realizado permitió encontrar diversos errores. A continuación se detallan estos errores de seguridad, sus implicaciones y los cambios realizados con el fin de solucionar estos problemas:

1. *Ausencia de control de acceso a funciones*: este problema de seguridad permite que usuarios accedan a funcionalidad no autorizada para su perfil. Por ejemplo permite que usuarios autorizados accedan a funciones privadas, o también que usuarios anónimos accedan a funcionalidad propia de usuarios registrados.

Para evaluar si un sitio tiene esa vulnerabilidad se debe revisar la interfaz de cada posible usuario y verificar que no se muestren enlaces a funciones no autorizadas.

También se debe comprobar que se encuentren todos los chequeos de autenticación y autorización del lado del servidor. Estas verificaciones son fundamentales, ya que aunque el usuario no tenga un enlace directo a la funcionalidad, puede tratar de acceder a la misma a través de su URL.

En el caso del módulo, no se encontraban enlaces a funcionalidad no correspondiente al perfil de cada usuario, pero el problema que se presentaba era que no se validaba que el usuario estuviera autorizado a realizar la operación correspondiente en el servidor. Para solucionar esto se agregó código que verifica que el usuario tenga los permisos necesarios, por ejemplo las siguientes líneas de código muestran cómo se valida que un usuario tenga permiso para ejecutar una sección de código:

```
require_course_login($course);
require_login();
if(has_capability(
    'mod/sword:view', context_user::instance($USER->id)))
```

A su vez en los archivos que no son accedidos directamente en el flujo normal de un usuario, se agregan restricciones de seguridad para que no puedan ser operados directamente en caso de un ataque:

```
defined('MOODLE_INTERNAL') || die();
```

2. *Secuencia de Comandos en Sitios Cruzados (XSS)*: Esta vulnerabilidad se presenta siempre que no se sanitizen todas las entradas de datos al sistema, para asegurar que éstas no tengan código malicioso. El problema de no realizar la sanitización es que existe la posibilidad que los datos sean llevados a la vista del usuario, donde se ejecute el código malicioso lo cual podría, por ejemplo, robar información de sesión del mismo. Para solucionar esta vulnerabilidad se deben tener en cuenta varias cuestiones. Antes que nada se debe asegurar que todos los datos ingresados al sistema, por sistemas externos o por usuarios, sean correctamente sanitizados. La sanitización consiste en obviar los caracteres especiales, con esto se logra evitar que se ejecute código presente en los datos. En vez de ejecutarse, el código será presentado al usuario como texto plano. En el caso particular del repositorio, se recibe un JSON con datos los cuales se presentan al usuario. En la versión previa a las modificaciones realizadas, la entrada de datos no era sanitizada produciendo la vulnerabilidad descrita. Para solucionar esto se utiliza la función *htmlspecialchars()*, que escapa los caracteres especiales, con lo cual cualquier código malicioso es presentado como texto simple, como se muestra en la Fig. 3. Además de usar la función ya mencionada, se utilizó la función *strip_tags()*, como vemos a continuación, de forma tal que el JSON recibido sea sanitizado de una forma en la cual, una vez realizada esta acción no tuviera código de ningún tipo ni tampoco tags HTML.

```

3 require_once($CFG->libdir.'/weblib.php');
4 function sec_print($s) {
5     return htmlspecialchars(strip_tags($s), ENT_QUOTES);
6 }
7 function sec_print_array($arr){
8     foreach ($arr as &$act) {
9         if(!is_array($act)){
10             $act = sec_print($act);
11         }
12         else{ sec_print_array($act);}
13     }
14     return $arr;
15 }

```

Fig. 3 Cadena JSON sanitizado

3. *Redirecciones y reenvíos no validados*: esta vulnerabilidad consiste en que el sistema recibe una URL del usuario y no valida que la misma lleve a un sitio confiable. Para eliminar esta vulnerabilidad se debe:

- a. Inspeccionar el código para verificar que sólo se puedan realizar reenvíos o redirecciones a sitios confiables.
- b. Validar que los sitios supuestamente confiables no realizan ninguna redirección a un sitio no confiable. Para revisar esto se debe validar que al enviar un requerimiento HTTP a este sitio, no se reciba como respuesta un mensaje de HTTP que indique un reenvío (éstos son los que tienen código entre 300 y 307, el más usado para reenvíos es el que tiene código 302).

En el caso particular del módulo no se validaba que las direcciones utilizadas correspondieran a sitios confiables. Inicialmente la configuración que determina hacia qué sitio se intenta realizar un depósito se basaba en una URL ingresada por el usuario en un campo de texto. Se modificó el formulario para que el usuario, en vez de ingresar la URL manualmente, deba seleccionar entre un conjunto de direcciones correspondientes a sitios remotos ya verificados. Esta modificación además dificulta la posibilidad de un ataque XSS.

4. *Almacenamiento de contraseñas sin hashear*: El almacenamiento de contraseñas de una forma segura es un punto clave para la seguridad de un sistema informático. Si un atacante logra tener acceso a una base de datos y las claves no se encuentran hasheadas, el atacante podrá conocer la contraseña de cada usuario lo cual compromete la seguridad del sistema. Para solucionar este problema lo que corresponde es hashear las contraseñas antes de almacenarlas en la base de datos. En el caso particular del módulo se deben tener en cuenta otras consideraciones, ya que lo que se almacenan no son contraseñas de Moodle, sino que son contraseñas del repositorio con el cual el módulo se comunica. En base a esto se deben tener algunas consideraciones extras, por ejemplo en el caso de almacenar contraseñas hasheadas el algoritmo de encriptación debería ser el mismo en ambas plataformas. Además el repositorio debe estar configurado para validar correctamente el acceso al recibir contraseñas ya encriptadas. Esto significa un alto nivel de acoplamiento entre el módulo y el repositorio, lo cual supone un problema si se realizan cambios en alguno de ellos. En base a las consideraciones mencionadas se optó por no almacenar las

contraseñas y pedirle al usuario que la introduzca junto con su nombre de usuario en el repositorio cada vez que realiza un envío, tal como se muestra en la Fig. 4.

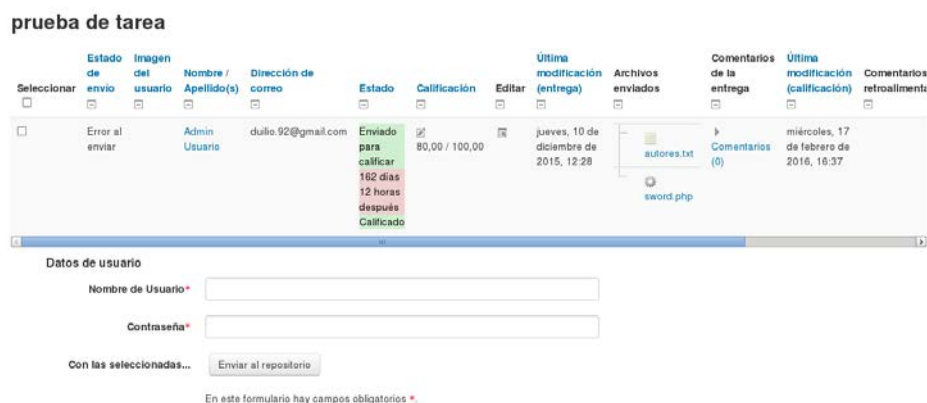


Fig. 4 Página de envío de Tareas de Moodle al repositorio DSpace.

5. *Comunicación insegura con el repositorio:* Al establecer comunicaciones en la web se debe tener en cuenta consideraciones de seguridad consecuentes a la información que se envía y recibe. En este sentido existen diversos protocolos los cuales afrontan los posibles ataques de seguridad en distinta medida. HTTP es vulnerable a ataques man-in-the-middle y eavesdropping. Un ataque de man-in-the-middle consiste en que un tercero intercepte, lea y modifique los mensajes entre dos partes sin que éstas se enteren. Por otro lado un ataque de eavesdropping consiste simplemente en que un atacante tenga la posibilidad de leer los paquetes enviados entre las dos partes. HTTPS está diseñado para resistir esos ataques, Para hacer esto, provee una comunicación más segura usando SSL para encriptar y desencriptar mensajes, con esto se hace mucho más resistente a los ataques mencionados anteriormente. SSL provee seguridad en dos cuestiones fundamentales: en primer lugar garantiza que la comunicación se está realizando con el servidor correcto, esto se verifica revisando un certificado que éste envía. En segundo lugar se asegura la confidencialidad de los datos dado que el servidor y el cliente intercambian mensajes encriptados, de esta manera en caso en que un atacante pudiera leerlos, inicialmente no podría entender su contenido ya que no conoce como desencriptarlos. Aunque es mucho más seguro que HTTP, un ataque man-in-the-middle sigue siendo posible en HTTPS. Esto se debe a que para iniciar una comunicación segura, el cliente y el servidor deben intercambiar claves, estas claves serán utilizadas para desencriptar los mensajes. La comunicación con el repositorio se hacía inicialmente usando el protocolo de interoperabilidad SWORD sobre el protocolo de comunicación HTTP. Tomando en cuenta la información sensible que se envía en estas comunicaciones (usuario y contraseña del repositorio) se decidió reemplazar el protocolo HTTP por el protocolo HTTPS, mejorando la seguridad de la comunicación del módulo con el repositorio.

Estas adecuaciones se llevaron a cabo en contacto con un representante del Cert-UNLP asignado al caso, para evacuar dudas y verificar procedimientos, en forma conjunta y efectiva.

6 Conclusiones

La integración entre plataformas se viene llevando a cabo desde hace tiempo en el marco de un proyecto global que se desarrolla en el LINTI. En una primera instancia a través de la comunicación del LMS Moodle con sistemas de gestión académica, redes sociales y repositorios digitales. En una segunda instancia, se avanzó en extender la funcionalidad del repositorio de manera de permitir su comunicación con otras herramientas muy utilizadas hoy en día. Las extensiones incorporadas tienen como objetivo ampliar el espacio propio del repositorio con el fin de integrarlo a plataformas de la nueva generación que van cambiando los hábitos de uso de la información. Tanto el repositorio como Moodle están basados en una plataforma de software libre, que permiten su personalización y adaptación a las necesidades de un proyecto concreto. Sin embargo, la adopción de herramientas abiertas y su modificación implica realizar un registro riguroso de las normas de seguridad para evitar vulnerabilidades y de esta manera poner en riesgo la información. El análisis se llevó a cabo en conjunto con el equipo de trabajo del CERT-UNLP, quienes evalúan las implementaciones que se realizan en el ámbito de la universidad y presentan un reporte exhaustivo de los errores que detectan. Este trabajo condujo la implementación del módulo de comunicación, hacia una versión ajustada que cumple con las normas de seguridad vigentes. Las correcciones realizadas permitieron evaluar exhaustivamente algunos aspectos claves, tales como los relacionados con la encriptación de claves y la organización de las distintas componentes del módulo para evitar que puedan ser accedidos en forma insegura. Por otro lado, en el repositorio se comenzó a usar SSL, norma impuesta según los requisitos del CERT, lo que llevó a realizar un cambio en la forma de comunicación. Los procesos de análisis, evaluación y corrección nos permitieron adaptar el módulo implementado a una versión segura utilizando tecnologías modernas que otorgan un mayor nivel de seguridad. Las modificaciones realizadas resultaban imprescindibles para la incorporación del módulo de comunicación en una plataforma de uso masivo como es el LMS Moodle que se encuentra en producción en la Facultad de Informática. La interacción con el equipo del CERT-UNLP nos permitió conocer y tener en cuenta determinados aspectos de seguridad que resultan cruciales en toda implementación y así poder establecer pautas y fijar metodologías de trabajo a seguir en los desarrollos que involucren las plataformas utilizadas en el proyecto de integración.

Agradecimientos

Queremos agradecer a todo el equipo de CERT-UNLP que realizó el análisis y testeo del módulo implementado y nos asistió en forma permanente, orientándonos en las correcciones necesarias.

Referencias

1. B. Cope y M. Kalantzis. Traducción Emilio Quintana. Aprendizaje Ubicuo. University of Illinois Press, 2009. 264 pp-
2. J. Díaz, A. Schiavoni, A. Osorio, P. Amadeo, E. Charnelli, "Integrating a Learning Management System with a Student Assignments Digital Repository. A Case Study", IADIS 2013, IADIS Multi Conference, Computer Science and Information Systems, e-Learning 2013, Praga, República Checa, 22 - 26 Julio, 2013.
3. J. Díaz, A. Schiavoni, P. Amadeo, E. Charnelli, "Diseño y construcción de objetos de aprendizaje. Su integración en repositorios y plataformas virtuales de aprendizaje", WICC 2012, XIV Workshop de Investigadores en Ciencias de la Computación, Posadas, Misiones, 26 y 27 de Abril, 2012.
4. J. Díaz, A. Schiavoni, P. Amadeo, E. Charnelli, J. Schulz, A. Humar. "Integrando un Repositorio Digital de Objetos de Aprendizaje con Servicios que Promuevan su Uso y Mantenimiento". LACLO 2014, IX Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje, Pág. 523-529, ISSN 1982-1611, Volumen 5, Open Access, Manizales, Colombia, 20-24 Octubre, 2014.
5. Cursos de Grado: <https://catedras.info.unlp.edu.ar>, Cursos de Postgrado: <http://postgrado.linti.unlp.edu.ar> y Cursos de Extensión: <http://cursos.linti.unlp.edu.ar>
6. Twitter Activity Module, https://github.com/mcharnelli/moodle-module_twitter
7. CERT-UNLP, Centro de Respuestas de Incidentes de Seguridad (CSIRT) Académico de la UNLP, <http://www.cespi.unlp.edu.ar/cert>
8. J. Díaz, P. Venosa, E. Lanfranco, N. Macia. "Definición e Implementación de un Centro de Atención de Incidentes (CERT) para un Ámbito Universitario". Anales de CACIC 2009, Universidad Nacional de Jujuy, octubre 5 al 9 de 2009 - ISBN 978-897-24068-4-1.
9. M. Dova, C. Grunfeld, F. Monticelli, M. Tripiana, A. Veiga, V. Ambrosi, A. Barbieri, J. Díaz, M. Luengo, N. Macia, L. Molinari, P. Venosa, M. Zabaljáuregui. "Progress of Grid Technology in Argentina: Lessons Learned from EELA". Anales de la III conferencia de EELA, Catania, pp.225-232. ISBN: 978-84-7834-565-6 http://www.eu-eela.org/3_conference/index.html
10. Open Web Application Security Project (OWASP): Top Ten Project: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project, Cheat Sheet Series: https://www.owasp.org/index.php/Cheat_Sheets, Testing Project: https://www.owasp.org/index.php/Category:OWASP_Testing_Project

Procedimiento para la detección empírica de Infraestructuras de Clave Pública anómalas o inseguras

Antonio Castro Lechter, Marcelo Cipriano, Eduardo Malvacio

{antonio.castrolechter; cipriano1.618; edumalvacio}@gmail.com.

Criptolab. Escuela Superior Técnica - Facultad de Ingeniería - Universidad Nacional de la Defensa (UNDEF) - Cabildo 15. A1406CCC – Ciudad Autónoma de Buenos Aires, Argentina

Abstract: El presente trabajo ofrece un procedimiento empírico susceptible de ser codificado para la creación de un software Auditor de una Infraestructura de Clave Pública (PKI). El mismo consiste en el cálculo de los parámetros estadísticos producidos por una PKI teórica – y por lo tanto libre de errores y defectos- y la obtención de los mismos parámetros de una PKI real. Así se podrá realizar el análisis comparativo de los mismos y determinar si dicho sistema se ajusta o no al comportamiento esperado, detectando sistemas defectuosos o inseguros.

Keywords: PKI, RSA, Certificados Digitales.

1 Introducción

Muchos son los usos y aplicaciones de la **Criptografía de Clave Pública o Asimétrica**. En ella podemos hallar el Sistema **RSA** que posibilita “confidencialidad” y “autenticación” a través de su esquema de cifrado y firma digital, respectivamente. Para poder llevar adelante estos servicios se puede montar una **Infraestructura de Clave Pública (PKI)** por sus siglas en inglés) en los servicios de redes y demás. Estos sistemas ¿podrían contener errores que permitan su vulnerabilidad? Confiados en su “buen funcionamiento” los usuarios podrían tener comprometida su seguridad y ser blanco fácil de ataques.

Para poder detectar comportamientos “no aceptables” de estos sistemas, el presente trabajo y sus antecedentes presentan un **Procedimiento Probabilístico-Estadístico para la Determinación Experimental de Infraestructuras de Clave Públicas Anómalas**. El mismo analiza el comportamiento de las **PKI's** de acuerdo a la distribución de los factores primos que cada “*certificado digital*” contiene en su “*módulo público*” m . El mismo es factible de ser codificado y poder así crear un **Software Auditor**.

En los párrafos 2 y 3 se tratará acerca de los conceptos generales de la **Criptografía de Clave Pública**, el sistema **RSA**, generalidades acerca de las **PKI**. También se presentarán algunas de los últimos descubrimientos acerca de debilidades e inseguridad de los mismos.

En el párrafo 4 se presenta la fórmula que permite calcular la *Función de Probabilidad* de no hallar/hallar colisiones o repeticiones de factores primos en muestras de certificados digitales. Dado que la misma requiere el cálculo de

factoriales muy grandes, se presentan también fórmulas de aproximación de los mismos que no tengan la misma complejidad computacional del procedimiento original.

En el párrafo 5 se presenta, finalmente, el procedimiento empírico para la detección de PKI, junto con las fórmulas que permiten calcular los valores de referencia probabilísticos y su proceso estadístico provenientes de la inferencia estadística.

2 Criptografía de Clave Pública.

2.1 Orígenes

Durante miles de años, conocidos como *La Era de la Criptografía Clásica*, sólo existía la Criptografía de *Clave Privada* o *Simétrica*: la misma clave, que se empleaba para cifrar un mensaje, debía usarse para descifrarlo. La clave debía ser compartida por el emisor y el receptor.

Esto, sin embargo, traía consigo un ineludible escollo: el “*Problema del Intercambio de Claves*” (PIC). En algún momento ambas entidades tuvieron que acordar la clave a utilizar. ¿cómo acordar la clave sin que la misma esté comprometida por usar un canal inseguro?

En el año 1976 **Whitfield Diffie** y **Martin Hellman** publican un trabajo en el que resuelven el PIC. Proponen lo que hoy se conoce como “**el Intercambio de Claves de Diffie-Hellman**” (DH) [1].

En 1977 **Ronald Rivest**, **Adi Shamir** y **Leonard Adleman**¹ publican un memo técnico ante el **Massachusetts Institute of Technology (MIT)**. En dicho trabajo presentan un sistema que permite cifrar/descifrar un mensaje eludiendo el PIC. A su vez el mismo principio permite la realización de un esquema de “*Firma Digital*”, adicionando el servicio de “*Autenticación*” al ya ofrecido servicio de “*Confidencialidad*”. Al año siguiente, 1978, publican el artículo por el que se conoce mundialmente al **Sistema RSA** [2].

La **Asociación de los Sistemas Informáticos ACM** (por sus siglas en inglés: **Association for Computing Machinery**), fundada en 1947, ofrece anualmente el máximo galardón en esta disciplina: el Premio Turing. Por sus importantes aportes a la criptografía en el año 2002 fue otorgado a Rivest, Shamir y **Adleman**. Y recientemente, en el año 2015, los galardonados fueron Diffie y Hellman [3].

Los trabajos fundacionales de *DH* y *RSA* dieron nacimiento a lo que se conoce como la “*Criptografía de Clave Pública*” (CCP) o “*Criptografía Asimétrica*”. Este tipo de Criptografía utiliza *Aritmética Modular*. Dentro de esta área de la Matemática, se abordan dos problemas sin solución hasta hoy².

¹ No se usa el orden alfabético de sus apellidos pues así lo quisieron sus autores. El sistema que propusieron fue bautizado RSA por las siglas de sus apellidos y en ese orden.

² El “Problema del Logaritmo Discreto” (PLD) para el esquema propuesto por Diffie y Hellman y el “Problema de la Factorización de Números” (PFN) para el esquema de Rivest, Shamir y Adleman.

2.2 Vulnerabilidades descubiertas recientemente.

Desde su nacimiento la Criptografía de Clave Pública ha sido blanco de ataques y la comunidad científica ha buscado e investigado sus vulnerabilidades. Hasta el presente, no se han hecho avances matemáticos significativos en ninguno de los problemas matemáticos que se encuentran detrás de los esquemas criptográficos mencionados. Y desde ese punto de vista, se puede considerar a la CCP como segura. Muchos de los problemas y debilidades hallados corresponden al nivel de la implementación. Entre las más recientes, en el año 2015, varias vulnerabilidades, conocidas como **LogJam** [4], han sido documentadas para DH³.

Lo mismo ocurre para el esquema **RSA**[5]. Debilidades a nivel hardware [6, 7] como a nivel software. En particular cabe mencionar la investigación llevada adelante por el **Dr. Lenstra** [8] el cual, junto a otros investigadores han evaluado más de un millón de certificados de clave pública y descubrieron que cerca del 5% de los mismos compartían factores primos. Esta cantidad de certificados, que tienen comprometida su seguridad ¿es un valor alcanzable dada la magnitud de la muestra analizada o está fuera de las posibilidades en vista del tamaño de los módulos analizados y la cantidad de primos posibles?

3 Infraestructura de Clave Pública.

3.1 Qué es una PKI.

En entornos y sistemas, tanto militares como del ámbito civil en redes Públicas o Privadas, Lan's, o Wan's como asimismo Internet, tienen amplia difusión las **Infraestructuras de Clave Pública (PKI)** por sus siglas en inglés: **Public Key Infrastructure**). Una **PKI** entrega, a sus usuarios, "*certificados*". Entre otras aplicaciones, con ellos se pueden realizar logueos y autenticación de usuarios, equipos y sistemas, cifrado y firma digital, no repudio, determinación de claves de sesión, etc. Los certificados que emite una **PKI**⁴, incluyen entre otros, un módulo m y un número e (generalmente 65537) conocidos como "*clave pública*" y un número d llamado "*clave privada*". El valor m , que tiene un tamaño t (medido en bits) se obtiene por el producto de 2 valores primos. Este trío (m, e, d) es calculado por la **PKI** al momento de solicitar el certificado digital correspondiente y entregado a un usuario que será su poseedor.

Una vulnerabilidad⁵ interna o propia de la **PKI** se produce si la misma manifiesta alguna anomalía al calcular los valores m .

³ Entre los consejos que los investigadores sugieren para resolver parte de este problema, se propone generar un grupo **Diffie-Hellman** de 2048 bits.

⁴ El X.509 es un estándar de la **Unión Internacional de Telecomunicaciones (ITU: International Telecommunication Union)**, por sus siglas en inglés) que estipula, entre otras cosas, los tipos y el formato de los certificados, el algoritmo para validarlos, etc.

⁵ La seguridad del **Sistema RSA** se basa en la dificultad de factorizar en tiempo aceptable, módulos m , (por ejemplo, $t=1024, 2048$ o 4096 bits como los empleados en la actualidad) y por ello preservar la clave secreta d . Con el conocimiento de uno de los factores primos de un

El conocimiento de esta vulnerabilidad permitiría a un “atacante” eludir la seguridad ofrecida por el algoritmo **RSA** y obtener sin dificultad las claves privadas comprometidas, pudiendo entonces acceder a la información que se pretende proteger o suplantar identidades de usuarios, etc.

3.2 Por qué se producen las vulnerabilidades.

Los sistemas actuales tienen un alto nivel de complejidad y detectar determinados tipos de errores no es fácil [9]. La lectura y fiscalización de todas las líneas de código que forman la **PKI** puede resultar ser una tarea ardua.

La detección de errores que provocan disminución de la seguridad en estos sistemas es una realidad con muchos antecedentes. Por ejemplo una vulnerabilidad expuesta en la emisión de certificados digitales en **OpenSSL** de **Debian** [10].

Estos “errores” ¿son sólo inocentes “bugs” que superaron las pruebas y se filtraron para ser detectados años después de su creación o fueron “plantados” con la intención de debilitar la seguridad?

4 Cálculo de la Función de Probabilidad de encontrar o no colisiones de factores primos.

4.1 Obtención de la fórmula.

Se presenta la fórmula que permite calcular la probabilidad que 2 o más certificados digitales tengan primos compartidos, dentro de una muestra de tamaño mu . Una versión más detallada de la obtención de esta función puede encontrarse en [11].

Sea t el tamaño, medido en bits, de los módulos públicos m . Sea b el tamaño en bits de los factores primos de m . Por ejemplo, si $t=1024$ entonces los valores primos tendrán el tamaño $b=512$ bits.

Sea P el conjunto de números primos de tamaño b .

$$P = \{p / p \text{ primo}; 2^{b-1} < p < 2^b\}. \quad (1)$$

El cardinal o cantidad de elementos de P -llamado aquí p - puede calcularse con una fórmula asociada al *Teorema de los Números Primos*⁶:

$$p = \text{Card}(P) \approx \pi(2^b) - \pi(2^{b-1}). \quad (2)$$

determinado módulo, se puede calcular de manera sencilla el otro factor y con él la clave d , previo paso, que se calcula sin dificultad.

⁶ Conjeturado por el matemático alemán **Carl Gauss** (1777-1855) y demostrado de forma independiente por el matemático belga **Charles-Jean de la Vallée Poussin** (1866-1962) y el matemático francés **Jacques Hadamard** (1865-1963).

$$p \approx \frac{2^b}{\ln 2^b} - \frac{2^{b-1}}{\ln 2^{b-1}} = \frac{2^{b-1}}{\ln 2} \left(\frac{2}{b} - \frac{1}{b-1} \right). \quad (3)$$

Sea M el conjunto de todos los módulos públicos que se pueden determinar a partir de los elementos del conjunto P :

$$M = \left\{ m / m = pq; p \neq q; p, q \in P \right\}. \quad (4)$$

El cardinal de M (que se indicará por m) es la cantidad de subconjuntos de 2 elementos del conjunto P , ya que cada módulo público es el producto de 2 valores primos y por la conmutatividad del producto, no importa el orden en el que se los multiplique.

$$m = \text{Card}(M) = \binom{p}{2} = \frac{p(p-1)}{2}. \quad (5)$$

Finalmente, sea R el conjunto de todas las muestras de mu módulos, en las que no hay colisiones de primos. Su cardinal queda determinado por:

$$\text{Card}(R) = \prod_{i=1}^{mu} m_i = \prod_{i=0}^{mu-1} \binom{p-2i}{2}. \quad (6)$$

$$\text{Card}(R) = \frac{\prod_{i=0}^{2(mu-1)} (p-i)}{2^{mu}}. \quad (7)$$

Luego

$$\text{Card}(R) = \frac{p!}{2^{mu} (p-2(mu-2))!}. \quad (8)$$

Para calcular la **Función de Probabilidad** se aplicará la definición clásica de probabilidad de **Laplace**⁷. Se asume como hipótesis que la **PKI** no almacena ni registra los primos que ya ha usado. Se tiene entonces:

$$p(R) = \frac{\text{card}(R)}{m^{mu}}. \quad (9)$$

$$p(R) = \frac{p!}{2^{mu} [p-2(mu-2)]!} \cdot \frac{1}{m^{mu}}. \quad (10)$$

Luego, operando convenientemente:

⁷ Definición laplaciana de probabilidad: casos favorables sobre casos totales.

$$p(R) = \frac{p!}{[p - 2(mu - 2)] [p(p-1)]^{mu}}. \quad (11)$$

Y su probabilidad complementaria sería:

$$p(\bar{R}) = 1 - \frac{p!}{[p - 2(mu - 2)] [p(p-1)]^{mu}}. \quad (12)$$

4.2 Cálculo de factoriales grandes

A continuación se exponen algunas fórmulas para aproximar el valor de factoriales grandes, como los que se usan en el cálculo de la función de probabilidad de las fórmulas (11) y (12). La complejidad computacional de la fórmula tradicional para la obtención de factoriales es demasiado elevada para que las fórmulas sean viables. Es por ello que la introducción de fórmulas de aproximación puede justificarse.

$$n! = e^{lnn!} \approx e^{n(ln-1)}. \quad (13)$$

$$n! \approx n^n e^{-n} \sqrt{2\pi n}. \quad (14)$$

$$n! \approx n^n e^{-n} \sqrt{\pi} \sqrt[6]{8n^3 + 4n^2 + n + \frac{1}{30}}. \quad (15)$$

Las fórmulas (13) y (14) se conocen como las fórmulas de **Stirling**⁸ y la fórmula (15) de **Ramanujan**⁹.

$$n! \approx \sqrt{2\pi} \left(\frac{n + \frac{1}{2}}{e} \right)^{n + \frac{1}{2}}. \quad (16)$$

$$n! \approx n^n e^{-n} \sqrt{\pi} \sqrt{2n + \frac{1}{3}}. \quad (17)$$

$$n! \approx n^n e^{-n} \sqrt{2\pi} \left(n + \frac{1}{6} + \frac{1}{72n} - \frac{31}{6480n^2} - \frac{139}{155520n^3} + \frac{9871}{6531840n^4} \right). \quad (18)$$

Estas últimas 3 expresiones son conocidas como las fórmulas de **Burnside**¹⁰, **Gosper**¹¹ y **Batir**¹², respectivamente.

⁸ **James Stirling** (1692-1770). Matemático escocés.

⁹ **Srinivasa Ramanujan** (1887-1920). Matemático indio. No dejó una demostración de su fórmula. Fue demostrada en 2000 por la matemática rusa **Ekatherina Karatsuba**.

¹⁰ **William Burnside** (1852-1927). Matemático inglés.

¹¹ **Ralph Gosper, Jr.** (1943-). Matemático y científico de computadoras estadounidense.

¹² **Necdet Batir** (1959 -). Matemático turco.

5 Algoritmo auditor de PKI's

5.1 Valores de referencia de la distribución binomial de una PKI libre de sesgos

La fórmula (11) permite calcular la probabilidad que en una muestra de certificados digitales de tamaño mu hayan al menos 2 de ellos que compartan un valor primo.

Habrán muestras con primos repetidos y en otras no. Esto representa un **Experimento de Bernoulli**. Esto es: ensayos o experimentos aleatorios e independientes entre sí, con dos resultados posibles y complementarios, llamados “éxitos” y “fracasos”.

Como consecuencia, se obtiene la conocida **Distribución Binomial**:

$$X \sim B(n, p(R)). \quad (19)$$

$$f(x) = \binom{n}{x} p(R)^x [1 - p(R)]^{n-x}. \quad (20)$$

Donde x es variable aleatoria discreta que representará la cantidad de éxitos/fracasos¹³ que se pueden hallar en n ensayos o muestras de tamaño mu . Y $p(R)$ es la probabilidad de no hallar/hallar colisiones de primos en una muestra de tamaño mu , tal como indica la fórmula (11).

Algunos de los parámetros de esta distribución de probabilidad son la *media* y la *varianza* (con la que se puede computar el *desvío estándar*, que es la raíz cuadrada de la *varianza*):

$$\bar{x} = np(R). \quad (21)$$

$$s^2 = np(R)[1 - p(R)]. \quad (22)$$

5.2 Detección experimental de Infraestructuras de Clave Pública Anómalas

Finalmente hemos arribado al objetivo principal de este trabajo y su línea de investigación portadora: la detección experimental de Infraestructuras de Clave Públicas anómalas por medio de su comportamiento empírico. Tal proceso es susceptible de ser automatizado. Así poder crear un “*algoritmo auditor de PKI's*”.

Se acepta como hipótesis del trabajo la existencia de “*permanencia estadística*”: el procedimiento experimental al que se someterá la **PKI** permitirá revelar su comportamiento desconocido a través de herramientas estadísticas.

Habría anomalía en el comportamiento de la **PKI** si se verifican divergencias entre los valores de referencia propuestos en este trabajo y los obtenidos por “*experiencia directa*”

¹³ Se puede optar por la probabilidad que en la muestra no haya primos repetidos o su complemento. Ambas calculadas en (11) y (12) respectivamente.

El procedimiento probabilístico-estadístico es el siguiente:

- 1- Determinación de los parámetros de evaluación de la **PKI**: tamaño m de los módulos y por consiguiente de b , el tamaño de sus factores primos.
- 2- Cálculo del valor p , según la fórmula (3).
- 3- Determinación del tamaño de cada muestra mu .
- 4- Cálculo¹⁴ del valor $p(R)$, según fórmula (11).
- 5- Cálculo de los parámetros, según fórmulas (21) y (22).
- 6- Solicitar a la **PKI certificados digitales** de manera que con ellos se los puedan agrupar en n muestras de tamaño mu .
- 7- Contar en cuántas de tales muestras no hay/hay primos repetidos (búsqueda de colisiones).
- 8- Con los valores obtenidos en el paso anterior, calcular los parámetros empíricos de la media y la varianza.
- 9- Comparar los valores teóricos predichos en el punto 5 con los obtenidos en el punto 8.
- 10- Determinar si la **PKI** manifiesta un comportamiento anómalo o no, de acuerdo a la comparación realizada en el punto 9.

Para poder evaluar si hay o no primos repetidos en una muestra hay varias alternativas. Si la **PKI** puede indicar los valores primos que utilizó en el cálculo del módulo público de cada certificado. Si no se puede tener información de los primos, se podría recurrir al procedimiento indicado en [12].

6 Conclusiones y futuros trabajos

Se han presentado las fórmulas que permiten calcular el comportamiento libre de sesgos y vulnerabilidades de una **PKI** ideal. Se ha presentado un procedimiento probabilístico-estadístico que permite determinar empíricamente el comportamiento real de una **PKI**.

La comparación entre ambos valores (el predicho teóricamente y el verificado experimentalmente) permite determinar el comportamiento anómalo o no de una determinada **Infraestructura de Clave Pública**.

Este proceso puede ser llevado a algún format de código y así crea un “**Software Auditor de Infraestructuras de Clave Públicas**”.

Futuros trabajos de investigación podrían probar el procedimiento presentado, como así también resolver algunos interrogantes que no se han abordado aún, entre otros:

- las fórmulas de aproximación para el cálculo de grandes factoriales ¿introducirán incertidumbres que interfieran en la determinación de las anomalías a detectar?

¹⁴ Se podrá optar por una de las fórmulas de aproximación de factoriales grandes, tal como se expone en el párrafo 4.2.

- ¿cuál es la mejor de ellas para aplicar en el procedimiento, dada la relación costo computacional/beneficio?
- ¿cuál es la brecha aceptable entre los parámetros teóricos y los empíricos?

7 Agradecimientos

El presente trabajo es parte de un **Proyecto de Desarrollo Tecnológico y Social (PDTs)** [13] de la **Escuela Superior Técnica -Facultad de Ingeniería-** perteneciente a la **Universidad Nacional de la Defensa (UNDEF)**.

El proyecto tiene como entidades adoptantes al **Instituto de Investigaciones Científicas y Técnicas para la Defensa (CITEDEF)** y al **Consejo Profesional de Ingeniería de Telecomunicaciones, Electrónica y Computación (COPITEC)**.

A ellos el agradecimiento de los autores.

8 Referencias

1. Diffie, W. y M.E.Hellman. "New directions in cryptography", IEEE Transactions on Information Theory 22 pp. 644-654. 1976.
2. R. Rivest, A. Shamir, L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems." Communications of the ACM, Vol. 21 (2), pp.120-126. 1978.
3. <http://amturing.acm.org/byyear.cfm> consultada el 20/6/2016.
4. Adrian, D.; Bhargavan, K.; Durumeric, Z.; Gaudry, P.; Green, M.; Halderman, J.; Heninger, N.; Springall, D.; Thomé, E.; Valenta, L.; VanderSloot, B.; Wustrow, E.; Zanella-Béguelin, S.; Zimmermann, P. "Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice." Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. Pages 5-17. ACM New York, NY, USA. 2015.
5. Boneh, D. Twenty Years of Attacks on the RSA Cryptosystem. Notices of the American Mathematical Society, Volume 46, Number 2. Providence, 1999.
6. Chen, S.; Wang, R.; Wang, X.; Zhang, K. "Side-Channel Leaks in Web Applications: a Reality Today, a Challenge Tomorrow". IEEE Symposium on Security & Privacy. Oakland, 2010.
7. Pellegrini, A.; Bertacco, V.; Austin, T. Fault-based attack of RSA authentication. Proceedings Design, Automation & Test in Europe Conference & Exhibition. The IEEE Council. Dresden, 2010.
8. Lenstra, A; Hughes, J; Augier, M y otros. Ron was wrong, Whit is right. e-print International Association for Cryptologic Research. 2012. <http://eprint.iacr.org/2012/064>.
9. Glass, Robert "Facts and Fallacies of Software Engineering". Addison-Wesley Professional, 2003.
10. Bello L, Bertacchini M. "Generador de Números Pseudo-Aleatorios Predecible en Debian". III Encuentro Internacional de Seguridad Informática. Manizales, Colombia. Octubre 2009.
11. Castro Lechtaler, Antonio; Cipriano, Marcelo; Malvacio, Eduardo. "Detección de Infraestructuras de Clave Pública anómalas". XXI Congreso Argentino de Ciencias de la Computación CACIC 2015. Junín, Buenos Aires. Octubre 2015.
12. Cipriano, M. "Factorización de N: recuperación de factores primos a partir de las claves pública y privada". XIV Congreso Argentino de Ciencias de la Computación. CACIC 2008. Chilecito, La Rioja. Octubre 2008.
13. <http://www.iese.edu.ar/investigacion.html#antecedentes> consultada el 20/6/2016.

Un enfoque para la detección de anomalías en el tráfico de red usando imágenes y técnicas de Computación de Alto Desempeño.

Mercedes Barrionuevo, Mariela Lopresti, Natalia Miranda, Fabiana Píccoli
LIDIC. Universidad Nacional de San Luis,
Ejército de los Andes 950 - 5700 - San Luis - Argentina
{mbarrio, omlopres, ncmiran, mpiccoli}@unsl.edu.ar

Abstract. La variedad y la complejidad del tráfico actual en Internet superan lo imaginado por los diseñadores de su arquitectura. Detectar posibles ataques en la red requiere contar con tecnologías para su clasificación, asociando flujos de datos con las aplicaciones que los generan. Uno de los desafíos actuales es trabajar con un conjunto de datos que crece a mayor velocidad que su capacidad de procesamiento. Utilizar y procesar imágenes para representar el tráfico de red a fin de detectar tráfico anómalo, tiene como ventajas no sólo contar con una herramienta de visualización de tráfico, sino también con las propiedades de las imágenes y su procesamiento: técnicas bien conocidas y naturaleza paralela de las computaciones. Este trabajo presenta la arquitectura de una herramienta eficiente para detectar anomalías en el tráfico de una red mediante un trabajo computacional sobre imágenes y aplicando técnicas de computación de alto desempeño, mostrando algunos resultados.

Palabras claves: Tráfico de Red. Anomalías. Procesamiento de Imágenes. SIFT. Computación de Alto Desempeño. GPUs.

1 Introducción

En la actualidad, Internet constituye uno de los principales actores de nuestra sociedad, juega un papel crucial en la educación, el entretenimiento, los negocios y la vida social. Este nuevo rol trae como consecuencia un rápido y sostenido tráfico en la red mundial, no sólo en cuanto al volumen de los datos sino también en la heterogeneidad y complejidad de los mismos. Varias son las causas del incremento del tráfico global, destacándose el constante crecimiento tanto de usuarios conectados como de dispositivos de Internet, la mayor velocidad de las conexiones, el creciente consumo de vídeo. Se estima para el año 2018 una generación de tráfico de alrededor de los 1,6 Zettabytes, producto de las conexiones fijas y móviles [8]. Teniendo en cuenta esto, realizar un análisis de tráfico en redes de gran escala resulta una tarea costosa y desafiante.

El tráfico circulante en Internet debe ser caracterizado con el fin de lograr una comprensión completa de su composición real, otorgando información detallada de las distintas actividades de gestión de red tales como la ingeniería de tráfico, diagnóstico de fallas, rendimiento de las aplicaciones, detección de anomalías, etc. La caracterización de tráfico tiene como principales obstáculos: la falta de una variedad de huellas, las cuales puedan servir como datos de prueba; la ausencia de objetos de flujo utilizados como referencia para su validación; y los inesperados problemas de congestión en la red central causado por distintas anomalías [8,9].

Detectar anomalías en redes de datos consiste en la identificación de patrones que se desvían del comportamiento normal de tráfico, por lo que está íntimamente relacionado con el modelado de tráfico. Con el fin de descubrir comportamientos anormales, se deben utilizar modelos de tráfico preciso y estable para describir lo que constituye un comportamiento de tráfico libre de anomalías. Este es de hecho un paso crítico en la detección de anomalías, ya que un modelo de tráfico incorrecto o inestable puede echar a perder completamente la capacidad de una correcta detección y causar un alto número de falsas alarmas.

Para modelar el tráfico de red de una forma entendible, varias de las investigaciones se han enfocado en el análisis de señales como ondas y en el modelado de tráfico como imágenes. Nuestro interés está puesto en este último dado que la visualización proporciona una capacidad de comprender enormes cantidades de datos, permitiendo percibir propiedades emergentes no anticipadas. Esto facilita la comprensión de las características tanto a gran como a pequeña escala de los datos, permitiendo revelar propiedades no sólo relacionada a los datos en sí, sino a la forma en la cual son recolectados [18].

En base a lo expresado anteriormente y teniendo en cuenta la alta demanda de recursos computacionales para producir un análisis eficiente del tráfico de red mediante imágenes produciendo alertas en tiempo real, es necesario analizar la posibilidad de aplicar técnicas de programación, las cuales permitan tomar ventajas de las características del procesamiento de imágenes como son el paralelismo natural que contiene. Por todo ello, considerar la inclusión de técnicas de computación de alto desempeño (HPC) en la solución, particularmente GPGPU (General Purpose Graphics Programming Unit) resulta ser una buena opción.

El objetivo de este trabajo es detectar posibles anomalías en el tráfico de una red de alta velocidad haciendo uso de una combinación de técnicas de análisis de tráfico de red, procesamiento de imágenes y computación de alto desempeño. Para cumplir con el objetivo se propone analizar los paquetes de red, generando a partir de los datos capturados imágenes, procesarlas buscando similitudes para encontrar anomalía en el tráfico de la red de interés. En cada tarea se pretende aplicar técnicas de HPC.

Este documento está organizado como sigue: la próxima sección describe los conceptos previos necesarios para los desarrollos. La sección 3 describe al sistema propuesto y sus módulos. En la sección 4 se muestra resultados experimentales del módulo analizador de Imágenes, y finalmente, se detallan las conclusiones y trabajos futuros.

2. Conceptos Básicos

El presente trabajo involucra diferentes conceptos, los cuales pertenecen a distintas áreas de la informática, podemos considerar tres grandes áreas: tráfico en las redes de computadoras y sus anomalías, imágenes y su procesamiento, y computación de alto desempeño. En esta sección describimos los conceptos básicos de cada una de las áreas involucrados en el desarrollo del presente trabajo..

2.1 Redes de Computadoras: Tráfico y Anomalías

En una red, las computadoras se comunican entre ellas a través de mensajes, los cuales son transportados por la red en paquetes de datos. Para que la comunicación se lleve a cabo,

las computadoras partícipes siguen un protocolo, el cual determina la forma en que los mensajes viajan en los paquetes y cómo éstos son armados.

Una secuencia unidireccional de paquetes con características comunes es denominada flujo de datos. Entre las características comunes consideradas en un flujo (f) de datos se destacan: las direcciones del origen (SRC_{IP}) y destino ($DEST_{IP}$), la identificación del protocolo a nivel de capa 3 (*Transport Protocol*) y los puertos origen (SRC_{Port}) y destino de la comunicación ($DEST_{Port}$), $f = \{SRC_{IP}, SRC_{Port}, DEST_{IP}, DEST_{Port}, Transport Protocol\}$. Por ejemplo los flujos:

$$f_{Skype} = \{10.0.0.1, 31215, 212.48.72.19, 80, UDP\}$$

$$f_{SMTP} = \{10.0.0.1, 2233, 13.29.10.199, 25, TCP\}$$

describen al flujo de datos de la aplicación Skype y del protocolo SMTP respectivamente.

En el tráfico de una red, co-existen distintos flujos, dependiendo de las comunicaciones que se establezcan, entre los cuales podemos encontrar tráfico normal o alteraciones del mismo. Detectar estas alteraciones se refiere a encontrar patrones no conformes con el comportamiento esperado [8]. Para lograrlo es necesario, primero, clasificar el tráfico de una red.

La clasificación del tráfico de red no sólo consiste en la identificación de datos, sino también en la extracción de la información de interés [18]. Si bien existen varios métodos de clasificación de tráfico, los cuatro principales son aquellos basados en puertos, payload, comportamiento de host y basado en características de flujos.

A continuación se da una breve descripción de cada uno de ellos.

- Basado en *puertos*: tienen en cuenta la asignación de puertos bien conocidos asignados por la Autoridad de Asignación de Números de Internet (IANA).
- Basados en *payload*: consisten en inspeccionar el contenido de los paquetes a nivel de transporte para identificar patrones característicos o para identificar secuencias relacionadas con el protocolo a nivel de aplicación, coincidente con un conjunto de reglas predefinidas.
- Basado en *comportamiento de host*: este enfoque considera el número de conexiones abiertas, los puertos usados y alguna combinación de ellas para trazar un perfil típico del host y compararlo con perfiles previamente almacenados.
- Basado en *características de flujo de paquetes*: este enfoque usa características de flujo tales como tamaño promedio de paquetes, tiempos entre arribos, duración de flujos, entre otras. La clasificación consiste en realizar una comparación estadística entre el tráfico desconocido y las reglas aprendidas previamente [15]. Este enfoque incluye principalmente técnicas de minería de datos y algoritmos de aprendizaje automático.

En base a varias evaluaciones existentes [2] se ha podido observar que no es posible proponer el mejor enfoque para la selección de un único algoritmo con un rendimiento óptimo para todos los escenarios posibles. Es por ello que es necesario pensar en métodos alternativos o combinaciones de métodos de diversas áreas para realizar la tarea de clasificación y análisis de tráfico de red.

Respecto a las anomalías, identificarlas, diagnosticarlas y tratarlas en el momento oportuno es fundamental para el administrador de red, debiendo estar entre las tareas principales de éste, ya que sin esta capacidad, las redes pueden funcionar de manera ineficiente y no confiable. Para una identificación precisa y un correcto diagnóstico es necesario contar con datos sólidos y buenos métodos.

Si consideramos que las anomalías de red son aquellas situaciones en que el tráfico de una red difiere notablemente de su perfil habitual, teniendo en cuenta que el tráfico, intrínsecamente, presenta una serie de variaciones producidas por la diversidad de usuarios, servicios y aplicaciones, así como de ciclos temporales diarios y semanales; el problema de

analizar el tráfico de una red y obtener información de él es un problema con muchas dimensiones, tornándose difícil y computacionalmente costoso.

Como mencionamos antes, la primera tarea es la recolección y procesamiento de los paquetes que viajan en la red. Esta tarea puede ser realizada de distintas maneras y a través de distintas herramientas, entre ellas se destacan:

- Bibliotecas como *libpcap* (www.tcpdump.org/), portable en C/C++ con las funciones necesarias para la captura de tráfico de red. Es utilizada por Tcpdump (www.tcpdump.org/), Snort (www.snort.org/), Wireshark (www.wireshark.org/) y Nmap (nmap.org/),
- Herramientas *NetFlow* (www.manageengine.com/products/netflow/) o JFlow (www.juniper.net) pertenecientes a Cisco y Juniper, respectivamente, usadas por Flow-Tools y Ntop.
- Formatos *NLANR* (www.caida.org/projects/nlanr/), como por ejemplo DAG, Coral y TSH, Time Stamped Headers.

Una vez recolectados los paquetes, se inspeccionan las cabeceras extrayendo, por ejemplo las direcciones IP del origen y destino, los números de puertos, el volumen de tráfico en bytes, el protocolo, entre otros. Qué datos extraer dependerá de lo que se desee analizar.

En la siguiente sección introducimos las características principales relacionadas a las imágenes y su procesamiento.

2.2 Características de las Imágenes y su procesamiento.

Usar imágenes para representar datos ayuda a mostrarlos de manera más sencilla, y obtener conclusiones de forma más fácil. En el ámbito del análisis de tráfico de red, un conjunto de paquetes de datos puede ser representado como una imagen. Esto hace que se puedan descifrar patrones de tráfico más rápidamente.

Teniendo en cuenta esto, representar el flujo de datos del tráfico de una red mediante imágenes implica dos tareas, la primera es obtener la imagen a partir de los datos recolectados, y el procesamiento de la imagen obtenida a fin de extraer información y encontrar evidencia de anomalías.

Para la primer tarea, es sabido que un píxel es la menor unidad homogénea en color que forma parte de una imagen digital, la codificación de imágenes se puede realizar escribiendo en forma sucesiva los bits correspondientes a cada píxel línea por línea, o escribiendo los valores de los píxeles en formato ASCII. Si la imagen simboliza el tráfico de una red, el significado de un píxel puede ser, según NetViewer[11], la comunicación entre un origen y un destino, siendo la intensidad del color la cantidad de tráfico originado entre ellos.

Otra de las características a tener en cuenta es el formato de la imagen digital con la cual se va a trabajar, éste debe ser simple, sencillo y eficiente.

Extraer información relevante de las imágenes digitales constituye un gran reto en el campo de visión artificial, más precisamente en la detección y reconocimiento de objetos[3,13,15]. Para la búsqueda de características distintivas de las imágenes se han utilizado características locales, las cuales han demostrado ser muy efectivas. La idea es determinar las características distintivas de la imagen, encontrando sus puntos significativos y una descripción discriminante. A partir de esta descripción se podrán establecer similitudes con otras imágenes. Dos objetos iguales en diferentes imágenes deben ser identificados independientemente de la posición y escala.

Para lograr esto, existen diferentes métodos, uno de los más ampliamente usados es el método SIFT (Scale Invariant Feature Transform) propuesto por Lowe en 1999 [14]. Este algoritmo extrae aquellas características invariantes a escala, rotación, modificación de iluminación, ruidos y pequeños cambios de perspectiva. En función de ellas, es posible comparar dos imágenes aplicando la diferencia euclidiana entre los respectivos descriptores.

La idea de este trabajo es tener imágenes representativas de las anomalías típicas en el tráfico de red, las cuales serán utilizadas para compararlas con las generadas por el tráfico actual. De la comparación de las imágenes, se determinará la presencia o no de alguna anomalía. Si el resultado de la comparación es un valor inferior a un umbral de tolerancia, el tráfico actual no presenta anomalías, caso contrario estaremos en presencia de una, pudiendo también establecer cuál. En la sección 3 se detalla el sistema propuesto.

2.3 Computación de Alto Desempeño usando GPUs

El bajo costo y marcado incremento del rendimiento en las aplicaciones, permitieron a las unidades de procesamiento gráfico GPU, su fácil incorporación al mercado, convirtiéndose en una parte integral de los sistemas actuales de computación. La GPU tiene una estructura altamente paralela y buen ancho de banda de memoria, ofreciendo aceleración a cualquier aplicación de gráficos [12].

Hoy en día el poder computacional de la GPU es utilizada para resolver cualquier tipo de problemas, no solo de naturaleza gráfica. Esto se conoce como GPGPU (Computación de Propósito General en GPU)[12, 17]. Cuando se usa la GPU como computadora paralela se tienen en cuenta el número de unidades de procesamiento, la estructura híbrida de memoria y los posibles hilos trabajando en paralelo.

El modelo de programación CUDA, desarrollado por NVIDIA y soportado a partir de la GeForce 8 Series, permite utilizar la GPU como una computadora altamente paralela para aplicaciones no gráficas [16,17], proporciona un entorno de programación de alto nivel con el lenguaje estándar C/C++. Define una arquitectura para la GPU como una unidad programable que actúa como un coprocesador para la CPU. Ésta tiene múltiples multiprocesadores de streaming (SMS), cada uno de ellos con varios procesadores escalares (SP).

CUDA tiene dos características principales: el trabajo paralelo a través de threads concurrentes y la jerarquía de memoria. El usuario provee un programa fuente, el cual incluye el código para la CPU y el de la GPU (kernel). Un programa CUDA se compone de varias fases a ejecutarse en la CPU o la GPU. En la GPU múltiples threads resuelven el problema en paralelo aprovechando la jerarquía de memoria.

En [5,6,7,10] utilizan la GPU, por las características mencionadas antes, con diferentes técnicas de análisis de datos masivos. Estas técnicas se han desarrollado para ámbitos tales como aprendizaje automático, búsqueda y ordenación, minería de datos, base de datos, entre otros.

El problema abordado en este trabajo se caracteriza por tener varias etapas, la mayoría de las cuales son aptas para resolverse usando técnicas de computación de alto desempeño y, en particular para GPU. El objetivo es acelerar la detección de anomalías en el tráfico de una red casi inmediata a su ocurrencia.

3. Sistema de Detección de Anomalías

Para detectar posibles anomalías en una red de computadoras, combinando técnicas de procesamiento de imágenes y computación de alto desempeño, se propone un sistema encargado de capturar el tráfico de la red, organizarlo en flujos de datos, generar las imágenes que los representan para luego analizarlas respecto a un repositorio de imágenes de anomalías preestablecidas. En la Figura 1 mostramos una representación gráfica del sistema.

De la arquitectura del sistema planteado, se puede distinguir claramente cuatro módulos cada uno con funciones bien diferenciadas. Las funciones y características de cada uno de los ellos son:

- **Módulo de Captura de tráfico:** tiene la función de capturar el tráfico circulante en la red y clasificarlo en flujos. Para la primer tarea, como se mencionó en la sección anterior existen varias herramientas, en nuestro caso seleccionamos la ampliamente utilizada herramienta Wireshark por su robustez.
Del gran conjunto de paquetes obtenidos, se seleccionan sólo aquellas características necesarias para el análisis. En este primer enfoque, y dada su relevancia, las características que consideramos son: direcciones IP de origen y destino, puertos origen y destino de la comunicación y protocolo a nivel de capa 3. A partir de estos datos se realiza la clasificación de los paquetes en flujos de datos, los cuales son la entrada del módulo de Generación de Imágenes.
- **Módulo de Generación de imágenes:** Este módulo toma como entrada los distintos flujos y los transforma en imágenes, salida del módulo.
Como se mencionó anteriormente, las imágenes generadas deben tener un formato simple, sencillo y eficiente, por dicha razón se seleccionó el formato *pgm*. Una imagen *pgm* (Portable Gray Map) se caracteriza por ser una representación en escala de grises. Es un formato muy fácil de entender y usar en programación, además de muy versátil. Una imagen está formada por sólo 3 líneas para el descriptor: tipo de formato (P2 si es formato ASCII o P5 si es binario), ancho y alto de la imagen, y niveles de grises (máximo 255). El resto son los píxeles. La Figura 2 ilustra una codificación y su imagen correspondiente.

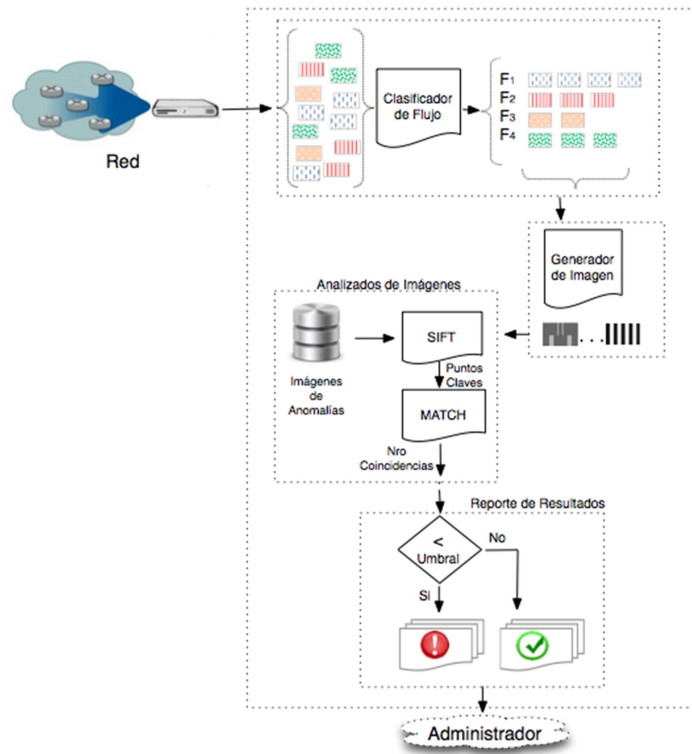


Figura 1. Estructura del Sistema de Detección de Anomalías Propuesto

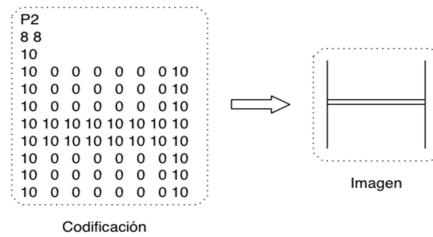


Figura 2. Código digital de una imagen PGM y la imagen correspondiente

Una vez logradas las imágenes, éstas son remitidas para su análisis al módulo siguiente.

- Módulo de Análisis de Imágenes:** A partir de las imágenes generadas, este módulo es responsable de encontrar similitudes entre ellas con cada una de las imágenes que representan anomalías. Las mismas conforman un repositorio en constante actualización. Esto se debe a diversos criterios o nuevas características de los ataques ya conocidos. Ejemplos de imágenes de anomalías son:

1. Líneas Horizontales: definirían una exploración desde la misma dirección IP de origen a múltiples direcciones IP destino.
2. Líneas Verticales: Representan un recorrido secuencial de varios equipos (en una subred) a una única dirección de destino. Se estaría llevando a cabo un probable DDoS (Denegación de Servicio Distribuido) a un nodo.

Como se mencionó en secciones previas, la comparación de imágenes debe ser capaz de encontrar similitudes entre imágenes, o partes de ellas, digitalmente diferentes. El algoritmo SIFT, tiene la propiedad de tomar como entrada dos imágenes y encontrar puntos característicos de cada una de ellas. Si los puntos claves de una imagen de la anomalía son identificados en la imagen de consulta, un posible ataque o anomalía sería detectado, caso contrario el tráfico actual no presentaría intrusiones conocidas.

En este módulo y como primer paso, proponemos aplicar técnicas de alto desempeño y GPU. Tanto el algoritmo SIFT como el analizador de coincidencias (MATCH) son realizadas en paralelo. Si bien existen varias implementaciones, nosotros utilizamos la propuesta por [1,4].

- **Módulo de Reporte de Resultados:** En base a los resultados del analizador, este módulo es el responsable de determinar si existe una posible anomalía y generar las advertencias correspondientes. La decisión se toma en base a la cantidad de coincidencias o puntos claves reportados por la etapa anterior, si éstas superan un umbral preestablecido, se estaría en presencia de una anomalía. Entre las advertencias, se encuentra la generación de mensajes de log del sistema y alarmas de red. Cada una con su propio formato, formas o métodos de reporte y frecuencia de alerta.

4. Resultados Experimentales

En este primer paso nos hemos enfocamos en el módulo de Análisis de Imágenes. Para ello hemos evaluado dicho módulo a través de varios experimentos.

Como las imágenes son generadas a partir de los datos del flujo que se desean analizar, el tamaño de ellas va a depender de la información considerada de cada uno de los paquetes capturados. Es por ello que los experimentos fueron direccionados a evaluar el comportamiento del módulo para imágenes de diferentes tamaños, los cuales oscilan entre los 50KB y los 20MB.

Para la evaluación, se consideraron las mejoras en el tiempo, respecto al algoritmo secuencial, para obtener los puntos claves de las imágenes y compararlos.

Los resultados del algoritmo SIFT secuencial fueron tomados un procesador AMD FX(tm)-6300 Six-Core Processor x 6 con Memoria de 7,8 GB y un disco de 80,5 GB. Para el SIFT Paralelo, la GPU es una Tesla K20c con 2496 procesadores CUDA, 4,6 GB de Memoria y Frecuencia de reloj del procesador y memoria 706 MHz y 2600 Mhz respectivamente.

En la Tabla 1 se detallan los segundos necesarios para obtener los puntos claves de una imagen mediante el algoritmo SIFT en sus dos versiones: secuencial y paralela.

Tamaño de Imagen entre	SIFT Secuencial	SIFT Paralelo
10KB y 300 KB	1,78092	0,014045
300KB y 1MB	4,650075	0,020615
1MB y 3MB	22,08463	0,0646675
3MB y 5MB	40,114555	0,09601

Tabla 1. Tiempos promedios en segundos del Sift Secuencial y Paralelo para distintos tamaños de imágenes.

La Tabla 2 muestra cuantos segundos en promedio se requieren para determinar si las imágenes comparadas tienen puntos claves en común (Match). Los tiempos se especifican también para las versiones secuencial y paralela.

Tamaño de Imagen entre	Match Secuencial	Match Paralelo
10KB y 300 KB	0,63709	0,004635
300KB y 1MB	3,325095	0,00805
1MB y 3MB	30,42166	0,026885
3MB y 5MB	85,51374	0,07112

Tabla 2. Tiempos promedios en segundos del Match Secuencial y Paralelo para distintos tamaños de imágenes.

La Figura 3 representa gráficamente la información de ambas tablas. Para todos los resultados se tomó el promedio de 10 ejecuciones para cada imagen considerada.

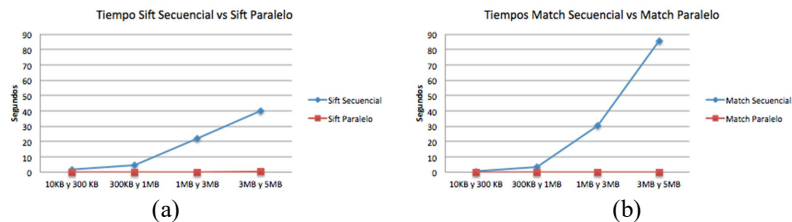


Figura 3. Tiempos en segundos de cada una de las etapas involucradas en el módulo de Análisis de Imágenes.

Como se puede ver en la Figura 3, tanto para el algoritmo SIFT(a) como para el proceso de Match(b), los tiempos de la solución paralela son sensiblemente inferiores a los de la solución secuencial para todos los tamaños de imágenes considerados, siendo mayor la diferencia en el Match. La reducción de los tiempos es mayor a los 100x.

5. Conclusiones y Trabajos Futuros

Existen varias herramientas cuya función principal es monitorizar y analizar tráfico de red, a fin de detectar posibles ataques. La captura y visualización de las tramas de datos puede ser muy útil o eficiente si además se tienen los paquetes de datos y la información del flujo de tráfico. Todo esto tendrá éxito si se puede detectar cualquier posible alteración y tomar buenas decisiones tan rápido como sea posible.

En este trabajo proponemos la arquitectura de un sistema para detectar anomalías en el tráfico de una red mediante el procesamiento de imágenes y técnicas de computación de alto desempeño. El análisis de las mejoras logradas considerando la aplicación de estas técnicas en una única etapa del proceso general, nos habilita para seguir trabajando en la misma dirección y lograr un sistema integralmente paralelo, el cual permitirá la detección de anomalías rápidamente a fin de lograr la prevención de ataques.

Entre los trabajos futuros, no sólo está la implementación de cada una de las demás etapas del sistema usando técnicas paralelas en el proceso, sino también evaluar y analizar la posibilidad de mejorar la existente. La propuesta realizada aplica técnicas paralelas al

proceso de resolver la comparación de dos imágenes, paralelismo intra-consulta. Una posibilidad es incluir computación paralela a nivel inter-consulta, es decir analizar al mismo tiempo la imagen de consulta con varias del repositorio.

Referencias

1. A. Acharya, R. Venkatesh Babu, "Speeding up SIFT using GPU," Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), 2013 Fourth National Conference. Pp1-4, 2013.
2. A. Callado, J. Kelner, D. Sadok, C. Kamienski, S. Fernandes. "Better network traffic identification through the independent combination of techniques". J. of Network and Computer Applications. Vol 33, N 4. Pp 433-446, 2010.
3. S. Belongie, J Malik, and J Puzicha. "Shape matching and object recognition using shape contexts". Pattern Analysis and Machine Intelligence (PAMI), IEEE Transactions on, pages 509–522, 2002.
4. M. Björkman, N. Bergström and D. Kragic, "Detecting, segmenting and tracking unknown objects using multi-label MRF inference". Comp. Vis. Image Underst. Pp 111-127. 2014.
5. S. Chen, J. Qin, Y. Xie, J. Zhao, P. Heng. "A Fast and Flexible Sorting Algorithm with CUDA". Algorithms and Architectures for Parallel Processing. LNCS Vol. 5574. Pp 281-290, 2009.
6. Y. Chen, Z. Qiao, S. Davis, H. Jiang, K. Li. "Pipelined Multi-GPU MapReduce for Big-Data Processing". Comp. and Inf. Science. Springer. Vol. 493. Pp 231-246, 2013.
7. D.Ciresan, A.Giusti, L.Gambardella, J. Schmidhuber. "Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks". MICCAI 2013. Pp 411-418, 2013.
8. Cisco Systems, "Global IP Traffic Forecast and Methodology, 2006-2011", white paper available at <http://www.cisco.com>, 2007 - updated 2008.
9. Cisco Systems, "The Exabyte Era", white paper. <http://www.cisco.com>, 2008.
10. S. Herrero-Lopez, "Accelerating SVMs by integrating GPUs into MapReduce clusters," Systems, Man, and Cybernetics. IEEE International Conference. Pp. 1298-1305, 2011
11. S. Kim, A. Reddy. "Image-Based Anomaly Detection Technique: Algorithm, Implementation and Effectiveness". IEEE Journal of Selected Areas in Communications 24. Pp 1942–1954, 2006.
12. D. Kirk and W. Hwu. "Programming Massively Parallel Processors, A Hands on Approach". Elsevier, Morgan Kaufmann, 2010.
13. D. Lowe. "Object recognition from local scale-invariant features". Computer Vision (ICCV), pages 1150–1157, 1999.
14. D. Lowe. "Distinctive image features from scale-invariant keypoints". International journal of computer vision. Pp 91-110, 2004.
15. K. Mikolajczyk, B. Leibe, and B. Schiele. "Local features for object class recognition". Computer Vision. 10th IEEE International Conference. Pp 1792–1799, 2005.
16. NVIDIA. "Nvidia cuda compute unified device architecture, C programming guide". Version 7.5. In NVIDIA, 2015.
17. J. Owens, M. Houston, D. Luebke, S. Green, J. Stone, and J. Phillips. GPU Computing. In IEEE, volume 96. Pp 879 – 899, 2008.
18. M.Thanigavel1, R. Mallika, T. Sujilatha. "An Open Platform for Internet Traffic Classification in TIE". IJRASET. Vol. 3 Issue VIII, 2015.

Modelo de Integración de Estándares para la Gestión de Identidad y Privacidad

Mag. Abogada. María del Carmen Becerra¹, Programador Universitario Pedro Zarate²,
Mag. Lic. María Claudia Gomez³

^{1,2} Instituto de Informática – Laboratorio de Análisis Forense e Informática Jurídica.
Departamento de Informática – FCFN-UNSJ

³ Departamento de Informática - Directora del Proyecto de Investigación “Representación genérica de modelos conceptuales en el campo de los Sistemas de Información” FCFN-UNSJ
mcbecerra2008@gmail.com,
pzarate@iinfo.unsj.edu.ar, cacugomez@yahoo.com.ar

Abstract. Este trabajo presenta un modelo para facilitar la gestión de la seguridad en el manejo de la identidad y privacidad. Se muestra la importancia y urgencia de este tema dado que ello es necesario y presta grandes beneficios para mejorar los servicios públicos. Se debe contar con estrictas medidas de seguridad basadas en la integración de normas, estándares y leyes más relevantes relacionadas con este tema, para mantener a salvo los datos personales. Se identifican el conjunto de prácticas y recursos de gestión, resultantes de la integración de estándares, y se discuten brevemente diversos aspectos involucrados en la formulación del modelo. El modelo propuesto permite establecer los pasos para que cualquier tipo de organización pública o privada, pueda realizar un manejo seguro de los datos sensibles de las personas.

Keywords: Identidad – Identidad Digital-Manejo de Identidad. Comparación de estándares. Modelo de Seguridad

1 Introducción

Con este trabajo se pretende brindar un modelo para que las organizaciones tanto públicas como privadas puedan gestionar la seguridad de la privacidad de los datos sensibles, con el fin de fortalecer la protección de datos conforme los estándares vigentes.

Se propone, caracterizar a la identidad digital como un derecho personalísimo y establecer su dimensión en una cuarta generación donde, la universalización del acceso a la tecnología, la libertad de expresión en la Red y la libre distribución de la información, juegan un papel fundamental [1], [2].

Estos derechos personalísimos, a la imagen, a la intimidad, al honor, a la reputación, a preservar la vida privada, así como cualquier otro del que resulte una emanación de la dignidad humana, están en los pactos internacionales y en nuestra Constitución. El ordenamiento positivo de nuestro país los ha tutelado, en diversas leyes como la N° 11.723, N° 24.766, N° 25.326, N° 25.506, N° 26.388 que reformó el Código Penal, y cobran especial relevancia después de la entrada en vigencia del nuevo Código Civil y Comercial, que recoge la doctrina y la jurisprudencia imperante[3].

Las tecnologías biométricas han sido puestas en uso sin las medidas adecuadas de seguridad para la información personal que es resguardada. Las organizaciones han incorporado en su estructuras organizacionales, tanto el registro de asistencia biométrico (huellas dactilares, registro facial), como los teléfonos corporativos con características biométricas (voz, patrón de la retina). Los riesgos se han incrementado y sofisticado y hay una demanda de mayor eficacia, que requiere respuestas de la tecnología [4].

Se plantea un modelo de integración, donde las normas y estándares a ser evaluados, respondan a un modelo general de valuación de los sistemas de información, como conjunto de elementos interrelacionados para lograr un objetivo específico.

La selección de estándares y normas se ha llevado a cabo utilizando el método de estudio de comparación de los mismos, se han creado plantillas para la comparación de las similitudes respecto a la gestión de la identidad y la privacidad.

El modelo propuesto, es una primera aproximación generada desde el proyecto referenciado, y está en proceso de contrastación.

2 Identidad Legal

La identidad se puede definir como conjunto de atributos y características que permiten individualizar a la persona en sociedad, pertenecientes a un individuo determinado, o compartidas por todos los miembros de una determinada categoría o grupo social. Según Rummens, el término proviene de la palabra francesa “identité” que tiene sus raíces lingüísticas en el sustantivo latino “identitas”, una derivación de “idem”, que significa “lo mismo” [5].

El Estado, debe garantizar el derecho a la identidad, poseer una identidad es un derecho humano de cualquier ciudadano y, al mismo tiempo, es parte de las obligaciones que un Estado tiene para con sus ciudadanos, en este sentido el acta de nacimiento es tanto el primer reconocimiento de la existencia de un individuo, como el primer instrumento legal con que cuenta una persona para demostrar su identidad.

En Argentina, la Ley 17.671 es la que establece la identificación, registro y clasificación del potencial humano nacional, ella establece que la identidad- incluye el nombre, el apellido, la fecha de nacimiento, el sexo y la nacionalidad del individuo. Además es la primera prueba de existencia de una persona como miembro de una sociedad, como parte de una nación, y es aquello que lo caracteriza y lo diferencia de los demás ciudadanos. Actualmente, el Decreto N°1766 del 2011, estableció un Sistema Federal de Identificación Biométrica, y tanto el D.N.I. como el pasaporte incorporan datos biométricos de las personas. La identidad la constituyen datos personales como el nombre, lugar de nacimiento, datos de filiación, nacionalidad, sexo, domicilio, fotografías, teléfono, correo electrónico.

La integran componentes físicos como: la huellas dactilares y plantares, el patrón del iris, el patrón de la retina, la forma de la mano, el patrón de las venas del dorso de la mano, la geometría de la mano, el rostro (registro facial), el reconocimiento vascular, las pulsaciones cardíacas, la identificación de la ondas cerebrales, ADN.

Los componentes de comportamiento como firma manuscrita, el análisis de los gestos, la forma de caminar, de gesticular. La voz se considera una mezcla entre componentes físicos y componentes de comportamiento. Existen elementos de identificación electrónica como el DNI, pasaporte, números de licencia y de seguridad social; números de tarjeta de crédito y de cuentas bancarias; nombres de usuario y contraseñas; incluyendo información financiera o médica, que contienen casi la totalidad de los datos que permiten identificar a una persona. Según la pertinencia corresponden a distintas áreas de las organizaciones, como muestra la Tabla 1.

Usos de datos personales		Forense	Legal	Gubernamental	Financiero	Salud	Inmigración
Aplicación en las Distintas áreas							
De cara al Ciudadano	ID Criminal	X	X	X			
	Verificación Identidad	X			X		X
	Vigilancia						X
	Voto			X			
	Salud			X			
De Cara al empleado	Remóvil	X	X	X	X	X	X
	Acceso físico	X					
	Acceso remoto	X					
	Sistemas	X					
	Redes	X					
	Peritaje	X					
De Cara al cliente y/o usuario	E-Commerce	X	X		X		
	Transacción			X	X		
	Punto de venta						
	Auditoria			X			

Tabla 1. Ejemplo de esquema de uso de datos personales por areas dentro de las organizaciones

3 Manejo de la Identidad Digital

Las tecnologías de la información han generado un nuevo concepto de identidad: la digital. El perfil de la "persona virtual" se define en la red y se nutre de los contenidos que la misma proporciona respecto de un determinado individuo o compañía: la web otorga contenido, identifica e individualiza a la persona de una u otra manera.

Así, la identidad digital, representa las mismas características y actividades, que la identidad real, pero llevadas a cabo en internet, como consecuencia del crecimiento de las comunicaciones digitales. Esta identidad es a la que generalmente se refiere como "vida virtual".

Otros autores la conceptualizan, como "El conjunto de rasgos y características particulares, que una persona expresa a través de internet, forma una parte inescindible de la identidad personal de cada sujeto, en su faz dinámica, y más precisamente en su aspecto psicológico, social y moral".

La identidad digital se construye conforme lo enfatizan otros autores [6] de forma activa, aportando textos, imágenes y vídeos a Internet, participando, en definitiva, del mundo web. En los sitios de redes sociales, se construye a partir de un perfil de usuario, que a menudo se enlaza a perfiles de otros usuarios o contactos. En la identidad digital convergen muchos aspectos de carácter sociológico, cultural e incluso psicológico [7].

Para Benantar [8], es una representación de una entidad activa en la computadora. Dicha entidad puede ser física (usuario, servidor u otro dispositivo) o software. La identidad esta asignada a un identificador el cual a su vez contiene atributos y derechos los cuales están referenciados a un perfil, un sistema de manejo de identidades pretende que la creación, asignación de derechos o negación de permisos de un perfil sea lo más sencillo posible. Por lo tanto se puede definir el manejo de la identidad como la administración de la misma, bajo estándares establecidos para que la seguridad de la información sea la correcta.

4 Comparación de Normas y Estándares para la Gestión de Seguridad de la identidad.

Se partió de, realizar una búsqueda en los documentos publicados hasta el momento vemos que el proceso de autenticación de identificación biométrica se basa en las normas ISO/IEC 17.799, 27.001 y 27.002 (estándares centrados en la seguridad de la Información, COBIT (estándares centrados en la gestión), ITIL Estándares centrados en los organismos públicos, y ANSI NIST-ITL 378 estándares centrados en la seguridad de datos biométricos.

Para la correcta administración de la seguridad de la información, se deben establecer y mantener acciones que busquen cumplir con los tres requerimientos de mayor importancia para la información, estos son:

- **Confidencialidad:** Busca prevenir el acceso no autorizado ya sea en forma intencional o no intencional a la información. La pérdida de la confidencialidad puede ocurrir de muchas maneras, como por ejemplo con la publicación intencional de información confidencial de la organización.
- **Integridad:** Busca asegurar que no se realicen modificaciones por personas no autorizadas a los datos o procesos, o que no se realicen modificaciones no autorizadas por personal autorizado a los datos o procesos, o que los datos sean consistentes tanto interna como externamente.
- **Disponibilidad:** Busca asegurar acceso confiable y oportuno a los datos o recursos para el personal apropiado. Diferentes organizaciones internacionales han definido estándares y normas que apoyan en diferente medida el cumplimiento de los requerimientos indicados anteriormente.

A continuación se detallan los estándares de mayor utilización a nivel mundial, y que fueron tomados como base para el modelo propuesto IRAM ISO/IEC 17.799. Esta norma internacional ofrece recomendaciones para realizar la gestión de la seguridad de la información dirigidas a los responsables de iniciar, implantar o mantener la

seguridad de una organización. En el punto 5.2 establece que la información tiene distintos grados de sensibilidad y criticidad. En el punto 9.23 administración de contraseñas de usuarios.

Las Normas IRAM ISO/IEC 27.001 y 27.002. Definen y documentan, los roles y responsabilidades de la seguridad de los empleados, contratistas y terceros en concordancia con la política de seguridad de la información de la organización. Garantizan la protección y privacidad de los datos según lo requieran las legislaciones y si fueran aplicables, las cláusulas relevantes contractuales.

Existen además, normas como la ISO 24.760 e ISO 29.100, que proporcionan un marco de referencia de alto nivel para la protección de los datos personales, y regulan la Gestión de identidad y Privacidad. Aportan a la gestión de privacidad las ISO 29134, e ISO 29151, ISO 29190/91 que presentan un modelo de evaluación de la capacidad en privacidad.

COBIT Acrónimo de “Control Objectives for Information and related Technology” (Objetivos de Control para la Información y Tecnologías Relacionadas), es un estándar desarrollado por la Information Systems Audit and Control Foundation (ISACA), la cual fue fundada en 1969 en EE.UU., y que se preocupa de temas como gobernabilidad, control, aseguramiento y auditorías para TIC.

Actualmente tiene más de 60.000 miembros en alrededor de 100 países. Esta organización realiza eventos y 239 conferencias, y desarrolla estándares en TI de gobierno, aseguramiento y seguridad, siendo COBIT el más importante. El rol de COBIT 5 en la estrategia de seguridad, objetivos de control, nuevo marco para la gobernanza en TIC’S, y guías de COBIT 5 sobre seguridad y riesgo.

En los últimos 5 años ha cobrado fuerza debido a que fue desarrollado en específico para el ámbito de las TIC. ITIL Acrónimo de “Information Technology Infrastructure Library”, ITIL es una norma de mejores prácticas para la administración de servicios de Tecnología de Información (TI), desarrollada a finales del año 1.980 por entidades públicas y privadas con el fin de considerar las mejores prácticas a nivel mundial. El organismo propietario de este marco de referencia de estándares es la Office of Government Commerce, una entidad independiente de la tesorería del gobierno británico. ITIL fue utilizado inicialmente como una guía para el gobierno de británico, pero es aplicable a cualquier tipo de organización.

Los Estándares ANSI/NIST-ITL son estándares que se aplican internacionalmente para proteger los datos biométricos. El propósito de esta norma es que un sistema biométrico dactilar pueda realizar procesos de verificación de identidad e identificación, empleando información biométrica proveniente de otros sistemas.

Normas y Estándares	ISO/IEC 17799	ISO/IEC 27001/02	ISACA-COBIT	ITIL	ANSI NIST/ITL (Ansi 378)
De cara al ciudadano	■	■	■	■	■
De cara al empleado	■	■	■	■	
De cara al cliente y/o usuario	■	■	■		

Tabla 2. Comparación de las características de los modelos de estándares.

El modelo se basó en la evaluación de normas y estándares, que proporcionan una base sólida para el cumplimiento de los objetivos de la Organización, en cuanto a la seguridad de los datos personales. En la Tabla 2, se compararon los modelos y normas presentadas anteriormente y se contrastaron sus puntos fuertes y débiles en cuanto a gestión de seguridad de la información, estableciendo que “■” significa una fortaleza del modelo/norma relacionada con las características evaluadas, mientras que un espacio en blanco significa que hay una debilidad. Así por Ej. Los modelos que tienen más fortalezas en la relación a la gestión de capacidad de servicio de TI, son el modelo ITIL [9] y las Normas ISO/IEC 27.001 y 27.002 permiten que la información y la tecnología relacionada se rijan y se gestionen de manera integral en toda la empresa [10].

5 Modelo de Integración de Estándares para la Gestión de Identidad y Privacidad

Las Organizaciones desarrollan un conjunto de actividades y procesos, que deben gestionarse sistemáticamente de tal forma que permitan el cumplimiento de sus objetivos [11].

En el modelo general se evalúan las normas y estándares de seguridad, Figura 1; en primer lugar se detectó el estado del arte de las normas y estándares, luego se las comparó en función del uso e impacto de cara al ciudadano, de cara al empleado y de cara al usuario.

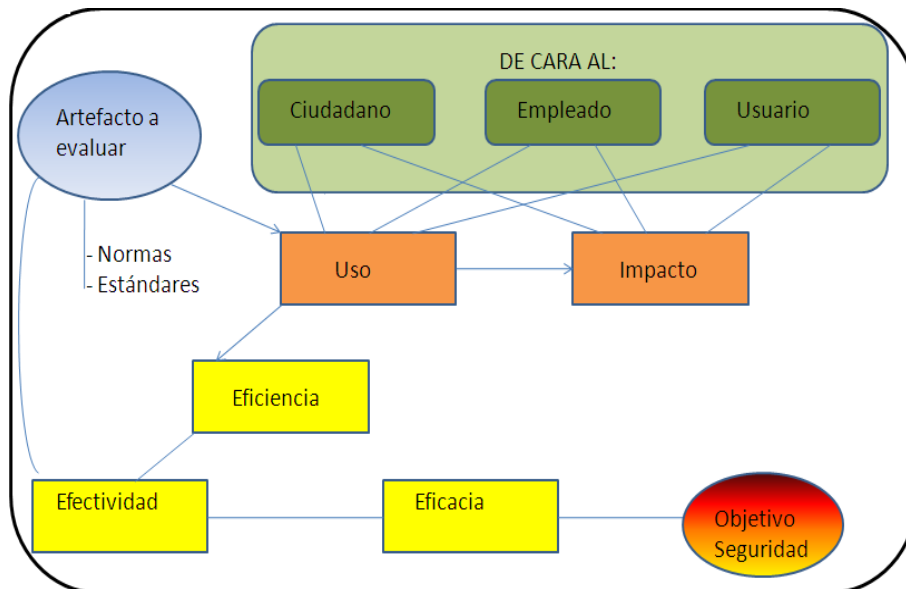


Fig. 1. Modelo General

Se midió el impacto desde estas tres perspectivas y la eficacia mediante el cumplimiento del principal objetivo que es la seguridad, este análisis se hará previo a su implementación, formulando criterios de evaluación que permitan ponderar su importancia en la protección de la privacidad [12].

La incorporación del Objetivo individual y organizacional y conceptos como el de efectividad que se basa en la eficacia (cumplimiento de Objetivos) y la Eficiencia (relación insumo/producto) pretende realizar una representación de Los sistemas de Información desde la perspectiva de su análisis cuantitativo y cualitativo.

De esta forma todos los artefactos TI, en la mayoría de los casos representados por proyectos de software[13], los que generalmente son evaluados mediante métricas específicas, requieren de una perspectiva desde las diferentes ciencias, lo que nos lleva a pensar que los SI necesitan de una visión interdisciplinaria[14].

6 Conclusiones

El aumento de la regulación y la legislación sobre la privacidad también está impactando en los entornos TI. La adopción de modelos y normas facilita la rápida ejecución de los buenos procedimientos y ayuda a evitar retrasos innecesarios en el desarrollo de nuevos enfoques. Todas las empresas tienen que adaptar el uso de modelos y establecer normas para ajustar sus requisitos individuales.

Cada organización debe establecer su propia estructura de gestión y recoger en todos ellos las recomendaciones que resulten más útiles. El uso de estándares ayuda al cumplimiento de las leyes, reglamentos, acuerdos contractuales y políticos y a ganar en ventajas competitivas sobre otras organizaciones [15].

Finalmente es muy importante, como con este modelo general para la gestión de la seguridad va a ser evaluado de cara al ciudadano, de cara al usuario y de cara al empleado, porque es fundamental medir la efectividad y la eficiencia, teniendo en cuenta la eficacia y como se usa e impacta sobre estas tres perspectivas. Pretende ser un aporte ante la preocupación internacional sobre la gestión de la identidad, tanto en el ámbito ciudadano como en el corporativo.

7 Referencias

- [1] Ossorio, Manuel, "Diccionario de Ciencia Jurídicas, Políticas y Sociales", Heliasta, 2005, Pág. 240.
- [2] <http://www.oei.es/revistactsi/numero1/bustamante.htm#1a>
- [3] Bueres Alberto, Código Civil y Comercial de la Nación analizado, comparado y concordado. 1ra Ed. Bs.As. Ed. Ammurabi. ISBN 978-950-741-680-4
- [4] Becerra. Navarro Mirta, Becerra, María del Carmen. Gestión Integral de Infraestructuras Críticas en las Organizaciones Locales alineados a las Normas IRAM ISSO 27.001 y 27002. WSI - II Workshop de seguridad informática CACIQ 2013
- [5] Borghello Cristian, Temperini Marcelo G. JAIIO. Suplantación de Identidad Digital como delito informático en Argentina.
- [6] La gestión de la identidad digital: Una nueva habilidad informacional y digital. BID. Universidad de Barcelona <http://bid.ub.edu/24/giones2.htm>
- [7] <http://blog.segu-info.com.ar/2012/07/como-se-construye-una-identidad-digital>.
- [8] Benantar, M. Access Control Systems-Security-Identity Management and trust Models New York:Springer.2006.
- [9] Alleinni Félix-Sánchez*, Jose Antonio Calvo-Manzano. Comparison of models and standards for implementing IT service capacity management. www.redalyc.org/pdf/430/43038629008.pdf
- [10] El derecho informática y la gestión de seguridad de la información una perspectiva con base a la norma Iso 27001. Revista del Derecho 2008. Biblioteca de Ciencia y Técnica de la Nación
- [11] Jacobson, I; Booch, G; Rumbaugh, J. El proceso unificado del desarrollo de software. Pearson 2000.
- [12] Object Management Group, Inc. OMG. <http://www.omg.org/2016>.
- [13] Gonzalo Perez-Tomé Estevez. Estudio sistemático de literatura de metodologías para la obtención de requisitos de privacidad. 2015. www.dit.upm.es/.../TFM
- [14] Diana M. Castillo Pinzon, DM. Enfoque para combinar e integrar la gestión de sistemas. 2010. ISBN 978-958-8585-06-2
- [15] Burgos Salazar, Jorge; Pedro G. Campos. Modelo para Seguridad de la Información en TIC. Web: ceur-ws.org/Vol-488/paper13.pdf



V Workshop de Innovación en Educación en Informática (WIEI)

Comité Científico:

Baldassarri Sandra (España)
Cukierman Diana (Canadá)
De Giusti Armando (Argentina)
Feierherd Guillermo (Argentina)
Manresa-Yee Cristina (España)
Marco Javier (España)
Motz Regina (Uruguay)
Pesado Patricia (Argentina)
Roig Vila Rosabel (España)
Simari Guillermo (Argentina)



V Workshop de Innovación en Educación en Informática (WIEI)

ID	Trabajo	Autores	
9100	Las Bibliotecas Digitales en el Currículum de las carreras de Ciencias de la Computación: una propuesta posible para Argentina y Venezuela	José Texier (UNdeC), Jusmeidy Zambrano (UNdeC), Fernanda Carmona.(UNdeC)	1189
9105	Herramienta didáctica para la enseñanza de contenidos y procedimientos en un curso universitario de Simulación de Eventos Discretos	Daño Weitz (UTN-FRRo)	1201
9108	El paradigma de la enseñanza por competencias y los principios del conectivismo: una experiencia práctica	Marisa Tumino (UAP), Juan Manuel Bournissen (UAP)	1211
9135	Sistema de Tutorías en Computación: Logros, dificultades y desafíos en el DCIC	Sonia V. Rueda (UNS), Luciano H. Tamargo (UNS)	1221
9164	Experiencia con Realidad Aumentada. Análisis de la incidencia en la comprensión de las estructuras de control	Natalí Salazar Mesía (UNLP), Gladys Gorga (UNLP), Cecilia Sanz (UNLP)	1231
9174	Estrategias Basadas en Robótica para Apoyar el Pensamiento Computacional	Maritza A. García (ITI Pascual Bravo), Claudia Deco (UNR), César Alberto Collazos Ordóñez(UniCAUCA)	1241
9203	Herramientas Educativas para la Enseñanza de la Lógica	Virginia Mauco (UNCPBA), Laura Felice (UNCPBA), Carmen Leonardi (UNCPBA)	1251
9210	Estilos de Aprendizaje y Rendimiento Académico de los Alumnos Ingresantes a la Lic. en Sistemas de la UNRN	Edith Lovos (UNRN), Tatiana Gibelli (UNRN)	1261



ID	Trabajo	Autores	
9217	Aplicación del enfoque descendente para la enseñanza de Redes de Computadoras, un estudio comparativo	Nelson R. Rodriguez, Daniel Arias Figueroa (unas), María Antonia Murazzo (UNSJ), Jorge N Mercado (UNSJ), Marcelo P. Moreno (UNSJ), Ernesto Sanchez (unas), Alvaro Gamarra (unas), Gustavo D. Gil (unas)	1270
9222	ALGOLIPSE: una herramienta educativa para mejorar la comprensión de algoritmos y estructuras de datos	Laura Fava(UNLP), Alejandra Schiavoni (UNLP), Jorge Rosso (UNLP), Facundo Falcone (UNLP), Lisandro Ronconi (UNLP)	1280
9243	Una paralelización del método de Householder	Osvaldo Mario Sposito (UNLaM), Gastón Procopio (UNLaM), Fabio Quintana (UNLaM), Hugo Emilio Ryckeboer (UNLaM)	1291
9251	Evaluación y Seguimiento Durante el Proceso de Enseñanza y Aprendizaje del Modelado UML en Entornos Colaborativos	Nicolás Battaglia (UAI), Carlos Neil (UAI), Dario Cardacci (UAI), Marcelo De Vincenzi (UAI), Roxana Martínez (UAI)	1301
9287	Herramientas lúdicas como apoyo a la enseñanza de la programación	Gladys N. Dapozo (UNNE), Cristina Greiner (UNNE), Raquel H. Petris (UNNE)	1311
9288	Scrum como herramienta metodológica en el Entrenamiento Cooperativo de la programación: Proceso Evaluativo	Lorena E. Del Moral Sachetti (UNSa), Ángel R, Barberis (UNSa)	1320
9297	Experiencia de enseñanza de la programación en un contexto de vulnerabilidad social	Francisco Bavera (UNRC), Sonia Permigiani (UNRC), Gaston Scilingo (UNRC), Maria Marta Novaira (UNRC), Cecilia Kilmurray (UNRC), Marcelo Arroyo (UNRC), César Cornejo (UNRC), Violeta Castresana (UNRC), Carmiña Verde (UNRC)	1329
9300	Integración de materiales didácticos en 3D para el Aprendizaje Basado en Problemas en la enseñanza de Artes	Mirta Fernández (UNNE), María Viviana Godoy Guglielmone(UNNE)	1338
9303	Construyendo Bases para la Formación Profesional en las Carreras de Ingeniería y Sistemas de la UNRN	Martin M. J. Goin (UNRN), Edith Lovos (UNRN)	1347



ID	Trabajo	Autores	
9305	Introduciendo conceptos de Cloud Computing utilizando el entorno CMRE	Laura De Giusti (UNLP), Franco Chichizola (UNLP), Sebastián Rodríguez Eguren (UNLP), Mariano Sanchez (UNLP), Juan Manuel Paniego (UNLP), Armando De Giusti (UNLP)	1357

Las Bibliotecas Digitales en el Currículum de las carreras de Ciencias de la Computación: una propuesta posible para Argentina y Venezuela

Jose Texier^{1,2}, Jusmeidy Zambrano^{1,2}, Fernanda Carmona²

¹ Universidad Nacional Experimental del Táchira, San Cristóbal-Táchira, Venezuela

² Universidad Nacional de Chilecito, Chilecito-La Rioja, Argentina

{jtexier, jzambrano, fbcarmona}@undec.edu.ar

Resumen. El Currículum en Bibliotecas Digitales (Curriculum DL) nace hace 10 años aproximadamente con el fin de homogeneizar la enseñanza y el perfil profesional del dominio LIS (Library and Information Science), aunado al incremento de sistemas y servicios de Bibliotecas Digitales (BD) y Repositorios Institucionales (RI). Las propuestas del Currículum DL están direccionadas a cursos de postgrados, sin vínculo directo con carreras de grado en Ciencias de la Computación. Se realizó una revisión de los planes de estudio y documentos de las carreras de Ciencias de la Computación en Argentina y Venezuela en los que no se hallaron cursos específicos en el dominio LIS. Por tanto y de acuerdo con este contexto, el objetivo de este trabajo es proponer un curso para estudiantes de grado de las carreras de Computación o afines, de esta manera se abre una oportunidad para conocer y profundizar en una de las disciplinas que hoy en día necesitan del dominio LIS y, a su vez se convierte en un posible campo de trabajo, en desarrollo de tesis de grados y en actividades de investigación.

Palabras clave: curriculum DL, propuesta, bibliotecas digitales, repositorios institucionales, ciencias de la computación.

1 Introducción

Las Bibliotecas Digitales (BD) o *Digital Library* (DL) han surgido gracias a la aparición, cada vez mayor, de un conjunto de sistemas y servicios de información académica y científica en formato digital. Surgieron a partir de la década de los noventa y, a principios del 2000 nacen los Repositorios Institucionales (RI) [1] con el mismo propósito de recopilar, catalogar, gestionar, acceder, difundir y preservar los recursos [1] del ámbito académico y científico. En la actualidad se han entrelazado los términos BD y RI [2] y su crecimiento se puede observar a través de los registros internacionales como OpenDOAR [3] y ROAR [4].

La importancia de las BD en las universidades ha cambiado los paradigmas de adquisición de información y acceso al conocimiento, como puede observarse en múltiples conferencias y revistas (Cf. *Joint Conference on Digital Libraries, JCDL*;

Theory and Practice of Digital Libraries, TPDL; la International Conference on Asian Digital Libraries, ICADL; y la IEEE/TCDL, Technical Committee on Digital Libraries) [5–9]. Morato [10] afirma que existe una baja demanda de profesionales de las Ciencias de la Información y Ciencias Documentales (LIS, *Library and Information Science*) a pesar del aumento del grado de especialización tecnológica, por lo que predicen un incremento en la demanda y necesidad de competencias tecnológicas/informáticas en el dominio LIS, tales como: procesamiento de información, indexación, búsquedas, almacenamiento, interoperabilidad, preservación, etc.

De manera tal que se han implementado algunos proyectos internacionales que vinculan LIS y las Ciencias de la Computación (CC) con el fin de desarrollar un Currículo DL¹, es decir, un plan de estudio que consiste en un esquema estructurado con el fin de aplicar contenidos, temas y problemas del área a través de procesos pedagógicos adecuados a las carreras de postgrado del dominio LIS.

Algunos avances realizados para fortalecer el Currículo DL son dos proyectos que han propuesto un modelo sistemático con profesionales LIS: el proyecto *Digital Libraries Curriculum Development* integrado por Virginia Tech junto con la Universidad de Carolina del Norte en Chapel Hill y el proyecto *Digital Curation Curriculum* (DigCcurr) de la Universidad de Carolina del Norte junto con la *National Archives and Records Administration* (con la colaboración de la Universidad de Toronto).

Asimismo, han surgido diversos postgrados a partir de: (a) la iniciativa de la Universidad de Illinois en Urbana-Champaign en colaboración con la Universidad de Indiana; (b) del fenómeno internacional de las *Information Schools* (iSchools²) creado en EE.UU; (c) de las propuestas curriculares en Europa favorecidas por la Convergencia de Bolonia; (d) del proceso de acreditación logrado por *American Library Association*³ (ALA) para carreras del dominio LIS fundamentado en la iniciativa iSchools, y, (e) de la conformación de paneles de especialistas trabajando en el tema.

A partir de tal revisión de la bibliografía se puede notar que todos los programas formativos (cursos, seminarios, talleres) son dirigidos a niveles de postgrado, ante lo cual este artículo propone un curso (optativo o electivo) para estudiantes de grado de las carreras de Computación o afines en dos universidades: Universidad Nacional de Chilecito en Argentina y Universidad Nacional Experimental del Táchira en Venezuela. ¿Por qué para esas universidades y en nivel de grado?. Primero, por la afiliación de los autores y posible viabilidad de la propuesta y, segundo y más relevante porque se considera que sería abrir una oportunidad para conocer y profundizar en una de las disciplinas que hoy en día necesitan del dominio LIS y, que a su vez se convierte en un posible campo de trabajo para estos egresados, para el desarrollo de tesis de grados y actividades de investigación, todo esto sobre la base de que la disciplina informática es transversal y necesaria en muchas áreas y esta no escapa de tal realidad.

¹ <http://curric.dlib.vt.edu/> y https://en.wikiversity.org/wiki/Digital_Libraries

² <http://ischools.org/>

³ <http://www.ala.org/>

2 Estado del arte

Existen variadas definiciones, algunas veces contradictorias, de lo que son las Bibliotecas Digitales (BD) y los Repositorios Institucionales (RI), en todo caso y más allá de mínimas particularidades, se puede concluir que un Repositorio Institucional es una Biblioteca Digital y una Biblioteca Digital es un Repositorio Institucional, puesto que actualmente ambos ofrecen servicios similares y el uso de cada término depende del ámbito donde se aplica y, por ende, de los recursos con los que se desean trabajar [1, 2]. Las BD y los RI son vistos como estructuras web interoperables que alojan recursos científicos, académicos y administrativos, tanto físicos como digitales, descritos por medio de un conjunto de datos específicos (metadatos). Tienen como propósito recopilar, catalogar, gestionar, acceder, difundir y preservar los recursos [1].

El auge de estos sistemas junto con los servicios informáticos que ofrecen han transformado la formación educativa y perfil profesional de los expertos en el dominio LIS, es decir, de las Ciencias de la Información y Ciencias Documentales con la implementación de un Currículum DL. A continuación se enumeran, *grosso modo*, algunas propuestas que se han desarrollado en diferentes universidades [11]:

- El principal proyecto en el ámbito de estudio curricular para la formación de bibliotecarios digitales es el *Digital Libraries Curriculum Development*, financiado por la National Science Foundation y puesto en práctica por un equipo multidisciplinar. Por el ámbito de la Biblioteconomía y la Documentación está la Universidad de Carolina del Norte (UNC) en Chapel Hill y al ámbito de las Ciencias de la Computación, lo representa el Virginia Tech (VT)⁴. Este proyecto describió los diez módulos que deberían formar parte de un Currículum DL modelo. Dichos módulos son las diez grandes materias o áreas temáticas principales, a saber: *Overview, Collection Development, Digital Objects, Info/Knowledge, Architecture, Spaces, Services, Archiving and Preservation Integrity, Project Management, DL Education & Research*. Según este grupo la recomendación es que se deben diseñar programas de asignaturas que conformen la formación integral de un especialista en DL [12].
- Otra iniciativa también en el ámbito norteamericano es el proyecto DigCcurr (*Digital Curation Curriculum*) [13], proyecto de análisis curricular financiado por el IMLS (*Institute of Museums and Library Services*) en el que participan: la escuela de Biblioteconomía y Documentación de la Universidad de Carolina del Norte y el NARA (*National Archives and Records Administration*), con la colaboración de la Universidad de Toronto, Canadá⁵. DigCcurr se enfoca en la necesidad de una formación profesional más extensa para los “*Digital Curation*” (Curadores Digitales). El curador digital, es un perfil profesional aún más especializado pero cercano al perfil de bibliotecario digital. En esta propuesta se desarrolla un marco curricular a

⁴ <http://curric.dlib.vt.edu>

⁵ <http://www.ils.unc.edu/digccurr>

nivel de postgrado ante la necesidad de tener especialistas y no profesionales básicos. Estos especialistas deben prepararse para trabajar en sistemas del dominio que abarcan sectores patrimoniales, educativos, comerciales, gubernamentales y científicos, entre otros. DigCcurr se desarrolló en dos fases, la primera de 2006-2009, con un listado de funciones que debe cumplir un curador digital, y la segunda fase de este proyecto (DigCcurr II, 2008-2012) tuvo como objetivo desarrollar un currículum internacional a nivel de doctorado [13].

- Otra propuesta es la presentada por la escuela de Bibliotecología de la Universidad de Illinois en Urbana-Champaign en colaboración con la Universidad de Indiana [14]. Es un proyecto de desarrollo curricular (financiado por el IMLS) para la formación de expertos tanto en el ámbito de las bibliotecas digitales como de la curaduría digital, el proyecto lleva por nombre *Data Curation Education Program* (DCEP)⁶.

Estos proyectos de desarrollo curricular sentaron las bases e intercambiaron sus experiencias en eventos, talleres, workshops. Por ejemplo, el taller de *Coordinating EU-US Digital Library Education* en el 2008 se integró con el Espacio Europeo de Educación Superior (EEES) de la Unión Europea (Convergencia de Bolonia), la idea fue poder homogeneizar las currículas según las necesidades europeas adaptadas al sistema de créditos ECTS (*European Credit Transfer System*) y del Suplemento Europeo al Título [11]. El otro evento destacado fue la conferencia de la *Association for Information Science and Technology* (ASIS&T) celebrada en Vancouver [15] con el fin de buscar puntos de convergencia y/o divergencia entre todas las aproximaciones curriculares para analizar las competencias de los profesionales relacionados con el área.

Por tal razón, en este artículo se ha planteado la siguiente interrogante **¿cuál es la relación entre el Currículum DL y las carreras de las Ciencias de la Computación?** Si bien los estudios relevados confirman que las iniciativas y propuestas del Currículum DL están direccionados a cursos de postgrados [16–23], ya que ofrecen la oportunidad de ampliar el conocimiento en el área con base en las carreras Ciencias Documentales y Ciencias de la Información, se infiere que estos programas de educación y formación son insuficientes para satisfacer la demanda porque, entre otras razones, exige que cada profesional sea cada vez más especializado en “todo”. Por ello, esta propuesta apunta a carreras de grado afines a las Ciencias de la Computación para no esperar la especialización en un campo, sino abrir oportunidades a los estudiantes para que conozcan el dominio LIS, fortalezcan el dominio y accedan a un campo profesional que tiene altas demandas en la actualidad [10]. Si los alumnos de carreras como Ciencias de la Computación o afines adquieren algunos conocimientos sobre el dominio LIS seguramente podrán generar trabajos interdisciplinarios con estudiantes de Ciencias de la Documentación y Ciencias de la Información entre otros aportes a este dominio.

⁶ <https://www.ideals.illinois.edu/handle/2142/877>

3 Contexto de la propuesta

Para proponer en cualquier institución de educación superior -particularmente, en carreras de Ciencias de la Computación- la inclusión de una asignatura, seminario o curso relacionado con LIS significa un desafío importante que acarrea toda una gestión académica al interior de las instituciones. De tal manera que como el propósito es hacer una propuesta se consideró imprescindible contextualizar en dos universidades: la Universidad Nacional de Chilecito⁷ (UNdeC) en Argentina y la Universidad Nacional Experimental del Táchira⁸ (UNET) en Venezuela. Instituciones que cuentan con carreras de Ciencias de la Computación y que son además lugares de trabajo de algunos de los autores de este trabajo por lo que pudiere ser viable la propuesta.

Antes de proponer cualquier asignatura o similar se consideró revisar cómo se estructuran las propuestas curriculares de estas carreras en ambos países. Tanto en Argentina como en Venezuela tienen diferentes nombres y planes de estudio las carreras relacionadas con las Ciencias de la Computación. En el caso de Argentina, la Resolución 786/2009 del Ministerio de Educación de la Nación dispone los lineamientos para la acreditación de carreras de informática [24], es decir, fija los contenidos mínimos y permite a las instituciones agregar otros contenidos que crean convenientes. En el caso de Venezuela no hay una disposición estatal sobre la disciplina específica, pero sí existen lineamientos básicos en las ingenierías y licenciaturas como tal (ver Tabla 1). Asimismo, en Venezuela los diversos programas de formación conducentes a títulos de grado en Ciencias de la Computación no están en alguna resolución que agrupe cada una de las ofertas curriculares [25, 26]. El Ministerio del Poder Popular para la Educación Universitaria, Ciencia y Tecnología (MPPEUCT) define ocho (8) áreas de conocimiento a través de su libro de oportunidades de estudio, una de esas áreas se llama “Ingeniería, Arquitectura y Tecnología” [27].

La Tabla 1 compara el estado de la situación (títulos de grado y contenidos curriculares básicos) de las dos carreras y una relación de los contenidos mínimos declarados en los planes de estudio de las CC de esos países afines con DL.

La revisión de los planes de estudio y las reglamentaciones del Estado para ambos casos y la vinculación con LIS arrojó que no se contemplan cursos específicos del ese dominio en forma directa, pero sí indirectamente a través de unidades curriculares como: sistemas de información, interoperabilidad, seguridad, almacenamiento, búsqueda de información, digitalización, multimedia, ontologías, arquitecturas de computadores y redes, preservación, visualización, entre otros. En todo caso debe resaltarse que tales temas curriculares en CC están enfocados, como es de esperarse, al área informática.

⁷ <http://www.undec.edu.ar/ingenieria-en-sistemas/#tabpanel3>

⁸ <http://docencia.unet.edu.ve/Departamentos/DptoInfo.php?ac=7>

Tabla 1. Lineamientos básicos gubernamentales en las ingenierías y licenciaturas.

	UNdeC - ARGENTINA (según Resolución 786/2009) [24]	UNET - VENEZUELA (según Lineamientos Ministerio) [27]
Carreras	<ul style="list-style-type: none"> - Licenciado en Ciencias de la Computación - Licenciado en Sistemas / Sistemas de Información / Análisis de Sistemas - Licenciado en Informática - Ingeniero en Computación - Ingeniero en Informática / Sistemas de Información 	<ul style="list-style-type: none"> - Licenciatura en Computación - Licenciatura en Informática - Ingeniería en Computación - Ingeniería en Información - Ingeniería de Sistemas - Ingeniería en Informática
Contenidos curriculares básicos	<ul style="list-style-type: none"> - Ciencias Básicas. - Teoría de la Computación. - Algoritmos y Lenguajes. - Arquitectura, Sistemas Operativos y Redes. - Ingeniería de Software, Bases de Datos y Sistemas de Información. - Aspectos Sociales y Profesionales. 	<ul style="list-style-type: none"> - Ciencia de la Computación - Humanidades - Formación de Empresas de Base Tecnológica - Ciencias Básicas
Contenidos afines con las propuestas DL	<ul style="list-style-type: none"> - Teoría de Base de Datos - Algoritmos y Programas - Programación Lógica - El Proceso de software. - Gestión de Configuración del Softwa. - Reingeniería de software. - Interacción Humano Computadora (HCI) - Sistemas WEB y móviles - Sistemas de Gestión de Bases de Datos. - Privacidad, Seguridad e Integridad en Bases de Datos. - Privacidad, integridad y seguridad en sistemas de información. - Sistemas basados en conocimiento - Cluster, Grid y Cloud Computing. - Sistemas cliente/servidor y sus variantes. - Propiedad Intelectual, licenciamiento de software y contratos informáticos. - Conceptos de Software libre, Hardware libre y Contenidos de acceso abierto. 	No se hallaron especificaciones

Es importante destacar que existen algunas disposiciones a nivel internacional que podrían ser tomadas en cuenta, aunque en los países en cuestión no hay estudios que detallen tales vínculos [25]. Para los profesionales de las Ciencias de Computación existe la *Association for Computing Machinery (ACM)*⁹ y la *IEEE-Computer Society*

⁹ <http://www.acm.org/>

(IEEE-CS)¹⁰ integradas por profesores de las universidades más importantes en el área y por profesionales de las empresas tecnológicas más representativas del mundo. La ACM es una asociación norteamericana que tiene como objetivo difundir el conocimiento y patrocinio de eventos del área de computación y afines. Por su parte, la IEEE-CS es una organización líder en la creación de estándares informáticos en el mundo, principalmente. Ambas organizaciones desarrollaron para los profesionales de las Ciencias de Computación una currícula que ha evolucionado en sus propuestas del 2001, 2008 y 2013 y abarcan una diversidad de áreas [28]. La recomendación del currículum de CC del 2013 [29] abarca dieciocho (18) áreas¹¹ de las cuales solo cuatro se relacionan con el Currículum DL: *HCI-Human-Computer Interaction*, *IAS-Information Assurance and Security*, *IM-Information Management* y *SE-Software Engineering*. A partir de tal revisión se destacan los siguientes contenidos curriculares de CC que trabajan conceptos del Currículum DL (incluso de forma indirecta): modelos de información y sistemas, sistemas de bases de datos, modelado de datos, procesamiento de transacciones, almacenamiento y recuperación de información, y el hipertexto e hipermedia, programación, diseño de interfases y sistemas, procesamiento de información humana, inteligencia artificial, ingeniería de software, preservación de los datos, entre otras.

Tomando en consideración las cuestiones mencionadas previamente (análisis de los planes de estudio de la UNdeC-UNET, las recomendaciones de la ACM y las propuestas relevadas del currículum DL) se pueden indicar los siguientes temas en un curso sobre bibliotecas digitales:

- Digitalización, almacenamiento e intercambio;
- Objetos digitales, composición y paquetes;
- Metadatos, catalogación, gestión de autores;
- Repositorios, archivos;
- Arquitecturas (agentes, buses, interoperabilidad, mediadores), servicios (búsqueda, linking, navegación);
- Gestión de derechos de propiedad intelectual, seguridad, privacidad, protección;
- Integridad de datos, preservación.

Esta propuesta brinda un paso preliminar que vincula el campo de las Ciencias de la Información y Ciencias Documentales (dominio LIS) con las Ciencias de la Computación mediante la incorporación de contenidos del Currículum DL. Se ha mencionado que el objetivo de esta propuesta es presentar un esquema inicial de una asignatura (electiva u optativa) ajustada al Currículum DL y los contenidos mínimos

¹⁰ <https://www.computer.org/>

¹¹ *AL-Algorithms and Complexity*, *AR-Architecture and Organization*, *CN-Computational Science*, *DS-Discrete Structures*, *GV-Graphics and Visualization*, *HCI-Human-Computer Interaction*, *IAS-Information Assurance and Security*, *IM-Information Management*, *IS-Intelligent Systems*, *NC-Networking and Communication*, *OS-Operating Systems*, *PBD-Platform-based Development*, *PD-Parallel and Distributed Computing*, *PL-Programming Languages*, *SDF-Software Development Fundamentals*, *SE-Software Engineering*, *SF-Systems Fundamentals* y *SP-Social Issues and Professional Practice*

recibidos por los estudiantes en sus carreras de las Ciencias de la Computación, que ayudará al perfil del profesional LIS en el dominio de las DL.

3.1 La propuesta

La propuesta se presentará en las carreras de Ingeniería en Sistemas en la Universidad Nacional de Chilecito de Argentina y de Ingeniería en Informática de la Universidad Nacional Experimental del Táchira de Venezuela. A continuación se detalla la propuesta:

- Disciplina: Ciencias de la Computación.
- Modalidad: curso electivo u optativo (según las disposiciones de cada Institución).
- Descripción: las propuestas para conformar un Currículum DL se desarrollan para especializar a los profesionales existentes y no para ampliar carreras de base que son parte del dominio LIS, como lo es las Ciencias de la Computación. Por ello, la propuesta presentada apunta a estudiantes avanzados de grado (cuarto y quinto año). De esta manera, dichos profesionales podrán ayudar a través de las competencias adquiridas a fortalecer con soluciones y propuestas en el dominio LIS (campo de trabajo, tesis de grados, investigación, etc). En la UNET se ofrecerá como una asignatura electiva de 64 horas y en la UNdeC como un seminario extracurricular de 64 horas, estructuradas en 4 horas por semana y en un lapso de 16 semanas. Se prevé implementarlo en el 2017, luego de la aprobación de los programas analíticos en cada institución.
- Objetivo: conocer el contexto de las bibliotecas digitales con el fin de fortalecer el perfil profesional en el dominio LIS.
- Temática: en primera instancia los alumnos deben entrar en contexto con los conceptos de las BD y RI, además de entender la razón del por qué se están implementando y su relación con el movimiento del Acceso Abierto. Luego, deben comprender el funcionamiento de las herramientas informáticas desarrolladas para la gestión de las BD/RI junto con la definición del concepto de objetos digitales (conocido también como recursos digitales). Después el alumno comienza a analizar y entender el proceso de recopilación, catalogación, almacenamiento, acceso, difusión y preservación de los objetos digitales, así como también, la interoperabilidad con los otros sistemas de gestión y funcionamiento interno de acuerdo con los diferentes estándares para el manejo de metadatos, tesauros, vocabularios controlados, esquemas ontologías, etc. Finalmente, el alumno debe interpretar el impacto bibliográfico de dichos sistemas y conocer los principales módulos de los mismos, a saber: búsquedas, navegación, recuperación, preservación e indexación. En síntesis, la estructura de contenidos mínimos de la propuesta son: concepto de las BD y RI (2 horas), Acceso Abierto (2 horas), sistemas de gestión de las BD/RI (4 horas), proceso de gestión de las BD conocido

por la norma ISO:14721 [30] (4 horas), modelo conceptual 5s [31] (4 horas), metadatos (4 horas), tesauros y vocabularios controlados (2 horas), ontologías (2 horas), indicadores bibliométricos (4 horas), digitalización (2 horas), búsquedas (4 horas), preservación (2 horas), almacenamiento-indexación y recuperación de información (6 horas), interoperabilidad (4 horas), gestión de autoridades (2 horas). Las horas restantes son parciales, consultas y prácticas.

- Metodología: se optará por una metodología participativa en la que los profesores y estudiantes puedan analizar casos, resolver situaciones y proponer diferentes soluciones. Se pretende que sea un espacio de intercambio y acceso a nuevos contenidos vinculados con la práctica real del dominio LIS.
- Evaluación: se considerará como un proceso continuo que posibilite el contacto con el dominio.
- Material bibliográfico adecuado a la orientación de la propuesta: artículo científico sobre el modelo conceptual 5s del 2004 [31] y los libros: *Theoretical Foundations for Digital Libraries* [32] del 2012, *Key Issues Regarding Digital Libraries* [33] del 2013, *Digital Library Technologies* [34] del 2014, y *Digital Libraries Applications* [35] del 2014.

4 Consideraciones finales

- El relevamiento realizado da cuenta de que no existen unidades curriculares del dominio LIS que se relacionen directamente con las Ciencias de la Computación, pero sí se evidenció que existe una relación indirecta que puede ser explotada con un esquema de asignaturas para estudiantes en el cuarto y quinto año de las carreras de CC, propuesta que ayuda al perfil profesional del dominio LIS como al de las Ciencias de la Computación, las cuales cada día están más relacionadas entre sí.
- La propuesta de asignatura en una carrera en Argentina y en Venezuela generará una retroalimentación que permita ver su impacto sobre el dominio LIS y CC. Luego, se podrá ampliar a otras asignaturas que profundicen sobre las bibliotecas digitales y/o ampliar los contenidos de las asignaturas existentes.
- Se destaca que si la información digital es global, los estudios para comprender la gestión de dicha información deberían de ser globales, se concluye que estas iniciativas antes descriptas han trabajado en poder agrupar ideas y maximizar el trabajo. Las propuestas de postgrados están logrando una unificación de criterios.
- Luego de la implementación y análisis de la propuesta presentada se realizará un estudio de las carreras de Ciencias de la Computación en Argentina y Venezuela, para tener una conclusión general en comparación

con lo desarrollado en este trabajo. Asimismo, se podrá realizar un estudio comparativo de estas carreras con la currícula recomendada por la ACM.

Referencias

1. Texier, J.: Los repositorios institucionales y las bibliotecas digitales: una somera revisión bibliográfica y su relación en la educación superior. Presented at the 11th Latin American and Caribbean Conference for Engineering and Technology - 2013 , Cancun, Mexico October 14 (2013).
2. Xia, J., Opperman, D.B.: Current Trends in Institutional Repositories for Institutions Offering Master's and Baccalaureate Degrees. *Serials Review*. 36, 10–18 (2010).
3. OpenDOAR: OpenDOAR - Home Page - Directory of Open Access Repositories, <http://www.opendoar.org/>.
4. ROAR: Registry of Open Access Repositories (ROAR), <http://roar.eprints.org/>.
5. Nguyen, S., Chowdhury, G.: Digital Library Research (1990-2010): A Knowledge Map of Core Topics and Subtopics. In: Xing, C., Crestani, F., and Rauber, A. (eds.) *Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation*. pp. 367–371. Springer Berlin / Heidelberg (2011).
6. Pomerantz, J., Wildemuth, B.M., Yang, S., Fox, E.A.: Curriculum Development for Digital Libraries. In: *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*. pp. 175–184. ACM, New York, NY, USA (2006).
7. Texier, J., De Giusti, M.R., Oviedo, N., Villarreal, G.L., Lira, A.J.: El uso de repositorios y su importancia para la educación en Ingeniería. Presented at the World Engineering Education Forum (WEEF 2012) “Educación en Ingeniería para el Desarrollo Sostenible y la inclusión social” October 18 (2012).
8. Liew, C.L.: Digital library research 1997-2007: Organisational and people issues. *Journal of Documentation*. 65, 245–266 (2009).
9. Chowdhury, G.G., Chowdhury, S.: Digital library research: major issues and trends. *Journal of Documentation*. 55, 409–448 (1999).
10. Morato, J., Sánchez-Cuadrado, S., Fernández-Bajón, M.-T.: Tendencias en el perfil tecnológico del profesional de la información. (2016).
11. Méndez, E., van Hooland, S.: Educación en Bibliotecas Digitales: análisis de un “nuevo” perfil profesional y su formación a nivel internacional. *Boletín de la ANABAD*. LIX, 147170 (2009).
12. Pomerantz, J.P., Oh, S., Yang, S., Fox, E.A., Wildemuth, B.M.: *The Core: Digital Library Education in Library and Information Science Programs*. (2006).
13. Lee, C.A., Tibbo, H.R., Schaefer, J.C.: Defining what digital curators do and what they need to know: the digcurr project. Presented at the Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries June 18 (2007).
14. Renear, A.H., Teffeau, L.C., Hswe, P., Dolan, M., Palmer, C.L., Cragin, M.H., Unsworth, J.: Extending an LIS Data Curation Curriculum to Include Humanities Data. (2009).

15. Mostafa, J., Pomerantz, J., Tibbo, H.R., McDonough, J., Palmer, C.: Developing curriculum for digital libraries and digital curation education: Reflections on synergies and divergencies. *Proc. Am. Soc. Info. Sci. Tech.* 45, 1–3 (2008).
16. Saracevic, T., Dalbello, M.: A survey of digital library education. *Proceedings of the ASIST Annual Meeting.* 38, 209–223 (2001).
17. Järvelin, K., Vakkari, P.: The evolution of library and information science 1965–1985: A content analysis of journal articles. *Information Processing & Management.* 29, 129–144 (1993).
18. Blummer, B.: Graduate and Post-MLS Study in Digital Libraries. *Journal of Access Services.* 3, 53–60 (2005).
19. Mitchell, E.: Trending Tech Services: Library and IT curriculum integration Part I. The case for a designed curriculum. *Technical Services Quarterly.* 31, 161–172 (2014).
20. Matusiak, K.K., Hu, X.: Educating a new cadre of experts specializing in digital collections and digital curation: Experiential learning in digital library curriculum. *Proc. Am. Soc. Info. Sci. Tech.* 49, 1–3 (2012).
21. Sue Myburgh, Anna Maria Tammaro: Education for Digital Librarians: Some European Observations. In: *Library and Information Science Trends and Research: Europe.* pp. 217–245. Emerald Group Publishing Limited (2012).
22. Kochtanek, T.R., Hein, K.K.: Delphi study of digital libraries. *Information Processing & Management.* 35, 245–254 (1999).
23. Franks, P.: *Infusing Digital Curation Competencies into the SLIS Curriculum. Lifelong Learning Programme.* (2014).
24. Ministerio de Educación de la Argentina: Resolución 786/2009. Boletín oficial Nro. 31.667, (2009).
25. Cuadros-Vargas, E., Silva-Sprock, A., Delgado-Castillo, D., Hernández-Bieliukas, Y., Collazos, C.: Evolution of the Computing Curricula for Computer Science in Latin America 2013. In: *Computing Conference (CLEI), 2013 XXXIX Latin American.* pp. 1–10 (2013).
26. García, S., Pereira, W., Paladino, R., Cardoso, L., de Gouveia, L.: Estudio sobre Competencias del Ingeniero en Informática: Caso Universidad Católica Andrés Bello. *Revista Venezolana de Computación.* 1, 01–07 (2014).
27. OPSU: Libro de Oportunidades de Estudios - MPPEUCT. 2016, http://loeu.opsu.gob.ve/vistas/carreras/areas_subareas_conocimiento.php?id=0.
28. ACM: Curricula Recommendations. 2016, <http://www.acm.org/education/curricula-recommendations>.
29. ACM Computing Curricula Task Force ed: *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science.* ACM, Inc (2013).
30. CCSDS: Reference Model for an Open Archival Information System (OAIS):ISO 14721. (2012).
31. Gonçalves, M.A., Fox, E.A., Watson, L.T., Kipp, N.A.: Streams, structures, spaces, scenarios, societies (5s): A formal model for digital libraries. *ACM Trans. Inf. Syst.* 22, 270–312 (2004).

32. Fox, E.A., Gonçalves, M.A., Shen, R.: Theoretical Foundations for Digital Libraries. The 5S Approach. Morgan & Claypool Publishers, North Carolina (2012).
33. Shen, R., Gonçalves, M.A., Fox, E.A.: Key Issues Regarding Digital Libraries: Evaluation and Integration. Synthesis Lectures on Information Concepts, Retrieval, and Services. 5, 1–110 (2013).
34. Fox, E.A., da Silva Torres, R.: Digital Library Technologies: Complex Objects, Annotation, Ontologies, Classification, Extraction, and Security. Synthesis Lectures on Information Concepts, Retrieval, and Services. 6, 1–205 (2014).
35. Fox, E.A., Leidig, J.P.: Digital Libraries Applications: CBIR, Education, Social Networks, eScience/Simulation, and GIS. Synthesis Lectures on Information Concepts, Retrieval, and Services. 6, 1–175 (2014).

HERRAMIENTA DIDÁCTICA PARA LA ENSEÑANZA DE CONTENIDOS Y PROCEDIMIENTOS EN UN CURSO UNIVERSITARIO DE SIMULACIÓN DE EVENTOS DISCRETOS

Darío Weitz

Facultad Regional Rosario – Universidad Tecnológica Nacional, Zeballos 1341, Rosario
dar.wtz@gmail.com

Abstract. Simulación es asignatura obligatoria en numerosas carreras de Ingeniería. Los contenidos del curso incluyen definiciones teóricas y metodológicas, elevado nivel de abstracción, cálculos rigurosos e intensivos. Se describe un software educativo que incluye elementos de animación. Se lo utiliza como herramienta didáctica para facilitar la comprensión y el aprendizaje de conceptos, procedimientos y métodos presentes en un curso de simulación de eventos discretos para la carrera de Ingeniería en Sistemas de Información. El software está conformado por módulos correspondientes a sistemas de espera, inventarios, atención de pacientes, comparación de los enfoques orientado a eventos, exploración de actividades, orientado a procesos. Fundamentado en teorías vigentes en enseñanza multimedia y en principios y heurísticas para la aplicación exitosa de gráficos animados, la herramienta didáctica demostró ser apropiada para propósitos educativos a nivel universitario de grado. El software educativo contribuyó a mejorar el nivel de aprobación del primer examen parcial de la asignatura.

Keywords: herramienta didáctica; simulación; eventos discretos; software educativo; animación.

1 Introducción

La mayoría de los sistemas del mundo real muestran una complejidad tal que inhiben su resolución mediante modelos analíticos. Por tal motivo, se utiliza la técnica de simulación como enfoque alternativo para modelizar al sistema bajo estudio y calcular ciertas medidas de rendimiento que permiten ganar comprensión sobre el funcionamiento del mismo. Se define a la simulación como el proceso de construir un modelo matemático o lógico de un sistema o problema de decisión, y experimentar con el modelo para obtener conocimiento del mismo o para asistir en la toma de decisiones [1].

Simulación es una disciplina consolidada de la Investigación Operativa y de las Ciencias de la Administración, motivo por el cual es asignatura obligatoria en numerosas carreras de Ingeniería o de Administración de Empresas. En la carrera de Ingeniería en Sistemas de Información, la asignatura conforma el nexo de vinculación entre la línea curricular de Sistemas, Matemáticas, Estadística e Investigación Operativa. Los contenidos del curso suelen poseer definiciones teóricas y metodológicas, elevado nivel de abstracción, procedimientos de cálculos rigurosos e intensivos, y requerir de los alumnos fundamentos sólidos en matemática y estadística.

En los cursos introductorios de simulación de eventos discretos se suele emplear SIMUL8 o Arena como entornos de simulación. Sin embargo, ambos muestran limitaciones cuando, en las primeras clases del curso, se deben explicar los conceptos básicos, los mecanismos de avance en el tiempo, las listas de eventos, los diferentes enfoques (orientado a los eventos, exploración de actividades, orientado a los procesos), los procedimientos de cálculo y cómo evolucionan los modelos de simulación de eventos discretos. En tal sentido, previo a la utilización de los entornos de simulación en ejercicios de aplicación de la técnica, es conveniente disponer de alguna herramienta que complemente la tarea del docente en el aula al momento de explicar los temas previamente detallados.

Las computadoras brindan la posibilidad de crear estrategias de aprendizaje innovadoras. Con frecuencia se utilizan software educativos, sistemas educativos multimedia y gráficos animados para enseñar sistemas complejos y conceptos abstractos. Se define al software educativo como aquellos programas de computación desarrollados con el objetivo de ser usados como facilitadores del proceso de enseñanza, y en forma consecuente del proceso de aprendizaje [2]. Los programas educativos comparten cinco características esenciales: i) tienen una finalidad didáctica; ii) utilizan la computadora como soporte; iii) son interactivos permitiendo el diálogo y el intercambio de información; iv) tienen la posibilidad de personificación de la velocidad de los aprendizajes; v) son relativamente fáciles de utilizar [3]. Entre las funciones que pueden realizar los software educativos se incluyen: informativa, instructiva, motivadora, evaluadora, investigadora, etc.

Las computadoras poseen también la capacidad de integrar elementos de animación como herramientas de enseñanza – aprendizaje. Se define a la animación como una visualización gráfica que cambia su estructura u otra propiedad a lo largo del tiempo, y que dispara la percepción de un cambio continuo [4]. La técnica de animación tiene un gran potencial para cumplir una función pedagógica en la enseñanza de conceptos complejos, particularmente aquellos que no son sencillos de describir verbalmente. También ayuda a crear imágenes mentales de sistemas que poseen elementos que cambian en tiempo y espacio. Sirve además para mostrar como la manipulación de una variable afecta a otras o la existencia de influencias simultáneas en sistemas cerrados [5].

En el presente trabajo se describe un software educativo que incluye elementos de animación. Se lo utiliza como herramienta didáctica para facilitar la comprensión y el aprendizaje de conceptos, métodos, procedimientos de cálculos y enfoques presentes en un curso universitario de simulación de eventos discretos para una carrera de ingeniería en sistemas de información.

2 Estrategia Didáctica

La carrera de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional (UTN) de la República Argentina tiene a la asignatura Simulación en el cuarto año del plan de estudios vigente. Proporciona la capacidad para simular situaciones que se presentan en empresas de manufactura, empresas de servicios o instituciones gubernamentales. El objetivo es que los alumnos puedan predecir, explicar, entrenar o identificar soluciones apropiadas en sistemas de espera, sistemas de inventario, sistemas de manufactura y ensamblado, organizaciones de salud, financieras, educativas y gubernamentales.

Se decidió desarrollar un software educativo para utilizarlo como apoyo educativo en el aula. Es un programa didáctico de carácter tutorial, cuya función en la estrategia didáctica es la proveer recursos y facilitar prácticas. Integra procesos cognitivos de análisis, síntesis,

clasificación y deducción. Está orientado a los alumnos con control del contenido del aprendizaje, pero a su vez está relacionado con las necesidades del docente en lo que concierne a la relevancia curricular [2].

Weitz [6] incorporó elementos de animación en un conjunto de módulos diseñados para la enseñanza de diversos contenidos educativos habitualmente incluidos en un curso introductorio de Teoría de Control Clásica. Fundamentado en teorías vigentes en enseñanza multimedia –teoría cognitiva de enseñanza multimedia [7]–, teoría de la fidelidad epistémica [8]–, en principios para la aplicación exitosa de gráficos animados –principio de congruencia, principio de aprehensión [9]–, y en heurísticas prácticas [10], se diseñó una herramienta didáctica muy eficiente para explicar las definiciones teóricas, los procedimientos de cálculos y los diagramas que se dictan en un curso de introducción a los sistemas de control realimentados. A partir del éxito de la citada estrategia didáctica, y dada las similitudes entre Teoría de Control Clásica y Simulación (sistemas que evolucionan en tiempo y espacio, procedimientos de cálculos intensivos y rigurosos, influencias simultáneas en sistemas cerrados, etc.), se decidió extender la idea conceptual y la experiencia ganada en el desarrollo del software educativo de Teoría de Control a contenidos de Simulación que presentan cierta complejidad para desarrollar en el aula.

3 Software Educativo

El software educativo fue desarrollado utilizando el lenguaje de programación C#. Es un lenguaje de programación de propósito general orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET. Además, es un lenguaje diseñado para la infraestructura de lenguaje común (Common Language Infrastructure, CLI), una especificación estandarizada que describe un entorno virtual para la ejecución de aplicaciones; tales aplicaciones, aunque estén escritas en diferentes lenguajes de alto nivel, pueden ejecutarse en diversas plataformas tanto de software como de hardware sin la necesidad de volver a compilar el código fuente.

Para la construcción de gráficos y curvas se utilizó la librería ZedGraph. Se trata de una librería de clases, control de usuario y control web escrita en C# que permite crear gráficas en proyectos .NET (Licencia: GNU Library o Lesser General Public License).

El software educativo está conformado por cuatro módulos correspondientes a procedimientos de cálculos que se utilizan en la simulación de eventos discretos para evaluar medidas de rendimiento de sistemas tales como los descritos en la estrategia didáctica. Corresponden a líneas de espera, sistemas de inventario, sistemas de atención de pacientes en organizaciones de salud y un módulo para la comparación entre el enfoque orientado a los eventos, el enfoque exploración de actividades y el enfoque orientado a los procesos. Los módulos se proyectan en el aula en formato ampliado mediante una computadora portátil y un proyector apropiado.

El módulo para la simulación de líneas de espera muestra la animación de una oficina en la cual los clientes arriban según una determinada distribución de probabilidad de los tiempos entre arribos y son atendidos por un servidor, también con una determinada distribución de probabilidad de los tiempos de servicio. La disciplina de la cola es FIFO, se asume una población infinita de clientes y un área de espera finita. Se puede seleccionar entre una distribución de probabilidad exponencial, uniforme o normal tanto para el proceso de arribo

como para el de servicio. Una vez elegida la distribución, se deben ingresar valores para los parámetros de la misma.

Haciendo click en el botón Simular comienza una animación de la llegada de clientes a la oficina (Figura 1). En la pantalla se observa el valor del reloj de la simulación, el status del servidor, el número de clientes en cola y el tiempo de partida del cliente que está siendo atendido. Cuando el número de clientes en cola supera un valor preestablecido se produce la negación del servicio. Pulsando el botón Pausa, se detiene la animación y se puede acceder a la visualización de dos gráficos: i) gráfico de la función número de clientes en cola $Q(t)$; ii) gráfico de la función ocupación del servidor $B(t)$. Estos gráficos se utilizan para la explicación de dos medidas de rendimiento habitualmente calculadas en los sistemas de espera: i) valor esperado del número promedio de clientes en cola $q(n)$; ii) valor esperado de la utilización del servidor $u(n)$.

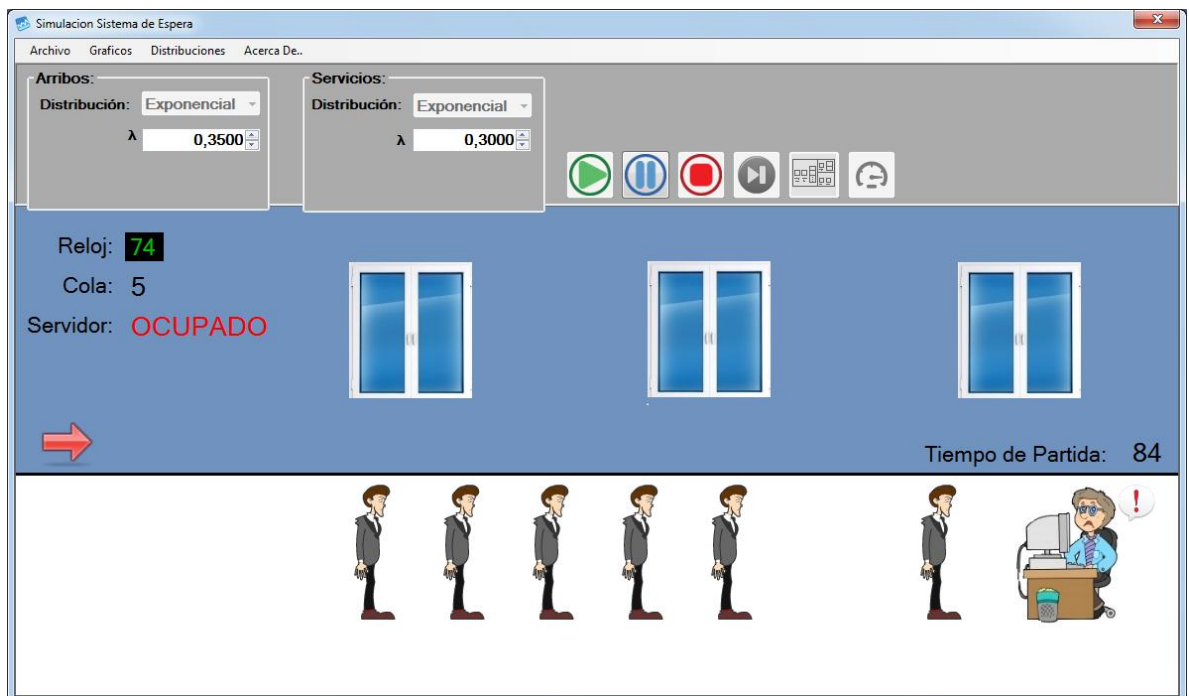


Figura 1: Animación de un sistema de espera

Law y Kelton [11] en su libro *Simulation Modeling and Analysis* proponen realizar un ejercicio de simulación evento a evento del sistema de espera para así mostrar las estructuras de datos involucradas, los cambios en las variables de estado, la acumulación en los contadores estadísticos y la interacción entre el reloj de la simulación y la lista de eventos. El módulo para la simulación de las líneas de espera permite que el docente tenga en simultáneo la representación del sistema físico, las gráficas de $Q(t)$ y $B(t)$ y las instantáneas de los valores que adoptan las variables de estado, el reloj de la simulación, la lista de eventos y los contadores estadísticos durante cada evento (Figura 2).

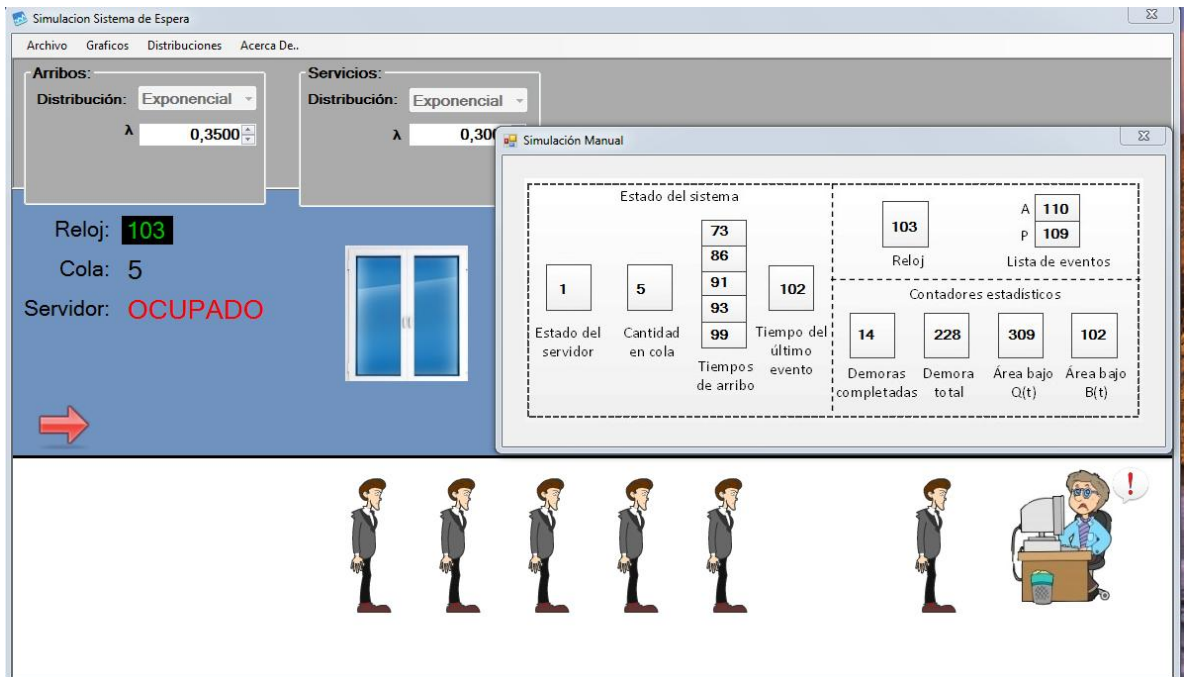


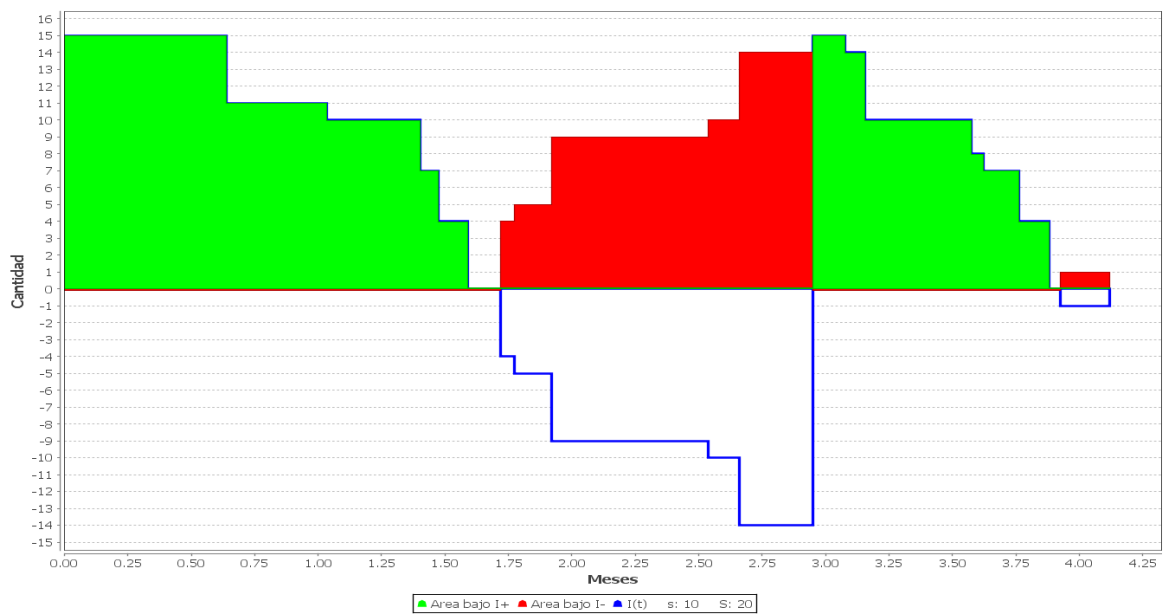
Figura 2: Instantánea de un evento del sistema de espera

El módulo sistema de inventario permite el cálculo del costo total promedio mensual como medida de rendimiento para evaluar políticas de reposición de stock. Se muestra la animación de un depósito (almacén) al cual llegan clientes según una determinada distribución de probabilidad de los tiempos entre arribos y demandan cierta cantidad de artículos según una distribución empírica. El módulo permite el ingreso de otros datos de entrada (inventario inicial, fin de la simulación, distribución y parámetros del retardo de envío) y valores para el cálculo de los costos de pedido, mantenimiento y faltantes (Figura 3). La política de reposición es estacionaria con parámetros nivel inferior de inventario (s) y nivel superior de inventario (S).

Haciendo click en el botón Animación comienza una representación de la llegada de clientes al depósito y la satisfacción de la demanda mientras el nivel de inventario pueda realizarla. Pulsando el botón Pausa se detiene la animación y se puede acceder a la visualización de gráficos que sirven para el cálculo del costo de mantenimiento, costo por faltante y costo de pedido. Se visualizan las gráficas de $I(t)$ –nivel de inventario al tiempo t –, $I+(t)$ –nivel real de inventario al tiempo t – e $I-(t)$ –nivel de faltantes al tiempo t – y sus correspondientes área bajo la curva (Figura 4). El módulo permite realizar una simulación evento a evento como explicación intuitiva de la técnica. De esta manera, el docente puede tener en simultáneo la representación del sistema físico, las gráficas de $I(t)$, $I+(t)$ e $I-(t)$ y las instantáneas de los valores que adoptan las variables de estado, la lista de eventos y los contadores estadísticos durante cada evento.



Figura 3: Pantalla de ingreso de datos al sistema de inventario



Ambos módulos incorporan un botón que permite realizar una simulación completa –sin animación– donde el parámetro a variar es la duración de la corrida. Al final, se muestran los valores finales de los estimadores de las medidas de rendimiento y gráficos de la evolución de las mismas a lo largo de la corrida. El objetivo es permitirle al docente explicar la conducta transiente y de estado estacionario del proceso estocástico que caracteriza tanto a las líneas de espera como a los sistemas de inventario previamente descriptos. También se incluye un botón

denominado Pronóstico que permite realizar 500 corridas independientes de los datos ingresados, con el objetivo de mostrar gráficos de frecuencia de las diferentes medidas de rendimiento para su posterior análisis durante el desarrollo del tema Análisis de Resultados.

El módulo para la simulación de atención de pacientes telemonitoreados intenta evaluar alternativas organizacionales en una institución de salud que se dispone a incorporar pacientes monitoreados de forma remota con sensores portátiles [12]. La Figura 5 muestra el modelo del sistema y las medidas de rendimiento que se calculan para ganar comprensión sobre el funcionamiento del mismo, y así ayudar a la toma de decisiones en un problema complejo y costoso.

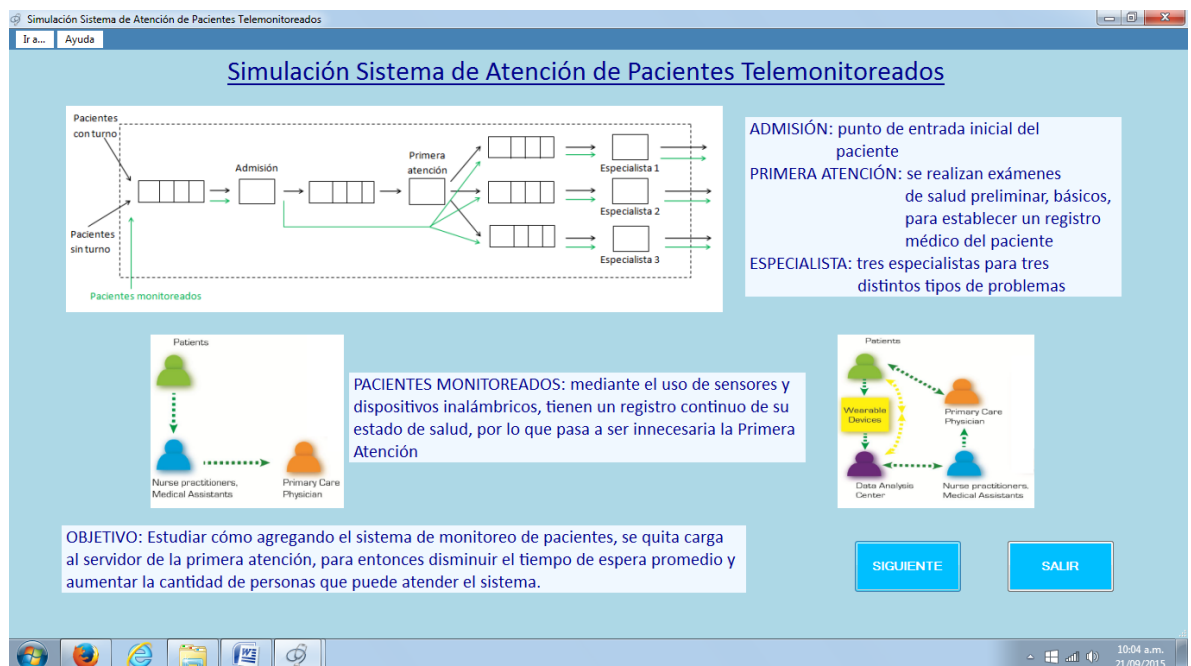


Figura 5: Modelo del sistema de atención de pacientes telemonitoreados

Haciendo clic en el botón Empezar, se procede al ingreso de los datos y comienza una animación de la llegada y atención de los cuatro tipos de pacientes (con turno, sin turno, monitoreados con turno, monitoreados sin turno) en el centro de salud. La animación puede visualizarse en dos (2D) o tres dimensiones (3D). La Figura 6 muestra la versión 3D y le sirve al docente para mostrar como la simulación puede utilizarse como medio de capacitación para la toma de decisiones y como se comporta el proceso durante las diferentes alternativas organizacionales. Otro recurso didáctico que provee el módulo es la realización del análisis de sensibilidad de las medidas de rendimiento.



Figura 6: Vista 3D del sistema de atención de pacientes telemonitoreados

Uno de los contenidos más complejos de explicar en un curso de Simulación está relacionado con tres enfoques utilizados habitualmente en el modelado de simulación: i) enfoque orientado a los eventos; ii) enfoque exploración de actividades; iii) enfoque orientado a los procesos. El primer enfoque, también denominado simulación conducida por eventos, describe los cambios que ocurren en el sistema debido a que ha ocurrido un tipo particular de evento. Los eventos se ordenan en orden cronológico y pueden no corresponder al flujo natural de las entidades del sistema [1]. Por su parte, la simulación conducida por procesos describe el flujo de las entidades individuales a través de todo el sistema y requiere para su codificación el uso de un lenguaje de simulación de propósito especial [11]. El enfoque evaluación de actividades enfatiza una revisión de todas las actividades presentes en la simulación para determinar cual puede iniciarse o finalizar al siguiente avance del reloj de la simulación. .

El enfoque orientado a eventos se explica mediante una animación que incluye dos elementos didácticos: i) el flujo de control que muestra las relaciones lógicas entre las diferentes componentes del modelo de simulación; ii) las instantáneas de los valores que adoptan las variables de estado, el reloj de la simulación, la lista de eventos y los contadores estadísticos durante cada evento (Figura 7). La animación resalta la actividad que se está ejecutando en la rutina que está siendo invocada y el cambio en el valor de la variable correspondiente al evento. El enfoque orientado a los procesos se explica mediante otra animación que también incluye dos elementos didácticos: i) el diagrama de flujo que describe la “experiencia” de un cliente “típico” en su fluir a lo largo del sistema; ii) una tabla que se va completando con las variables que se usan para el cálculo de las medidas de rendimiento. Ambas animaciones se refieren a sistemas de espera y el módulo contiene también procedimientos equivalentes para sistemas de inventario. El módulo incluye animaciones análogas para el enfoque exploración de actividades.

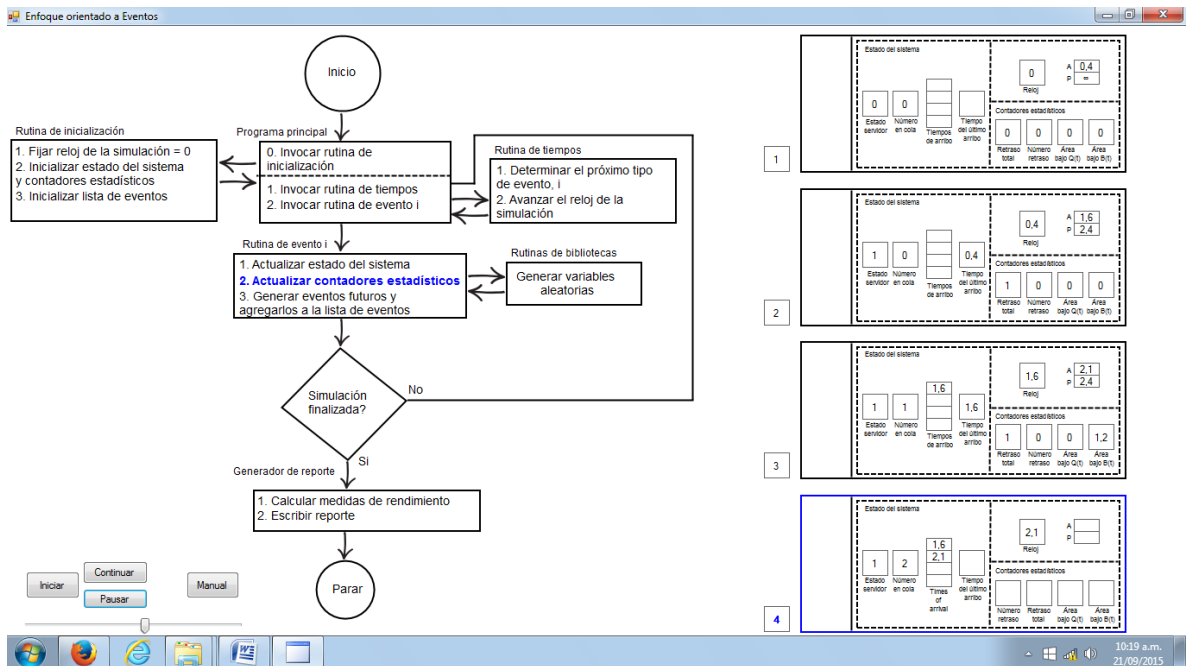


Figura 7: Animación del enfoque orientado a los eventos

4 Discusión

La utilización del software educativo para la enseñanza de conceptos, métodos, procedimientos de cálculos y enfoques presentes en la simulación de eventos discretos está justificada por los siguientes motivos: i) el movimiento y la trayectoria están presentes en los contenidos a enseñar; ii) los sistemas bajo estudio evolucionan con el tiempo; iii) el contenido y el formato de las imágenes se corresponden con el contenido y el formato de los conceptos y métodos presentados a los alumnos (principio de congruencia); iv) permite representar al modelo mental del docente con la mayor fidelidad posible (teoría de la fidelidad epistémica); v) el software educativo actúa en primer lugar como una estrategia de presentación al dar un contexto visual a una idea difícil de presentar verbalmente; vi) en una segunda etapa, actúa como función de clarificación al proveer al entendimiento de los métodos y procedimientos sin la necesidad de agregar información textual adicional; vii) los módulos fueron diseñados para que las imágenes fueran simples y no pasaran excesivamente rápidas por la pantalla (principio de aprehensión).

Los contenidos del curso de Simulación tuvieron importantes modificaciones con la entrada en vigencia del Plan de Estudios del año 2005. La primera etapa del software educativo (sistema de espera, sistema de inventario) se incorporó en el año 2013. La segunda etapa (sistema de atención de pacientes telemonitoreados, enfoques) se implementó para la cohorte 2015. Durante los años previos a la utilización del software educativo (2005-2012), la media de aprobación del primer examen parcial de la asignatura fue del 63,4%. Durante el período 2013 – 2014, la media de aprobación fue del 76,7%, mientras que el 81,4% de la cohorte 2015 aprobó tal evaluación parcial.

5 Conclusiones

Simulación es una asignatura obligatoria en la carrera de Ingeniería en Sistemas de Información en la Universidad Tecnológica Nacional de la República Argentina. Los planes de estudio de otras carreras de ingeniería y ciencias de la administración de universidades nacionales también la incluyen como materia obligatoria de grado. En la Facultad Regional Rosario de la UTN, un número significativo de alumnos mostraba dificultades para el aprendizaje de ciertos contenidos que se dictaban al comienzo del curso, lo que dificultaba la apropiada enseñanza de los contenidos siguientes.

El pizarrón y los libros de texto no son medios suficientes para visualizar apropiadamente la evolución de un sistema dinámico, como así tampoco para desarrollar y analizar la complejidad de los procedimientos de cálculos y enfoques que se utilizan en la simulación de eventos discretos. Por tal motivo, se diseñó un software educativo que incorpora elementos de animación para complementar la tarea del docente al momento de explicar sistemas de espera, sistemas de inventario, conducta transiente y de estado estacionario de los procesos estocásticos, gráficos de frecuencia, análisis de sensibilidad y enfoques en simulación de eventos discretos.

Fundamentado en teorías vigentes en enseñanza multimedia y en principios y heurísticas para la aplicación exitosa de gráficos animados, se desarrolló una herramienta didáctica que demostró ser apropiada para propósitos educativos a nivel universitario de grado. El software educativo descrito en este trabajo contribuyó a mejorar significativamente el nivel de aprobación del primer examen parcial de la asignatura.

Referencias

1. Evans, J.R., Olson, D.L.: Introduction to simulation and risk analysis. New Jersey: P. Hall. (1998).
2. Cataldi, Z.: Metodología de diseño, desarrollo y evaluación de software educativo. Tesis de Magister en Informática, Facultad de Informática, UNLP. (2000). <http://laboratorios.fi.uba.ar/lsi/cataldi-tesisdemagistereninformatica.pdf>
3. Marqués, P.: El software educativo. (1996). http://www.lmi.ub.es/te/any96/marques_software/
4. Schnotz, W., Lowe, R.K.: A unified view of learning from animated and static graphics. in: Lowe, R.K., Schnotz, W. (eds.) Learning with animation: Research implications for design. New York: Cambridge University Press, 304-356. (2008).
5. Rieber, L.P., Kini, A.: Theoretical foundations of instructional applications of computer-generated animated visuals. *Journal of Computer-Based Instruction*, 18(3), 83-88. (1991).
6. Weitz, D.A.: Effectiveness of Animation as a Learning Strategy in a Classical Control Theory Introductory Course. *World Journal Control Science and Engineering*, Volume 3, 1, 8-12. (2015).
7. Zoabi, W., Sabag, N., Gero, A. (2012). Using Animation to Improve the Student's Academic Achievement on Bipolar Junction Transistor. *American Journal of Engineering Education* – Fall 2012, 3, 2.
8. O'Donnell, F.: Simulation frameworks for the teaching and learning of distributed algorithms. Ph.D. Thesis, University of Dublin College, (2006). <http://www.tara.tcd.ie/bitstream/handle/2262/1277/TCD-CS-2006-20.pdf?sequence=1&isAllowed=y>
9. Tversky, B., Morrison, J.B., Betrancourt, M.: Animation: can it facilitate?. *Int. J. Human-Computer Studies*, 57, 247-262. (2002).
10. Weiss, R.E., Knowlton, D.S., Morrison, G.R.: Principles for using animation in computer-based instruction: theoretical heuristics for effective design. *Computers in Human Behavior*, 18, 465-477. (2002).
11. Law, A.M., Kelton, W.D.: Simulation Modeling and Analysis. New York: Mc Graw-Hill. (2000).
12. Weitz, D.A., Lianza, F., Nant, J.P., Schmidt, N., María, D.E.: Modelo de Simulación 3D para la Evaluación de Tecnologías de Monitoreo y Asistencia para Adultos Mayores. 3er Congreso Nacional de Ingeniería Informática / Sistemas de Información, CONAIISI 2015, Buenos Aires. (2015).

El paradigma de la enseñanza por competencias y los principios del conectivismo: una experiencia práctica

Marisa Cecilia Tumino y Juan Manuel Bournissen
Universidad Adventista del Plata, Libertador San Martín, Entre Ríos, Argentina
{marisatumino, juanbournissen}@doc.uap.edu.ar

Resumen. Uno de los grandes retos de las instituciones educativas consiste en formar profesionales competentes, capacitados para adaptarse a los incesantes cambios y desafíos que propone cada disciplina, con la consiguiente responsabilidad de implementar estrategias de enseñanza y evaluación que permitan identificar el grado de logro de las competencias profesionales.

El presente ensayo aborda la manera en que la Informática puede sentar las bases que acompañan estos cambios e introducir prácticas innovadoras tendientes a apropiarse de las virtudes del conectivismo en función del fortalecimiento de competencias que hacen al perfil profesional.

En el Conectivismo, el aprendizaje se produce mediante un proceso de conectar y generar información en el contexto de una comunidad de aprendizaje que actúa como un nodo, parte de una red más amplia con otros nodos que comparten entre sí recursos. A partir de la información recabada en el estudio, se estima que la metodología de aprendizaje auto-determinado, acompañado por los recursos que proporciona el conectivismo, crea un escenario beneficioso para la formación profesional que persigue el desarrollo de las competencias demandadas por la sociedad.

Palabras clave: Conectivismo, Aprendizaje auto-determinado, Tecnologías educativas, Proceso de enseñanza aprendizaje.

1. Introducción

El conductismo, el cognitivismo y el constructivismo son las tres grandes teorías de aprendizaje utilizadas más a menudo en la creación de ambientes instruccionales. Estas teorías, sin embargo, fueron desarrolladas en una época en la que el aprendizaje no había sido impactado por la tecnología. En los últimos años la tecnología ha reorganizado la forma en la que nos comunicamos y aprendemos.

La idea clave reside en el hecho de que “el aprendizaje es un proceso de formación de redes que tiene como agente principal al propio individuo”. La red conecta nodos que representan entidades externas tales como personas, organizaciones, bibliotecas, sitios web, wikis, libros, revistas y bases de datos, entre otros. “El acto de aprender consiste en crear una red externa donde los nodos se conectan para dar forma a una compleja fuente de conocimiento.”ⁱ

Según Laborda [1], introducir las TIC en la educación implica muchos cambios en el proceso de enseñanza-aprendizaje. Estas modificaciones se ven reflejadas en el tiempo, en el espacio y en los roles asumidos por los diferentes actores, lo que lleva a

rediseñar el contexto educativo. Entre los cambios que se producen se pueden mencionar los siguientes:

1.1. Cambios en el proceso educativo

Los constantes cambios en la sociedad del Siglo XXI llevan aparejado nuevas formas de relacionarnos, de comunicarnos y de interactuar. La información y el conocimiento tienen cada vez mayor influencia en los entornos profesionales que, sumado a una mayor diversificación de la población estudiantil, exige cambios en el sistema educativo, regido por el principio de igualdad de oportunidades y la no discriminación, que dé respuesta a la nueva sociedad de la información.

Para Laborda [1], se trata, no ya de enseñar sobre TIC, es decir de formar en las habilidades y destrezas que son necesarias para desenvolverse con soltura en la sociedad de la información; sino de dar un paso más y entender que utilizar las TIC en la educación significa seleccionar algunas de las herramientas que ofrecen las TIC y usarlas desde una perspectiva pedagógica, pero no como un complemento de la enseñanza tradicional sino como una vía innovadora que, integrando la tecnología en el currículo, consiga mejorar los procesos de enseñanza-aprendizaje y los progresos académicos de los estudiantes. Se trata, pues, de enseñar con y a través de las TIC, además de sobre TIC o de TIC.

Dado el reconocimiento de que la tecnología está propiciando la relación e interconexión de información, procesos y conocimientos de personas de diversas culturas de la manera más estrecha, inmediata y global, el progreso de la ciencia compromete enfoques metodológicos y referentes dialécticos diferentes que, entre otros aspectos, introducen la conectividad como un nuevo paradigma en los saberes y haceres. El aprendizaje no se vería favorecido si no se aceptase la posibilidad de mejorar las propias acciones mediante la corrección y la aceptación de la realidad compartida por los demás actores escolares. “La información ha cambiado el sentido del saber y hacer hacia el asentamiento de nuevos paradigmas planteados desde la tecnología” Padrón y Ortega [2].

1.2. Cambios en el objeto de la enseñanza

El objeto de la enseñanza es la transmisión de los conocimientos, pero con la llegada de las TIC a la educación, se cambia el foco de la misma para centrarlo en el alumno. Hay que enseñar a aprender para que este aprendizaje le sirva para toda la vida, lo que implica que hay que formar en competencias para que el alumno se adapte a la sociedad en constante cambio. En la actualidad se debe dejar el aprendizaje memorístico tan utilizado en la educación tradicional previa al Siglo XX.

El conectivismo promueve que la capacidad de saber más es más crítica que aquello que se sabe en un momento dado. Además sostiene que la habilidad de distinguir entre la información relevante y la irrelevante resulta vital. También es crítica la habilidad de reconocer cuándo una nueva información altera un entorno basado en las decisiones tomadas anteriormente. Por lo tanto nuestra habilidad para aprender lo que necesitamos mañana es más importante que lo que sabemos hoy, Siemens [3].

1.3. Cambios en los objetos educativos

Hoy se debe crear conocimiento de valor que perdure en el tiempo, y el alumno debe aprender a manejar la información eficaz y responsablemente, para de esa forma poder solucionar los problemas que se le presentan de la mejor manera.

MarquèsGraells[4] señala que los docentes del siglo XXI deberían estar capacitados para poder utilizar en forma transparente, es decir sin tener que recurrir a ayuda alguna, los siguientes tópicos:

1. Saber utilizar las principales herramientas de Internet.
2. Conocer las características básicas de los equipos.
3. Diagnosticar qué información se necesita, y saberla encontrarla
4. Evaluar la calidad y la idoneidad de la información obtenida.
5. Saber aprovechar las posibilidades de comunicación en Internet.
6. Conocer las funciones básicas de un paquete de oficina.

1.4. Cambios en las formas pedagógicas

Los cambios que se producen por la inclusión de las TIC en educación provocan cambios en el rol de los actores intervinientes, es decir, tanto en los docentes como en los estudiantes. El rol del docente, por una parte, se torna más profesional, más creativo y exigente, debe ser un asesor, un orientador y un facilitador, conoce la capacidad de sus estudiantes, evalúa los recursos o crea los suyos y crea un entorno favorable al aprendizaje basado en el diálogo y la confianza. Por su parte, el estudiante aumenta el interés por la materia estudiada, mejora la capacidad de resolver problemas, aprende a trabajar en equipos y comunicar sus ideas, adquiere mayor confianza en sí mismo e incrementa su creatividad e imaginación.

El uso de Internet en educación proporciona la oportunidad de usar metodologías más activas y motivadoras, que acercan la realidad profesional a la etapa formativa y permiten la adquisición de otras capacidades aparte de las directamente vinculadas al conocimiento científico técnico de la profesión. Barragán Sánchez, Mimbbrero Mallado y Pacheco González-Piñal [5].

1.5. El Conectivismo de George Siemens

Para Siemens [3], el conectivismo es la integración de principios explorados por las teorías de caos, redes, complejidad y auto-organización. El aprendizaje es un proceso que ocurre al interior de ambientes difusos de elementos centrales cambiantes – que no están por completo bajo control del individuo. El aprendizaje (definido como conocimiento aplicable) puede residir fuera de nosotros (al interior de una organización o una base de datos), está enfocado en conectar conjuntos de información especializada y las conexiones que nos permiten aprender más tienen mayor importancia que nuestro estado actual de conocimiento.

El conectivismo es orientado por las decisiones basadas en principios que cambian rápidamente. Continuamente se está adquiriendo nueva información. La habilidad de distinguir entre la información importante y no importante resulta vital. También es

crítica la habilidad de reconocer cuándo una nueva información altera un entorno basado en las decisiones tomadas anteriormente.

Principios del conectivismo:

1. El aprendizaje y el conocimiento dependen de la diversidad de opiniones.
2. El aprendizaje es un proceso de conectar nodos o fuentes de información especializados.
3. El aprendizaje puede residir en dispositivos no humanos.
4. La capacidad de saber más es más crítica que aquello que se sabe en un momento dado.
5. La alimentación y mantenimiento de las conexiones es necesaria para facilitar el aprendizaje continuo.
6. La habilidad de ver conexiones entre áreas, ideas y conceptos es una habilidad clave.
7. La actualización (conocimiento preciso y actual) es la intención de todas las actividades conectivistas de aprendizaje.
8. La toma de decisiones es, en sí misma, un proceso de aprendizaje. El acto de escoger qué aprender y el significado de la información que se recibe son vistos a través del lente de una realidad cambiante. Una decisión correcta hoy, puede estar equivocada mañana debido a alteraciones en el entorno informativo que afecta la decisión Siemens [3].

El conocimiento puede ser descrito de muchas maneras. Pero la cuestión que más preocupa aquí es “dónde reside”. Siemens [6] señala: el conocimiento “permanece” en el individuo pero “reside” en el colectivo.

2. Objetivos del estudio

Los objetivos del presente proyecto se describen a continuación:

Analizar las propuestas del conectivismo que dan respuesta a los desafíos de la formación por competencias en el desarrollo de sistemas informáticos.

Identificar los resultados de la aplicación de los principios del conectivismo como metodología de enseñanza-aprendizaje.

3. Metodología

Para el estudio, de tipo cuali-cuantitativo, se propusieron las siguientes etapas metodológicas:

1. Selección del conjunto de competencias en las que se enfoca la intervención.
2. Selección de la asignatura que atienda al desarrollo de las competencias seleccionadas en el paso anterior y de los temas relevantes involucrados en ese proceso.
3. Selección de estrategias de enseñanza y evaluación apropiadas al desarrollo de las competencias seleccionadas y las herramientas TIC adecuadas para dichas estrategias.

4. Implementación de las estrategias y aplicación de las herramientas TIC conforme al diseño propuesto.
5. Evaluación de los resultados. Los resultados fueron evaluados en términos del grado de desarrollo de las competencias profesionales involucradas. Las evidencias que sustentaron la evaluación se resumen en: (a) la argumentación presentada en cada situación de decisión del aprendizaje auto-determinado, (b) el producto materializado en los trabajos prácticos, (c) los procedimientos aplicados para resolver cada una de las consignas (la indagación se realizó mediante la puesta en común grupal y las indicaciones consignadas en el blog), (d) las evidencias que respaldaron el desarrollo de la responsabilidad y (e) las evidencias que respaldaron el desarrollo de la solidaridad manifestada en las respuestas obtenidas a preguntas precisas formuladas por la cátedra.

Por su parte, y a modo de complementar el estudio desde una perspectiva más integral, se procedió a adaptar y administrar un cuestionario elaborado por Cabero y Marín [7], con el objetivo de conocer las percepciones que los alumnos tenían acerca del trabajo en equipo colaborativo. Los reactivos de los 31 ítems del instrumento contemplaron una escala comprendida entre “Totalmente en desacuerdo” hasta “Totalmente de acuerdo”, pasando por “En desacuerdo”, “Neutral” y “De acuerdo”. Cabe señalar que el estudio se complementó con las observaciones de los investigadores, cuya interpretación atendió al enfoque cualitativo del paradigma mixto aplicado.

De esta forma se arribó a una aproximación del rol del conectivismo en el aprendizaje auto-determinado, como estrategias de enseñanza y evaluación que refuerza la formación de competencias profesionales, considerando el desarrollo holístico del individuo.

3.1. Descripción de los procedimientos

La asignatura seleccionada fue Programación V, dado su afinidad con el desarrollo de las competencias adoptadas para el estudio y descritas a continuación:

1. Desarrolla software en el marco de los paradigmas actuales de programación en función de las demandas del mercado y el contexto organizativo.
2. Diseña algoritmos para la solución de problemas, aplicando los conceptos de los paradigmas de programación.
3. Sirve a sus semejantes y se prepara para el liderazgo al identificar las necesidades del curso y trabaja junto a los compañeros para encontrar soluciones, con una actitud altruista y solidaria, en el marco de la responsabilidad social y el compromiso ciudadano.
4. Formula y gestiona proyectos con una actitud emprendedora constituyéndose en un agente de transformación de su entorno.

El tamaño de la muestra se vio limitada por la cantidad de estudiantes que cursaron la asignatura durante el 2015, conformando, en su totalidad, un curso de 11 estudiantes. Dado el tamaño de la muestra, solo se pretende obtener un acercamiento a las respuestas buscadas, sin procurar su generalización.

En el desarrollo de la asignatura se aplicó el enfoque de aprendizaje heurístico, el cual se centra en los estudiantes y sus experiencias previas de conocimiento obtenidos

en asignaturas anteriores. Se basa en encontrar, descubrir y crear contenidos a fin de guiar un ejercicio investigativo. Esta forma de aprender pone énfasis en el aprender a aprender, utilizando procesos no lineales y la autodeterminación del aprendiz. A medida que el estudiante aprende, diseña y desarrolla el mapa del aprendizaje, incluye todos los nodos que aportan al acrecentamiento de sus conocimientos. A tal efecto utiliza distintas herramientas, mecanismos y actividades que le permiten obtener la información necesaria para el logro de los objetivos, tales como lo mencionan Adell y Castañeda [8] al referirse a los PLE. Todo el proceso es acompañado por la reflexión que refuerza el aprendizaje autónomo y conduce a la obtención del producto que debe presentar como evidencia de la adquisición de las competencias en cuestión (aplicación software y portafolios), de la manera más eficientemente posible. Finalmente se compartieron los conocimientos obtenidos a través de: (a) una exposición en clase, en la que se explicó detalladamente tanto el proceso como el resultado y (b) un blog con el objetivo de llegar a estudiantes de otros ambientes académicos. En esta metodología los docentes se convierten en facilitadores del proceso de aprendizaje, creando ambientes propicios para la reflexión a partir de la formulación de preguntas esenciales y abiertas que promueven nuevos interrogantes. Los criterios se resumen en (a) Argumentación, (b) Trabajos prácticos, (c) Procedimientos, (d) Responsabilidad y (e) Solidaridad.

4. Resultados descriptivos

Luego de obtenidos y tabulados los datos recabados, se inició la etapa de procesamiento de los mismos. Las mayores medias se observaron en Trabajos prácticos y Responsabilidad (8,8 y 8,5 respectivamente), mientras que en Solidaridad se advierte la media de menor valor (6,2).

4.1. Resultados inferenciales: análisis de correlación

Una vez obtenidas las valoraciones por los medios descritos, se procedió a analizar las posibles correlaciones existentes entre los criterios considerados. Una relación lineal positiva entre dos variables, indica que los valores de las dos variables varían en forma similar. A los efectos de obtener las evidencias, se recurrió a los coeficientes de correlación, que permiten cuantificar el grado de relación lineal existente entre dos variables cuantitativas y valorar el grado de ajuste de la nube de puntos a una recta. En las pruebas ejecutadas se aplicó el Coeficiente de Pearson, que toma valores entre -1 y 1, puesto que es el coeficiente más utilizado para estudiar el grado de relación lineal existente entre dos variables.

El análisis de correlación lineal, que tuvo el propósito de identificar posibles relaciones entre las variables, mostró (a) el valor del coeficiente de correlación de Pearson y (b) el nivel crítico bilateral correspondiente.

El nivel crítico permite decidir sobre la hipótesis nula de independencia lineal. Se rechaza la hipótesis nula de independencia cuando el nivel crítico es menor que el nivel de significación establecido de 0,05, concluyendo que existe relación lineal estadísticamente significativa.

De esta forma, basando el análisis en los coeficientes de correlación y en los niveles críticos asociados, se corroboró la existencia de correlación significativa entre las variables consideradas en el estudio, dado que los coeficientes oscilaron entre 0.630 y 0.980, mientras que los niveles críticos, asociados a los estadísticos, fueron todos menores que 0,05.

Los datos recabados, acompañados de las pruebas estadísticas con sus correspondientes resultados, dan fe del desarrollo simultáneo de los atributos estudiados. Es decir que, para la muestra estudiada, se tiene evidencia del incremento de cada una de las variables consideradas, en la medida en que se observa el aumento de las restantes.

4.2. Análisis de regresión

A partir de la información obtenida del análisis de correlación, se procedió a aplicar el análisis de regresión lineal múltiple, como técnica estadística que permite estudiar la relación entre una variable dependiente y más de una variable independiente, con el doble propósito de (a) averiguar en qué medida la variable dependiente puede estar explicada por las variables independientes y (b) obtener predicciones de la variable dependiente a partir de las variables independientes.

A los efectos de analizar todas las combinaciones posibles, se procedió a trabajar cada una de las variables como variable dependiente y el resto como variables independientes, es decir que se reiteró la prueba para cada una de las variables incluidas en el estudio.

De las combinaciones posibles, se describen los resultados de aquellas pruebas cuyos resultados observaron el coeficiente de determinación cercano a uno y el estadístico F asociado a un nivel crítico menor que 0,05, lo que permite identificar la existencia de una relación lineal significativa.

En primer lugar, se tomó la Responsabilidad como variable dependiente mientras que la Solidaridad, los Procedimientos, la Argumentación y los Trabajos prácticos, participaron como variables independientes. Se encontró que estas últimas explican un 90% de la varianza de la Responsabilidad, pues, la R^2 corregida = 0,90. El nivel crítico asociado al estadístico $F(4)=23,48$ es de 0,001, indicando la existencia de una relación lineal significativa y, por lo tanto, un buen ajuste del hiper-plano, definido por la ecuación de regresión, a la nube de puntos definidos por los datos.

Los coeficientes Beta estandarizados indican la cantidad de cambio, en puntuaciones típicas, que se produce en la variable dependiente por cada cambio de unidad en la correspondiente variable independiente y manteniendo constantes el resto de las variables independientes. Estos coeficientes proporcionan información sobre la importancia relativa de cada variable independiente en la ecuación de regresión. En este caso, los coeficientes Beta indican que la variable Procedimientos es la más importante (1,204), seguida por Trabajos prácticos (1,202) y por Argumentación (0,991). Por su parte los niveles críticos menores que 0,05 de las pruebas t, indican que se debe rechazar la hipótesis nula de que un coeficiente de regresión vale cero en la población. En los niveles críticos asociados a cada prueba t, se advierte que solo la variable Argumentación posee un coeficiente significativamente distinto de cero ($p=0,003$). Los valores obtenidos dan evidencia de que solo la variable Argumentación es

relevante en la ecuación de regresión y, por lo tanto, contribuye de forma significativa a explicar lo que ocurre con la variable Responsabilidad.

En segunda instancia, se tomó Procedimientos como variable dependiente mientras que la Solidaridad, la Responsabilidad, la Argumentación y los Trabajos prácticos, participaron como variables independientes, encontrándose que estas últimas explican un 97% de la varianza de los Procedimientos, puesto que R^2 corregida = 0,967. El nivel crítico asociado al estadístico $F(4)=74,06$ es de 0,000, indicando un buen ajuste del hiper-plano, definido por la ecuación de regresión, a la nube de puntos.

Los coeficientes Beta estandarizados indican que la variable Trabajos prácticos es la más importante (0,962), seguida por Responsabilidad (0,398) y por Argumentación (0,392). Por su parte en las observaciones de los niveles críticos asociados a cada prueba t, se advierte que solo la variable Trabajos prácticos posee un coeficiente significativamente distinto de cero ($p=0,000$), por lo que se tiene evidencia de que solo esta variable es relevante en la ecuación de regresión y, por lo tanto, contribuye de forma significativa a explicar lo que ocurre con la variable Procedimientos.

En la tercera prueba se tomó Argumentación como variable dependiente mientras que Solidaridad, Responsabilidad, Procedimientos y Trabajos prácticos, participaron como variables independientes, encontrándose que estas últimas explican un 92% de la varianza de la Argumentación, con una R^2 corregida = 0,918. El nivel crítico asociado al estadístico $F(4)=29,078$ es de 0,000, indicando un buen ajuste del hiper-plano definido a la nube de puntos. Los coeficientes Beta estandarizados muestran que la variable Trabajos prácticos es la más importante (1,131), seguida por Procedimientos (0,967) y Responsabilidad (0,810). Los niveles críticos asociados a cada prueba t, revelan que solo las variables Responsabilidad ($p=0,003$) y Trabajos prácticos ($p=0,049$) poseen coeficientes significativamente distintos de cero, por lo que se tiene evidencia de que estas únicas variables contribuyen de forma significativa a explicar lo que ocurre con la variable Argumentación.

En último lugar, se seleccionó Trabajos prácticos como variable dependiente mientras que Solidaridad, Responsabilidad, Procedimientos y Argumentación, participaron como variables independientes, encontrándose que estas últimas explican un 97% de la varianza de los Trabajos prácticos, con una R^2 corregida = 0,968. El nivel crítico asociado al estadístico $F(4)=76,246$ es de 0,000, indicando un buen ajuste a la nube de puntos. Los coeficientes Beta estandarizados muestran que la variable Procedimientos es la más importante (0,935), seguida por Argumentación (0,445) y Responsabilidad (0,386). Los niveles críticos asociados a cada prueba t, revelan que solo las variables Procedimientos ($p=0,000$) y Argumentación ($p=0,049$) contribuyen de forma significativa a explicar lo que ocurre con la variable Trabajos prácticos.

A modo de descubrir el rol de la solidaridad en el proceso de aprendizaje auto-determinado, se analizó el comportamiento de los ítems adaptados del cuestionario de Cabero y Marín (2014), considerando para ello algunos de los ítems representativos de este atributo y la relación existente entre las potenciales variables en juego. De las combinaciones posibles, se encontró que los ítems “Me gusta que los cursos se desarrollen a través del trabajo en equipo” ($p=0,02$) y “Me gusta ayudar a otras personas al trabajar en equipo” ($p=0,034$), contribuyen de forma significativa a explicar lo que ocurre con la variable Responsabilidad, cuya varianza es explicada en un 60% por estos ítems.

Los resultados obtenidos muestran la interrelación entre gran parte de las variables estudiadas. Sin embargo, dada la limitación del tamaño de la muestra, los resultados deben ser interpretados con cautela, sin desatender las tendencias manifestadas por los datos recogidos.

Desde la interpretación de los resultados obtenidos, con la debida circunscripción a la muestra dado su tamaño limitado, se procedió a analizar la acción conjunta, sobre la variable Responsabilidad, de las variables (a) Argumentación, (b) Me gusta que los cursos se desarrollen a través del trabajo en equipo y (c) Me gusta ayudar a otras personas al trabajar en equipo; estas últimas como variables representantes de las preferencias por desarrollar los trabajos en equipo. Se encontró que las variables consideradas explican un 93% de la varianza de Responsabilidad, con una R^2 corregida = 0,929. El nivel crítico asociado al estadístico $F(3)=44,387$ es de 0,000, indicando un buen ajuste a la nube de puntos. Los coeficientes Beta estandarizados muestran que la variable Argumentación es la más importante (0,861), seguida por “Me gusta ayudar a otras personas al trabajar en equipo” (0,270). Los niveles críticos asociados a cada prueba t, revelan que solo las variables Argumentación ($p=0,000$) y Me gusta ayudar a otras personas al trabajar en equipo ($p=0,024$) contribuyen significativamente a explicar lo que ocurre con la Responsabilidad. Es decir que estas últimas variables podrían mejorar la predicción de los valores de Responsabilidad en un 93%.

5. Discusión y conclusión

Las múltiples fuentes utilizadas para recolectar los datos y los métodos aplicados para convertirlos en información, permiten señalar algunos aspectos interesantes que a su vez podrían ser profundizados en estudios con muestras representativas a fin de rescatar posibles generalizaciones y convertirlas en recomendaciones convenientes para ser consideradas en el currículo académico.

Se ha observado que la argumentación constituye una variable con posible poder predictivo sobre la responsabilidad de un estudiante. Este resultado conduce a suponer que cuando una argumentación tiene sustento sólido, da cuenta del cumplimiento previo de los requisitos de indagación de los contenidos abordados. Esto implicaría un proceso continuo de realimentación, donde el esfuerzo intencional proporciona los recursos que conducen al éxito de un emprendimiento.

Por su parte, cuando el estudiante adquiere habilidades para desarrollar con eficiencia los trabajos prácticos asignados, crea el andamiaje para construir procedimientos autónomos que robustecen la formación de las competencias esperadas; procesos reversibles que refuerzan el círculo virtuoso de responsabilidad y argumentación.

Si bien la solidaridad no apareció como variable predictiva de los atributos examinados, los análisis de correlación sugieren una estrecha vinculación entre ellos. Las observaciones de los investigadores permitieron advertir que aquellos estudiantes que mostraban predisposición al diálogo, al debate, al intercambio de ideas y al consenso, se esforzaban en mayor grado por participar en foros de discusión y, consecuentemente, obtenían información más relevante y actualizada que aquellos en quienes no se observó esta cualidad. La apertura hacia el aprendizaje colaborativo

facilitó el desarrollo de las competencias esperadas y propició el fortalecimiento de las relaciones interpersonales. La construcción de estas redes, con la correcta elección de nodos, redundó en resultados que confirman que el aprendizaje consiste en crear una red donde los nodos se conectan para dar forma a una compleja fuente de conocimiento. La medida de dicha complejidad permite contemplar de un modo más real, y acabado, la realidad en la que se espera actuar.

Se estima que la metodología de aprendizaje auto-determinado, acompañado por los recursos que proporciona el conectivismo, crea un escenario ventajoso como fundamento de la formación profesional que persigue el desarrollo de las competencias que la sociedad demanda de las instituciones educativas.

Las correlaciones estadísticamente significativas entre las variables en juego, sugieren el desarrollo simultáneo de los atributos representados. Estos hallazgos dan evidencia, para esta muestra, que las cualidades estudiadas se van nutriendo entre ellas y que no resultan de la improvisación sino del esmero y esfuerzo que cada individuo invierte no solo en su propia formación sino también en la formación de los demás compañeros que componen su red de aprendizaje.

Por lo expuesto, puede advertirse las bondades del conectivismo, y sus recursos, integradas a una metodología de aprendizaje auto-determinado donde se presentan las condiciones óptimas para desarrollar una red de interacciones en el marco de la formación holística del ser.

6. Referencias

1. Laborda, M. R. Las Nuevas Tecnologías en la Educación. Cuadernos/sociedad de la Educación 5. Fundación AUNA. Madrid, España. (2005).
2. Padrón, José y Ortega, Alfredo. La conectividad: Dogmatismo o nuevo referente paradigmático para el docente de vanguardia. *Revista de Investigación*, 75 n° 36, (2012): 129-142.
3. Siemens, G. Conectivismo: Una teoría de aprendizaje para la era digital. Recuperado de: [http://www.diegoleal.org/docs/2007/Siemens\(2004\)-Conectivismo.doc](http://www.diegoleal.org/docs/2007/Siemens(2004)-Conectivismo.doc). (2004)
4. MarquèsGraells, P. Criterios de calidad para los sistemas de teleformación. Plantilla de evaluación. Recuperado de: <http://www.peremarques.net/stfcalid.htm>. (2006).
5. Barragán Sánchez, R., Mimbreno Mallado, C. y Pacheco González-Piñal, R. Cambios pedagógicos y sociales en el uso de las IC: u-learning y u-portafolio. *Revista Electrónica de Investigación y Docencia (REID)*, n° 10, (2013): 7-20.
6. Siemens, G. Conociendo el conocimiento. Recuperado de: <http://craig.com.ar/biblioteca/Conociendo%20el%20Conocimiento%20-%20George%20Siemens.pdf>. (2006).
7. Cabero, Julio y Marín, Verónica. Posibilidades educativas de las redes sociales y el trabajo en grupo: Percepciones de los alumnos universitarios, *Revista Científica de Educomunicación*, n° 42(2014): 165-172.
8. Adell, J. y Castañeda, L. “Los Entornos Personales de Aprendizaje (PLEs): una nueva manera de entender el aprendizaje”. En Roig Vila, R. y Fiorucci, M. (Eds.) *Claves para la investigación en innovación y calidad educativas. La integración de las Tecnologías de la Información y la Comunicación y la Interculturalidad en las aulas*. Alcoy: Marfil – Roma TRE Universidadeglistudi. (2010).

¹<http://canaltic.com/blog/?p=800>

Sistema de Tutorías en Computación: Logros, Dificultades y Desafíos en el DCIC

Sonia V. Rueda y Luciano H. Tamargo

Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur,
Avenida Alem 1253, (B8000BCP), Bahía Blanca, Argentina
Tel: (0291) 459-5135 / Fax: (0291) 459-5136
`{svr,lt}@cs.uns.edu.ar`

Resumen En los últimos 10 años el Departamento de Ciencias e Ingeniería de la Computación de la Universidad Nacional del Sur ha invertido esfuerzo en la creación, desarrollo y consolidación de un sistema de tutorías. En este artículo se muestra una breve descripción de su gestación y crecimiento. Además, se muestra el trabajo realizado, enumerando las líneas de acción ejecutadas en pos de ayudar en la adaptación, curso y egreso de los alumnos, y en pos de despertar el interés de los estudiantes de nivel medio en las Ciencias de la Computación. En este trabajo también se describen los logros obtenidos, las dificultades encontradas y los desafíos planteados a futuro por el sistema.

Keywords: Tutorías, Computación, Líneas de acción.

1. Introducción

El Sistema de Tutorías del Departamento de Ciencias e Ingeniería de la Computación (DCIC) de la Universidad Nacional del Sur (UNS), se gestó en el año 2007 con la intención de colaborar en la difusión de la oferta académica, favorecer la integración de los alumnos ingresantes y brindar orientación a los estudiantes hasta su graduación. Se aspira además a reforzar la motivación que los impulsó en su momento a elegir su carrera y generar el sentido de pertenencia al DCIC en particular y a la UNS en general.

Con objetivos precisos, acciones concretas y el compromiso institucional de las autoridades del Departamento, el Sistema ha ido superando dificultades y ha alcanzado logros significativos. En la primera etapa la prioridad estuvo en establecer contacto con aspirantes y alumnos, explorando las estrategias de comunicación que resultaran más efectivas. Creado el vínculo, las actividades se orientaron luego a buscar soluciones para aquellos problemas que afectan al rendimiento de los alumnos y a su integración al ámbito universitario.

La primera sección de este trabajo presenta el marco institucional que sirve de contexto al Sistema de Tutorías del DCIC, formula los objetivos institucionales y los específicos de la unidad académica. En la siguiente sección se mencionan las principales acciones y actividades realizadas hasta el momento, para describir

luego las dificultades superadas y los logros alcanzados a través de dichas acciones y actividades. A continuación, se delimitan los criterios que orientan la selección de los tutores. Por último, se plantean los desafíos que se enfrentan actualmente y en base a los cuales se enfoca el trabajo futuro.

2. Marco institucional

En el año 2011 la Universidad Nacional del Sur, con la orientación de la Asesoría Pedagógica y la participación de todos los Departamentos Académicos, formula su **Proyecto Institucional para el Sistema de Tutorías** que establece como objetivos:

- Consolidar a la universidad como una institución que atiende a las dificultades de sus alumnos y procura soluciones desde su propio contexto.
- Generar la figura del tutor como vínculo entre el alumno y la institución universitaria.
- Fortalecer la permanencia de los alumnos de primer año a partir de distintas estrategias.
- Detectar los problemas que inciden en la inserción del alumno de primer año en la vida académica.
- Afianzar la motivación y el interés de los alumnos en el quehacer de la vida universitaria.
- Asesorar al alumno durante toda su carrera, cuando el mismo lo requiera.

El Proyecto forma parte del **Programa Apoyo y Contención para el Ingreso y Permanencia del Eje Inclusión e integración de los alumnos del Plan Estratégico de la Universidad Nacional del Sur** [1]. El modelo adoptado para implementar el Proyecto es la conformación de equipos de tutores, integrados por alumnos y docentes.

Es importante mencionar que la Universidad Nacional del Sur está organizada en Departamentos Académicos, no en Facultades. Cada Departamento administra una o varias carreras, pero además recibe y brinda servicios a otras unidades académicas. Así, los alumnos del DCIC cursan asignaturas a cargo de docentes de los Departamentos de Matemática, Física, Química, Economía e Ingeniería Eléctrica.

El Proyecto de Tutorías delega en los Departamentos Académicos la selección de los tutores, el mecanismo de renovación y la definición de objetivos específicos, en el marco de los objetivos generales. Algunos Departamentos habían iniciado sus acciones de tutorías antes de que se formulara el proyecto institucional.

Tal es el caso del Departamento de Ciencias e Ingeniería de la Computación que designó su primer equipo de tutores en 2007, en el marco del proyecto PACE-NI. En ese momento el equipo estaba integrado por cinco auxiliares de docencia. Con la coordinación de la Secretaría Académica del Departamento se establecieron los siguientes objetivos:

- Brindar orientación a los alumnos ingresantes para favorecer su adaptación al ámbito universitario.

- Identificar dificultades en la articulación entre las asignaturas que comparten cuatrimestre en primer año y que afectan al rendimiento académico de los alumnos, y transmitirlos a la Secretaría Académica.
- Colaborar en la difusión de la oferta académica participando en la *Muestra Anual de Carreras* y en el *Sistema de Adaptación formulado* con el objetivo de ofrecer pasantías para alumnos del último año del secundario.

Una premisa fundamental en la concepción de estos objetivos fue enfocar la acción tutorial tanto en las dificultades que los docentes percibían en los alumnos, como en los problemas e inquietudes que planteaban los mismos alumnos respecto al ámbito universitario, tanto académico como personal.

Al concluir el Programa PACENI el financiamiento del Sistema de Tutorías de cada Departamento que participó en dicho programa, se consolidó y se integró al presupuesto de la UNS. En 2013 el Programa de Becas Bicentenario permitió incorporar tutores pares al equipo y se potenciaron las actividades de difusión ofreciendo charlas en las escuelas secundarias y recibiendo visitas de escuelas en la UNS.

A partir de 2014, los planes de mejoras derivados de los procesos de acreditación de carreras de grado, permitieron sumar recursos y extender las actividades de tutorías al tramo medio y final de cada carrera. Con este financiamiento se incorporó además la figura de Coordinador de Tutores, un rol fundamental para fortalecer el Sistema.

En la actualidad, el equipo del DCIC está integrado por doce tutores que ofrecen orientación y acompañamiento a los alumnos de la *Licenciatura en Ciencias de la Computación, Ingeniería en Computación e Ingeniería en Sistemas de Información*. Los tutores brindan información, atienden consultas y escuchan las inquietudes y problemas que plantean los estudiantes durante toda la carrera. En algunas ocasiones colaboran en su resolución y en otras los derivan a las autoridades para su tratamiento.

3. Acciones y Actividades

El equipo de tutores del DCIC ha organizado sus actividades sobre cuatro ejes de acción:

- Difundir
- Acompañar
- Integrar
- Gestionar

La orientación, el asesoramiento, son funciones transversales a la difusión, el acompañamiento, la integración y la gestión.

Dentro de las actividades de difusión se incluyen las visitas a las escuelas, la atención del stand del Departamento en la Muestra Anual de Carreras y los laboratorios experimentales que se ofrecen a grupos de alumnos de escuela secundaria durante todo el año, pero en particular durante la Muestra. El objetivo

es brindar información acerca de qué es ser alumno de las carreras del DCIC, pero sobre todo qué hacen los graduados del DCIC, colaborando así en el descubrimiento de vocaciones [8].

El acompañamiento de los alumnos se inicia en la etapa de nivelación, el objetivo fundamental es establecer contacto de modo que los alumnos ingresantes reconozcan la figura del tutor. El evento de bienvenida y el Curso de Orientación Universitaria son oportunidades valiosas para presentar los canales de comunicación con el equipo.

La integración al ámbito universitario es un proceso durante el cual la expectativa es reforzar la motivación por estudiar la carrera elegida y desarrollar el sentido de pertenencia a la institución. El rendimiento académico está fuertemente ligado a la integración del alumno a su entorno y el aprendizaje de cuestiones:

- generales a toda la Universidad, como la distribución de las aulas, los períodos de inscripción en asignaturas o los requisitos para participar en los órganos de gobierno en representación del claustro de alumno;
- particulares de la carrera, como las restricciones de correlatividad o los requisitos de idioma;
- específicos de cada asignatura, como el mecanismo de evaluación o los requisitos de asistencia a clase.

Los alumnos enfrentan a su vez algunas dificultades:

- comunes a toda una cohorte, como puede ser una inadecuada distribución de los horarios de clase que provoca que tengan muchas horas de clases continuas o por el contrario, dos o más clases separadas por intervalos de 1 o 2 horas;
- particulares a un grupo, como por ejemplo los que desaprobaron una asignatura de nivelación o los que recursan una materia;
- específicas derivadas de problemas personales.

Las estrategias de comunicación para favorecer la integración atienden esta diversidad y ofrecen diferentes canales de interacción, tales como entrevistas personales, visitas a los cursos, reuniones de grupo o correo electrónico.

Las visitas a los cursos de primero y segundo año han relevado inquietudes y dificultades generales y particulares de un grupo. La clave es plantear algunos disparadores concretos que sirvan de punto de partida:

- ¿Conocen el mecanismo de evaluación de cada una de las asignaturas que están cursando? ¿Saben qué contenidos van a ser evaluados en cada parcial? ¿La cantidad de ayudantes en las clases prácticas es suficiente para consultar las dudas? ¿Toman notas durante las clases? ¿Consultan el material de estudio después de cada clase?
- El plan de estudios incluye un examen de suficiencia de inglés ¿saben que la universidad ofrece cursos gratuitos?

A partir de estas preguntas han surgido cuestiones vinculadas a la articulación entre los contenidos teóricos y prácticos de algunas asignaturas, la coordinación de las fechas de las evaluaciones, la superposición de horarios de consulta

con horarios de clases y otros inconvenientes vinculados a la gestión académica. También se han relevado observaciones vinculadas a aspectos pedagógicos y curriculares.

La principal dificultad de las visitas a los cursos es que el plan de estudios de las carreras del DCIC no incluyen una asignatura específica destinada a asesorar a los alumnos en temas tales como trámites administrativos, derechos, responsabilidades, etc. De modo que hay que solicitar autorización al profesor de una asignatura, en general se elige una materia del área Programación, para destinar unos minutos de la clase a la visita de los tutores.

Por lo general, se organizan tres visitas cada cuatrimestre, en las que se invierten entre 20 y 30 minutos. Aunque esta duración permite presentar brevemente algún tema concreto o recoger opiniones a partir de una pregunta específica, no es posible desarrollarlos en profundidad.

En las entrevistas personales y en las consultas a través del correo electrónico, surgen muchas preguntas vinculadas a trámites administrativos y requisitos del plan de estudio, en particular las restricciones de correlatividad. Estas cuestiones dan lugar a nuevas preguntas disparadoras, en las siguientes visitas a los cursos.

También se organizan reuniones con grupos específicos. Por ejemplo, los alumnos que avanzan de acuerdo al Plan de Estudios en las asignaturas del área de Programación y tienen dificultades en materias de Ciencias Básicas. Cuando se implementa un nuevo plan de estudios se organizan reuniones específicas para detectar dificultades que surjan en las materias nuevas o en las que han tenido un cambio de programa significativo.

En algunas reuniones, entrevistas o consultas se plantean problemas que demandan la participación de la Secretaría Académica del Departamento, la Secretaría General Académica o la Secretaría General de Bienestar Universitario. La tarea de los tutores es realizar el seguimiento posterior a la implementación de las soluciones de estos problemas.

El equipo de tutores organiza otro tipo de eventos. Por ejemplo, el Panel de Graduados el cual reúne a un grupo de entre 6 y 8 graduados de nuestras carreras. El fin de que los panelistas les cuenten a nuestros alumnos sus experiencias como estudiantes y como profesionales. Los tutores formulan una primera pregunta que responde por turno cada uno de los panelistas. A continuación, los alumnos plantean algunas consultas y luego los tutores proponen una segunda pregunta, que nuevamente vuelve a ser contestada por cada uno de los panelistas. El lema de esta actividad es: *“Ellos, hace poco estuvieron en tu lugar, hoy triunfan en el mundo de la computación, vení, escuchalos hablar de su experiencia y preguntales lo que quieras”*. Nuestro principal objetivo es reforzar la motivación de los alumnos de todos los tramos de nuestras carreras, al notar como los panelistas, que no hace mucho tiempo estaban en su lugar, llevan hoy una vida profesional exitosa.

El panel suele ser heterogéneo en cuanto al trabajo en el que se desempeñan los oradores, así como también a la forma de vivir que tiene cada uno. La expectativa es que los alumnos se sientan identificados con alguno de los profesionales,

y de esa forma puedan vislumbrar un posible camino en su futuro. Uno o dos de los tutores, actúan como moderadores favoreciendo la interacción.

La gestión involucran la planificación y evaluación de actividades, las reuniones del equipo de tutores, la elaboración de informes y los encuentros de socialización de experiencias con otros equipos. Todos los miembros del equipo participan en la planificación y el coordinador distribuye las tareas. Se incluyen en este eje la capacitación y la participación en Congresos y Jornadas de tutorías.

4. Dificultades superadas y logros alcanzados

Como se mencionó en la sección Marco Institucional uno de los objetivos del Sistema de Tutorías es Generar la figura del tutor como vínculo entre el alumno y la institución universitaria. En la actualidad podemos afirmar que la figura de tutor es reconocida en el ámbito del DCIC, tanto por los estudiantes como por los docentes.

En la primera etapa de la implementación del Proyecto los tutores manifestaban dificultad para establecer contacto, “los alumnos no nos consultan”, “no contestan las encuestas”, “muestran resistencia a ser entrevistados”. Sin embargo, en la actualidad esta dificultad se ha superado.

La participación de los tutores en las actividades de difusión de la oferta académica, durante los cursos de nivelación y en el evento de bienvenida, les permite establecer contacto con los aspirantes y alumnos ingresantes de manera temprana. En las visitas que realizan posteriormente a los cursos, los tutores no necesitan presentación, los alumnos los identifican y conocen sus actividades.

También es valioso mencionar en las primeras visitas, resultados concretos de las tareas de tutorías, porque estimulan a los alumnos a plantear sus inquietudes y recibir orientación. Así, no solo es importante que las actividades de los tutores estén respaldadas por acciones concretas de los órganos de gobierno departamental, sino que también es indispensable que estas acciones sean percibidas por los alumnos.

El panel de graduados es una actividad que resulta particularmente motivadora para los alumnos y contribuye a generar un sentido de pertenencia al Departamento. Los estudiantes, cualquiera sea el tramo de la carrera en el que se encuentran, comprueban que han elegido una profesión con un campo laboral ilimitado, que les va a brindar muchísimas oportunidades para elegir y desarrollarse.

Otro objetivo institucional alcanzado es que el trabajo de los tutores efectivamente brinda un canal de comunicación entre los alumnos y las autoridades. Cada año los tutores relevan inquietudes y problemas que manifiestan los alumnos y los transmiten a uno o varios órganos de gestión, como la Secretaría Académica del Departamento, los Directores de Carrera, el Consejo Departamental y la Secretaría General Académica. A su vez la gestión encomienda en el equipo de tutores reforzar la divulgación de información importante para los estudiantes.

Del relevamiento realizado por los tutores han surgido cambios en los horarios de las asignaturas, se han realizados ajustes en los cronogramas para mejorar la

coordinación de los contenidos de las clases teóricas y prácticas y se acordaron modificaciones en los mecanismos de evaluación para articular las asignaturas que corresponden a un mismo cuatrimestre en el plan de estudios. Además, se han debilitado algunos requisitos de correlatividad y se reforzó la presentación de algunos contenidos en las asignaturas iniciales a partir de observaciones realizadas por los alumnos del tramo medio, luego de un cambio en los planes de estudio.

Una dificultad surge cuando los tutores relevan una inquietud, la transmiten a la Secretaría Académica y en el cuatrimestre siguiente, o en el año siguiente, vuelve a detectarse la misma dificultad. Muchas de las situaciones se han ido resolviendo, pero otras vuelven a aparecer lo cual provoca desaliento.

Otra dificultad es que la cantidad de tutores es insuficiente para ofrecer un seguimiento personalizado. Esta dificultad se acentúa porque solo parte del equipo está financiado con fondos consolidados, los demás corresponden a programas especiales o planes de mejoras. La variación en la cantidad de integrantes del equipo, compromete el crecimiento del Sistema de Tutorías e incluso la continuidad de las actividades.

5. El perfil del tutor

El equipo de tutores del DCIC está conformado por docentes y estudiantes avanzados. Se aspira en primera instancia incluir representantes de las tres carreras del Departamento.

Las competencias comunicativas son fundamentales [5]. La capacidad para expresarse claramente es indispensable para brindar información a los alumnos y responder a sus consultas. También es muy importante para transmitir problemas, soluciones, ideas e inquietudes, tanto a otros miembros del equipo como a los docentes y las autoridades del Departamento.

Escuchar con atención es otra aptitud relevante, percibir las inquietudes de los estudiantes, aun cuando en primera instancia no se formulen con precisión y detectar eventuales conflictos. También es importante durante las reuniones grupales tener habilidades como mediador, cuando surgen posiciones contrarias, más aun cuando se manifiestan opiniones críticas.

Claramente las condiciones naturales van a incidir en las competencias comunicacionales, pero además es necesario ofrecer desde la Universidad mecanismos de capacitación que refuercen estas condiciones. Los talleres de socialización de experiencias permiten que los equipos de tutores de distintas unidades académicas compartan sus experiencias e identifiquen oportunidades para fortalecer acciones.

La capacidad de gestión es particularmente valiosa en el coordinador para planificar y evaluar las acciones que se realizan cada año, estimar recursos, establecer el cronograma y la duración de cada actividad, y elegir el momento adecuado para cada una. En ocasiones es necesario reprogramar, considerando la dinámica del contexto y las situaciones particulares que se suscitan [2].

No es responsabilidad de los tutores resolver los problemas que detectan, sino transmitirlos. Sin embargo, con frecuencia colaboran en la búsqueda e incluso implementación de soluciones.

Más allá de las capacidades mencionadas, una cualidad fundamental en el tutor es el compromiso con los objetivos del Sistema de Tutorías y la convicción de que su participación como miembro del equipo de tutores, puede incidir profundamente en algunos estudiantes.

6. Conclusiones y trabajo futuro

El Sistema de tutorías de la UNS ha adoptado un modelo basado en la conformación de equipos de trabajo integrados por docentes y alumnos. Un modelo alternativo consiste en comprometer al cuerpo docente en su conjunto. La acción tutorial ya no se concentra en este caso en un equipo específico de trabajo sino que involucra a todos los actores [9] y [7]. En nuestra opinión ambos modelos pueden complementarse.

La orientación académica es inherente a la función docente y la mayoría de los profesores y auxiliares de nuestro Departamento la ejercen cada día. Los docentes brindan un aporte valioso cuando surgen dudas respecto a la vocación, interactúan con otros docentes para analizar el rendimiento académico de sus alumnos, exploran nuevas estrategias pedagógicas buscando motivar a los estudiantes. Algunos estimulan a los alumnos a plantear inquietudes, más allá de las consultas propias acerca de los contenidos de la asignatura, pero en general la mirada está centrada en cuestiones académicas inherentes a la materia. Los docentes que forman parte del Consejo Departamental o las comisiones curriculares tienen la oportunidad de analizar y definir modificaciones en los planes de estudio o en los mecanismos de nivelación.

El docente que integra un equipo de tutores tiene un compromiso explícito con actividades específicas de asesoramiento, que no están ligadas a las asignaturas en las que cumple funciones. Algunos de los tutores docentes participan en la orientación de estudiantes de nivel secundario, otros colaboran en el acompañamiento de alumnos de primer año y otros están involucrados con acciones destinadas a contener a los alumnos del tramo medio o final de la carrera. En cada una de estas acciones surgen cuestiones vinculadas a aspectos académicos, pero también sociales y hasta afectivos.

El tutor par aporta una componente especial a la función orientadora, su perspectiva por lo general es más cercana a la de los alumnos a los que brinda orientación. Puede reconocer problemas que él mismo enfrentó recientemente, detectar dificultades e interpretar inquietudes.

El desafío es integrar la orientación académica del cuerpo docente, la capacidad de gestión de los órganos de gobierno y el trabajo del equipo de tutores para colaborar en la integración de los alumnos al ámbito universitario y favorecer el avance en cada tramo de la carrera, hasta alcanzar la graduación. En este sentido es importante aunar esfuerzos con los Programas de Articulación con Escuelas

Secundarias, el trabajo de las comisiones curriculares y los docentes involucrados en la Nivelación [4].

También ha surgido la propuesta de generar desde el Sistema de Tutorías espacios para desarrollar vínculos entre los propios alumnos. Esto es, aprovechar una de las visitas a los cursos para que cada tutor reúna a un grupo de entre 12 y 15 alumnos de modo que cada uno se presente y comparta alguna experiencia positiva y otra negativa ocurrida desde su ingreso. Esta actividad se ha implementado exitosamente en carreras que incluyen una asignatura dentro del plan de estudio, destinada a la integración de los estudiantes a la UNS.

Otro desafío es mejorar la gestión académica. Si esperamos que los alumnos planifiquen sus actividades y administren el tiempo adecuadamente, los docentes debemos atender a la coordinación. Además, debemos publicar el material anticipadamente, informar el mecanismo de evaluación, conocer los mecanismos de evaluación de las otras asignaturas y los horarios de clase, evitar ofrecer consultas en los horarios de clase de otras materias, articular las actividades de profesores y auxiliares, son algunas de las cuestiones que inciden directamente en el rendimiento de los alumnos.

La coordinación es particularmente importante en el último tramo de la carrera, ya que muchos alumnos trabajan y tienen dificultad para asistir a clase, sobre todo si no se definen franjas horarias. Cuando la carrera se prolonga suelen representar un obstáculo los trámites administrativos, tales como reincorporaciones, reválidas, solicitudes de equivalencias. El reconocimiento de estas dificultades y la implementación de mecanismos para superarlos, tiene incidencia en la graduación.

También pensando en el último tramo, la expectativa es colaborar desde el Sistema de Tutorías con la Secretaría de Extensión del DCIC, que ha iniciado el seguimiento de alumnos senior aplicando como modelo el Programa propuesto por el Departamento de Ingeniería Eléctrica y de Computadoras [6]. En ese sentido es muy importante contar con herramientas que permitan identificar situaciones específicas como puede ser por ejemplo estudiantes que terminaron de cursar hace 2 años o más y aun no se han recibido.

También dentro del ámbito de la gestión, una experiencia valiosa la aportó el Departamento de Ciencias de la Administración [3]. La propuesta consiste en sistematizar la planificación y evaluación de las actividades que se realizan durante el año. En cada mes de febrero se define un Diagrama de Gantt para organizar las tareas a desarrollar entre los meses de marzo y noviembre, partiendo de un conjunto de actividades tipificadas y estableciendo indicadores precisos. Los indicadores se utilizarán en la evaluación que se realiza en el mes de diciembre.

La sistematización de actividades permite asignarle un espacio específico a las reuniones del equipo de tutores, a la generación de informes y reuniones con autoridades, a los encuentros de socialización de experiencias y a las actividades de capacitación. Cuando los miembros del equipo se renuevan, la planificación sistemática facilita la transición.

Referencias

1. Plan estratégico de la universidad nacional del sur 2011 - 2016 - 2026. EdiUns, Bahía Blanca, Argentina, 2013.
2. Mastache A. Formar personas competentes. reflexiones y experiencias. In *Novedades Educativas. Buenos Aires*, 2007.
3. J. Beck, C.M. Genovese, M.J. Iglesias, J.M Lavia, C. Pignol Ramos, J.M. Salvador, and F. Schwerdt. Sistematización del trabajo del equipo de tutorías del departamento de ciencias de la administración de la universidad nacional del sur. In *III Congreso Argentino de Sistemas de Tutorías*. Universidad Nacional del Centro de la Provincia de Buenos Aires Tandil, 2015.
4. Clérici Tealdi J.V. and Amieva Camargo R.L.; Las perspectivas de docentes ingenieros sobre las tutorías en la universidad. In *Revista de Docencia Universitaria*, volume 11, pages 375–395, 2013.
5. A. Mastache, V. Sanchez Negrete, N. Goggi, and S. Barreiro. Una mirada integral al acompañamiento al estudiante. In *I Congreso Argentino de Sistemas de Tutorías*. Oberá, septiembre 2010, 2010.
6. A.P. Repetto, D.G. Sánchez, and G.L. Calandrini. El programa de seguimiento de alumnos seniors. In *III Congreso Argentino de Sistemas de Tutorías*. Universidad Nacional del Centro de la Provincia de Buenos Aires Tandil, 2015.
7. Amieva R.L. Metáforas del cambio educativo: los sistemas de tutorías en las facultades de ingeniería. In *Revista Argentina de Enseñanza de la Ingeniería*. Abril, 2014.
8. S. Rueda, A. Cohen, T. Delladio, Gottifredi S., and Tamargo L.H. Herramientas para apoyar el descubrimiento de vocaciones en ciencias de la computación. In *XX Congreso Argentino de Ciencias de la Computación (CACIC 2014)*, 2014.
9. D.G. Sánchez, Repetto A.P., and Doñate P.D. Programa de seguimiento de alumnos: La institucionalización del sistema de tutorías en el departamento de ingeniería eléctrica y de computadoras. In *I Congreso Argentino de Sistemas de Tutorías*. Oberá, septiembre 2010, 2010.

Experiencia con Realidad Aumentada. Análisis de la incidencia en la comprensión de las estructuras de control

Natali Salazar Mesia^{1,2}, Gladys Gorga², Cecilia Sanz².

¹ Becaria Tipo A – Universidad Nacional de La Plata

²Instituto de Investigación en Informática LIDI. Facultad de Informática – Universidad Nacional de La Plata
{nsalazar, ggorga, csanz}@lidi.info.unlp.edu.ar

Abstract. Se presenta una experiencia que utiliza el material educativo hipermedial EPRA que incorpora actividades de Realidad Aumentada para el uso y aplicación de estructuras de control en cursos iniciales de programación. Uno de los aspectos a analizar durante la experiencia ha sido el impacto en el aprendizaje de las estructuras de control. Se realizó una evaluación diagnóstica con 192 alumnos de Conceptos de Algoritmos, Datos y Programas (CADP) y 84 alumnos de Programación 1 de las carreras informáticas de la UNLP, que permitió visualizar las dificultades en la comprensión y aplicación de las estructuras de control. Luego la experiencia se llevó a cabo con 24 alumnos de CADP y 30 alumnos de Programación 1. Los resultados obtenidos en la experiencia permiten observar una mejora en la comprensión del tema, y a la vez, se percibe que la Realidad Aumentada ha logrado la motivación de los alumnos en el aprendizaje.

Keywords: Realidad Aumentada, Impacto en el aprendizaje, Material educativo digital

1 Motivación

La Realidad Aumentada (RA) se presenta como una tecnología con un importante potencial para el escenario educativo [1, 2]. En [3] se afirma que: "... [la contextualización que ofrece la RA] permite que los estudiantes adquieran experiencias y aprendan, además de la comprensión, el cómo los conceptos adquiridos en el aula se aplican para resolver problemas en situaciones del mundo real. En tales contextos, la RA potencia que los estudiantes obtengan una apreciación más profunda de aprendizaje, relacionando los contenidos de aprendizaje a sus propias experiencias".

Por otra parte, los estudiantes que ingresan a las universidades hoy en día se caracterizan por ser nativos digitales. En [4] se explica, retomando las investigaciones de Prensky [5]: "... [se reconoce a] los nativos digitales como la primera generación que ha crecido con las tecnologías digitales y que son 'nativos' del lenguaje de los ordenadores, videojuegos e Internet, mientras que los inmigrantes digitales son aquellos que no han crecido en un mundo digital, pero se han acercado a esta

tecnología adoptando algunos aspectos”. En [4] también se caracteriza a los nativos digitales como personas que realizan varias cosas al mismo tiempo (multitarea), precisan de la inmediatez en las respuestas, prefieren los gráficos al texto, crean videos, y utilizan presentaciones multimedia, entre otras características.

La experiencia que aquí se presenta parte de lo afirmado en estas investigaciones y propone innovar en la enseñanza de conceptos básicos de programación, en este caso en el tema de estructuras de control, para alumnos que mayoritariamente se caracterizan por ser nativos digitales. La propuesta incluye la utilización de actividades de Realidad Aumentada para vivenciar y experimentar las estructuras de control a partir de la resolución de problemas sencillos y con una inmediata visualización de lo que implica elegir en cada caso una estructura de control u otra, apelando a la modificación de la escena real con objetos virtuales acorde al problema planteado. Para ello se trabajó en sesiones experimentales durante 2015 y 2016, en las asignaturas Conceptos de Algoritmos, Datos y Programas (CADP) y Programación 1 de las carreras que se dictan en la Facultad de Informática de la UNLP.

En este artículo se presentan los resultados de estas sesiones experimentales. Se parte de un análisis diagnóstico de la comprensión del tema por parte de 192 alumnos que cursan la asignatura CADP y 84 alumnos de Programación 1. Luego, se selecciona un grupo de referencia, de cada asignatura, para que utilice el material EPRA, y desarrolle las actividades propuestas. Finalmente, se realiza un postest con este grupo participante de la experiencia. Los primeros resultados indican que se observa una mejora en la comprensión de las estructuras de control abordadas como parte del material desarrollado y a la vez se percibe que la Realidad Aumentada ha logrado la motivación de los alumnos para el tratamiento del tema seleccionado [6].

Este artículo se organiza de la siguiente manera, de aquí en adelante: en la sección 2 se presentan antecedentes del uso de RA en el aprendizaje de conceptos abstractos, en la sección 3 se describe el material educativo digital EPRA, en la sección 4 se realiza la descripción de la experiencia, en la sección 5 se analizan los resultados obtenidos a partir de los pretest y postest, y finalmente en la sección 6 se presentan conclusiones y trabajos futuros.

2 Realidad Aumentada en escenarios educativos

La RA es una tecnología que permite integrar el mundo real con el mundo digital. Tal como se describe en su nombre esta tecnología ‘aumenta’ el escenario real a partir de la incorporación de objetos virtuales [3]. Pueden utilizarse marcadores, códigos QR, el posicionamiento de la persona y/o partes de la escena para lograr registrar los objetos virtuales en la escena real.

El aporte de la RA está siendo cada vez más reconocido por los investigadores de la educación, donde se destaca que la coexistencia de objetos virtuales y entornos reales permite a los estudiantes visualizar conceptos abstractos y relaciones espaciales complejas. Esta situación se valora especialmente en el ámbito de la enseñanza de la Informática [7]. En [8] se afirma que hay consenso entre los educadores en relación a la necesidad de interacción con los contenidos por parte de los alumnos. Desde este punto de vista parece claro que, la tecnología de RA ayudará al proceso de

aprendizaje de los alumnos debido, entre otras razones, al alto grado de interacción que proporciona. Al mismo tiempo, se coincide con la referencia ya mencionada de [3], en la que se valora la posibilidad de que el alumno realice su propia experiencia en relación a los temas a aprender. A continuación, se presentan algunos ejemplos del uso de RA en diferentes ámbitos educativos enfocados principalmente al abordaje de conceptos abstractos.

En [9] se describe una experiencia en un ambiente de aprendizaje informal que consiste en la visita a una exhibición de Matemática en un museo y utiliza RA para incorporar información adicional a la muestra. Se trabaja con dos grupos: uno que participa de manera interactiva de la muestra utilizando RA y otro que realiza el recorrido tradicional. El objetivo que se propusieron estos autores es el de comparar los conocimientos matemáticos adquiridos en ambos grupos, para ello utilizaron pretest y postest que ayudaron a analizar los conocimientos de los participantes. Se detalla, a partir de los resultados presentados, que la incorporación de RA en un ambiente informal (museo) es efectivo en la adquisición de conocimiento sobre matemática y han demostrado que facilita la comprensión de contenido y la retención de largo plazo.

En [10] también se utilizó esta metodología de pretest y postest en una experiencia para aprender a volar un dron. Se tomó un pretest sobre el total de participantes acerca de las nociones básicas de un dron y luego se dividieron en dos grupos, uno que trabaja con contenido interactivo utilizando Realidad Aumentada, y otro grupo que realiza un abordaje más tradicional con texto, gráficos, audio y videos. Los resultados analizados en el postest indican que el 77% de los participantes del primer grupo tardaron menos tiempo en volar un dron real, mientras que en el segundo grupo sólo el 33% logró hacerlo dentro de ese lapso de tiempo. Estos resultados han sido alentadores para el trabajo que aquí se presenta y han servido de base para el diseño de la experiencia.

3 Material educativo digital EPRA

El material educativo EPRA es un sitio web¹ (con licencia *Creative Commons* de atribución no comercial) que incluye contenido teórico y actividades basadas en RA con el uso de marcadores y la detección del rostro de la persona que está interactuando para lograr aumentar la escena [11]. Incluye además una sección de presentación con el objetivo de este material educativo y otra de ayuda con tutoriales. Hay tres tipos de actividades de RA: Actividades de Exploración, Actividades de Repaso, y Actividad de Integración. A continuación se describen brevemente los diferentes tipos de actividades incluidas en EPRA. Una descripción más detallada de este material educativo y sus actividades puede ser encontrada en [6, 11, 12]:

- A. Actividades de exploración: a través de este tipo de actividades se busca que el alumno vivencie cada una de las estructuras de control (If, While, Repeat y For) a partir de situaciones reales y lúdicas que se le presentan (fig. 1a).

¹ Sitio web: <http://163.10.22.174>

- B. Actividades de repaso: a través de estas actividades, el alumno podrá observar y comparar el comportamiento de las estructuras de control. Así se presentan las opciones: If vs. While, While vs. Repeat y For vs. While donde el alumno a partir de una consigna dada elige una estructura de control y luego recibe un *feedback* como resultado de su elección (fig 1b).
- C. Actividad de integración: tiene como objetivo que el alumno aplique los conocimientos abordados previamente. Se trata de un juego en el que se propone al alumno que, de acuerdo a determinadas características personales, seleccione una estructura de control. Esto llevará a agregar en su rostro diferentes objetos virtuales (anteojos, sombrero) que provocarán un determinado efecto en la escena real que cambiará su aspecto (fig. 1c).

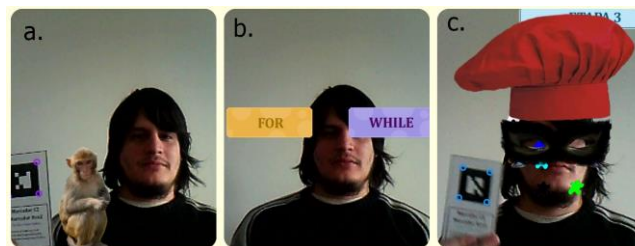


Fig. 1: Actividades de Realidad Aumentada (a) actividad de exploración (b) actividad de repaso (c) actividad de integración.

4 Experiencia realizada

La experiencia realizada consistió en la realización de sesiones de trabajo con el material EPRA, tanto con alumnos como con docentes de las cátedras mencionadas. La evaluación realizada se concentró en la satisfacción de alumnos y docentes en el trabajo con EPRA, la motivación intrínseca de los alumnos al realizar las actividades de RA, y el impacto en su aprendizaje, en términos de la aplicación de la estructura de control más adecuada en la resolución de problemas sencillos. En este trabajo sólo se focalizará en este último aspecto. Los otros ejes de evaluación serán motivo de otras publicaciones. En esta sección se presenta para dar contexto al análisis de los resultados, una descripción de las características de las asignaturas en las que se lleva adelante la experiencia, el perfil de los alumnos participantes y luego se detallan los resultados de la evaluación diagnóstica general para todos los alumnos, las sesiones de prueba llevadas a cabo con el grupo que realiza la experiencia con EPRA y los resultados específicos para los alumnos participantes de la experiencia.

4.1 Características de las asignaturas

La experiencia se realiza con alumnos regulares de dos asignaturas introductorias de la Facultad de Informática de la UNLP. Para el dictado de sus contenidos curriculares, cada una se organiza en clases de teoría y práctica. Las clases teóricas,

normalmente numerosas (alrededor de 200 alumnos por turno) son dictadas por el profesor y las clases prácticas (alrededor de 50 alumnos por comisión), están a cargo de los docentes auxiliares que acompañan en la resolución de los ejercicios prácticos los cuales proponen aplicar los contenidos vistos en la clase de teoría.

La asignatura Conceptos de Algoritmos, Datos y Programas (CADP), corresponde al primer año de las carreras de Lic. en Sistemas, Lic. en Informática y Analista Programador Universitario. Los alumnos de esta asignatura, han cursado previamente la asignatura Expresión de Problemas y Algoritmos (EPA), que se dicta en el curso de ingreso a las carreras, y que tiene como requisito de aprobación, el cumplimiento del 80 % de asistencia a clase.

En tanto que, la asignatura Programación 1 corresponde al primer año de la carrera de Ingeniería de Computación y a diferencia del grupo anterior, los alumnos que cursan esta asignatura, han aprobado previamente la asignatura Introducción a la Informática correspondiente al curso de ingreso a la carrera.

4.2 Características de los alumnos participantes y resultados de la evaluación diagnóstica en relación al aprendizaje de estructuras de control

Los alumnos que cursan actualmente las asignaturas ya citadas en su mayoría son del género masculino y una edad promedio de 19 años para CADP y 21 años para Programación 1. Se destaca que los alumnos participantes de la experiencia han abordado, previamente, en cada curso (CADP y Programación 1), el concepto de estructura de control (en particular, habían visto el If, While, Repeat y For), así como también han ejercitado su aplicación en la resolución de problemas simples.

Se realiza una evaluación diagnóstica en los alumnos de estas cátedras que tiene como objetivo principal analizar las principales dificultades detectadas en los alumnos, en relación a la comprensión del tema propuesto: uso y aplicación de las estructuras de control en la resolución de problemas sencillos. Esta evaluación se realiza en el horario de práctica de ambas asignaturas y participan todos los alumnos que han asistido a clase ese día. En total, 192 alumnos de CADP y 84 de Programación 1. Se trata de una evaluación en papel, con formato *multiple choice* en la cual se plantean cinco problemas sencillos. El alumno, de forma individual, debe elegir la estructura de control más conveniente para resolver cada uno de los problemas.

En los resultados se observa que en el uso de la estructura de control IF no se encontraron grandes dificultades. La muestra indica que alrededor del 79% de los alumnos contestó correctamente, mientras que el 21% eligió otra estructura de control en lugar del IF: el 9% eligió WHILE, el 5% FOR, el 2% REPEAT y el 5% no eligió ninguna de las opciones. Este mismo análisis se lleva a cabo con el resto de las estructuras de control trabajadas y los resultados se muestran a continuación en la figura 2.

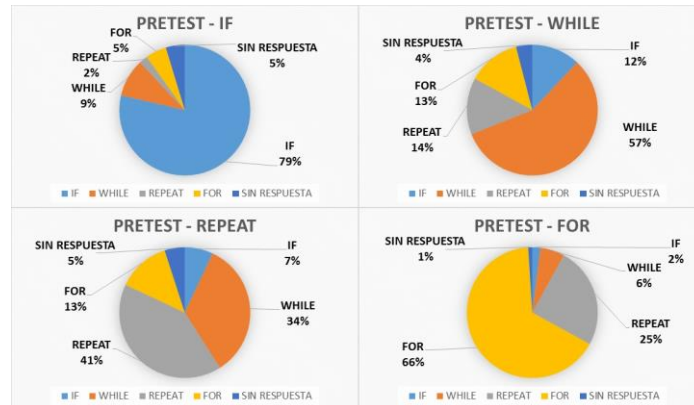


Fig. 2: Resultados de la evaluación diagnóstica de CADP

En la figura 2 se observa que un alto porcentaje de los alumnos se equivoca al utilizar las estructuras de control iterativas condicionales: WHILE y REPEAT. Principalmente confunden WHILE con IF, FOR y REPEAT en un porcentaje similar, mientras que el REPEAT lo asocian directamente con el WHILE.

Con respecto a los 84 pretest de Programación 1, los resultados respecto a las estructuras de control: iterativa precondicional (WHILE), repetitiva (FOR), y de decisión (IF) no muestran dificultades importantes. Sin embargo, la estructura poscondicional REPEAT alcanza el mayor porcentaje de errores al momento de aplicarla en un problema concreto. Por ejemplo, se plantea el siguiente problema: “Se lee el nombre, apellido y nota de alumnos hasta leer a ‘Juan Perez’. Elegir la estructura de control más adecuada para calcular el promedio de notas incluyendo a este último alumno.” El análisis de resultados arroja los siguientes porcentajes: el 57% contestó correctamente mientras que el 37% optó por WHILE, 4% por FOR, 1% por IF y 1% no eligió ninguna de las opciones. Se realiza el mismo análisis para las demás estructuras de control (fig. 3).

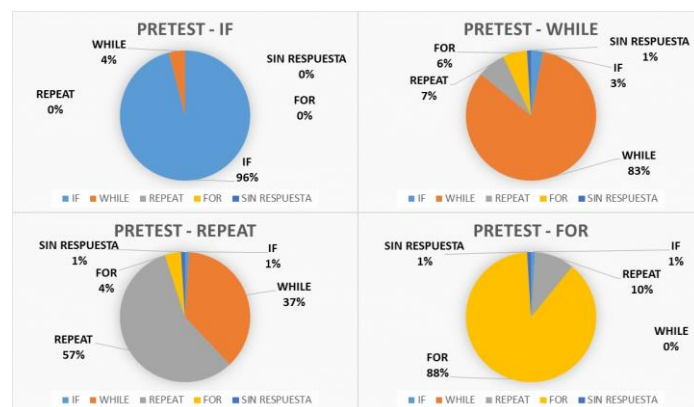


Fig. 3: Resultados de la evaluación diagnóstica de Programación 1

Los resultados presentados muestran que los alumnos presentan ciertas dificultades cuando deben seleccionar la estructura de control más adecuada para la resolución de un problema sencillo y estas falencias luego se visualizan en la resolución de los parciales donde se ve aún problemas en este sentido. Estos resultados han servido de motivación para el desarrollo y utilización del material educativo con actividades de Realidad Aumentada que aquí se presenta.

4.3 Sesiones experimentales realizadas

El grupo que realizó la experiencia con EPRA se conformó con la selección de 24 alumnos de CADP y 30 de Programación 1 que completaron la evaluación diagnóstica. La muestra es aleatoria y representativa de la población bajo estudio. La cantidad de alumnos participantes está condicionada por la disponibilidad de los recursos necesarios (aula, computadoras, cámaras web, etc.) para realizar la experiencia y también por la asistencia a la clase práctica, el día de la sesión experimental [12]. Se realizaron 4 sesiones con un máximo de 15 alumnos cada una, con una duración de aproximadamente 40 minutos por sesión, y en un aula acondicionada de la facultad en cuestión. Para la configuración del aula se tuvieron en cuenta aspectos relacionados con la disponibilidad de computadoras con acceso a internet y cámaras web, la adecuada iluminación y ubicación de las computadoras para evitar inconvenientes con la detección de los marcadores, entre otros. Los alumnos trabajaron de a pares y se les asignó una computadora. Cada sesión se desarrolló siguiendo los siguientes pasos: a. Los docentes presentaron el material educativo EPRA y explicaron cómo trabajar con él. Los alumnos realizaron las actividades de RA propuestas en el material EPRA y luego se les solicitó completar el postest, estructurado en forma similar a la evaluación diagnóstica.

5 Análisis de resultados

Se presenta el análisis realizado a partir de los resultados de las evaluaciones efectuadas: diagnóstica (que se toma como un pretest ahora sólo para los alumnos participantes en las sesiones) y postest (evaluación tomada en forma posterior a la realización de las actividades con RA). A continuación se analizan los resultados del postest en comparación con los obtenidos en la prueba diagnóstica para los mismos alumnos. El postest fue realizado por los 24 alumnos de CADP y los 30 alumnos de Programación 1, participantes en las sesiones experimentales. Los resultados se analizan por separado para cada grupo ya que las condiciones iniciales de cada grupo son diferentes y podrían influir en los resultados.

Respecto de los resultados de la muestra correspondiente a CADP para la estructura de control WHILE se observa que el 92% de los alumnos respondió correctamente mientras que sólo el 8% eligió otra estructura de control. El 4% optó por la estructura de control de decisión IF y el 4% por el FOR. Estos porcentajes demuestran una mejora respecto de lo que estos mismos alumnos habían realizado en la prueba diagnóstica, ya que los porcentajes para el WHILE eran: 38% había elegido

el WHILE, un 29 % el IF, el 21% el FOR, 8% el REPEAT, y un 4% no respondió La comparativa entre pretest y postest en relación al WHILE, se presenta en la figura 4.

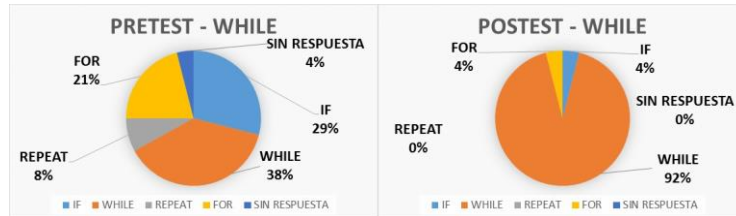


Fig. 4: Comparativa Pretest y Postest de WHILE para CADP.

En relación al FOR, también se observa una mejora en el postest respecto del pretest. En el pretest sólo un 54% de los alumnos seleccionó esta estructura como la más adecuada para resolver el problema, el resto eligió: 30% el REPEAT, un 12% el WHILE, y un 4% no respondió (figura 5).



Fig. 5: Comparativa Pretest y Postest de FOR para CADP.

En cuanto al IF, hay una leve mejora en esta estructura, se observa ya una mejor comprensión desde el pretest, ya que el 80% la ha seleccionado adecuadamente, y en el postest, aumenta al 88%. Finalmente, en cuanto al REPEAT, es en el que se observan mayores dificultades aún en el postest, a pesar de haber una leve mejora (figura 6).



Fig. 6: Comparativa Pretest y Postest de REPEAT para CADP.

Respecto de los resultados del postest de los alumnos de Programación 1 no se observan mejoras significativas en comparación a los pretest.

Sin embargo, algunos resultados indican una leve mejora en la selección de la estructura de control correcta para resolver un problema sencillo, mientras que en otras no se observan diferencias significativas o incluso se ha observado alguna disminución en el porcentaje de respuesta correctas. Se presentan sólo aquí los resultados correspondientes a la estructura de control WHILE donde se ha visualizado una mejora. El problema que involucraba responder eligiendo la estructura de control

WHILE, ha sido respondido con WHILE por un 80% de los alumnos en el pretest y en el postest por un 87% de los alumnos. Pueden verse estos resultados en la figura 7.

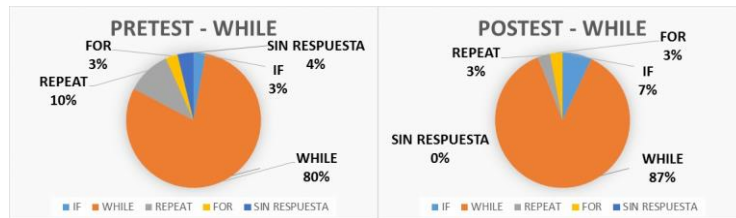


Fig. 7: Comparativa Pretest y Postest de WHILE para Programación 1.

En tanto para el IF las respuestas han sido correctas en un alto porcentaje tanto en el pretest como en el postest (rondando un porcentaje superior al 90%), el REPEAT es en la que presentan mayores dificultades con un porcentaje de respuestas correctas cercanas al 60% en el pretest, disminuyendo al 55% en el postest. Finalmente, en el FOR no se observan diferencias significativas entre el pretest y el postest, con respuestas correctas, superiores al 90% en ambos casos.

6 Conclusiones y trabajos futuros

Se presentó una experiencia que utiliza el material educativo hipermedial EPRA que incorpora actividades de RA para la aplicación de estructuras de control en la resolución de problemas sencillos, en cursos iniciales de programación. Para analizar si la realización de las actividades de RA han modificado de alguna manera el desempeño de los alumnos (incidencia), se seleccionó un grupo de alumnos, aleatoriamente en las cátedras antes dichas y se analizaron sus resultados particulares en los pretest, luego realizaron las actividades con RA, y se les administró un postest con características similares al pretest.

Los resultados difieren en el grupo correspondiente a CADP respecto del grupo de Programación 1. En estos cursos las condiciones iniciales son diferentes, ya que en el segundo caso los alumnos han aprobado los contenidos del curso de ingreso, lo que ya implica una mayor comprensión de los contenidos, incluyendo el trabajo con estructuras de control, mientras que en CADP, el grupo incluye alumnos que han aprobado los contenidos del ingreso y otros que no. Es decir el grupo es más heterogéneo. La única estructura de control no trabajada por ninguno de los dos grupos durante el ingreso ha sido la estructura REPEAT, que es en la que se visualiza mayor dificultad en ambos grupos, aunque en el grupo de CADP se ha visto una mejora en su aplicación a partir del trabajo con EPRA. En el grupo de CADP se ha visto una mayor incidencia del uso de las actividades de RA en la capacidad del alumno por elegir la estructura de control más adecuada para resolver un problema. Esto se ha dado en todas las estructuras de control. En el grupo de Programación, se ha visto ya mejores resultados en el pretest, salvo para el REPEAT y el WHILE. En el WHILE se ve una incidencia positiva a partir de la realización de las actividades RA, mientras que no se ha visualizado una mejora en el REPEAT. Esto lleva a reflexionar por un lado, sobre la necesidad de profundizar en la diferenciación del uso de la

estructura REPEAT respecto de las otras estructuras. Al mismo tiempo, se considera a las actividades de RA diseñadas, importantes en términos del aprendizaje de los alumnos, especialmente aquellos que recién se inician en la comprensión de las estructuras de control, ya que se ha visualizado una mejora mayor en alumnos con estas características y al mismo tiempo en su motivación (aspecto presentado en [12]).

Aún resta profundizar en estos resultados a partir de la conjugación de los tres ejes de evaluación propuestos: esto es motivación, satisfacción, e incidencia en el aprendizaje, pero se cree que los resultados ya obtenidos son de interés y un punto de partida para otros docentes e investigadores en la temática.

Referencias

1. Johnson, L., Adams, S., & Cummins, M.: The NMC Horizon Report: 2012 Higher Education Edition. Austin, Texas: The New Media Consortium (2012).
2. Chen, C. M., & Tsai, Y. N.: Interactive augmented reality system for enhancing library instruction in elementary schools. *Computers & Education*, 59, 638-652. doi:10.1016/j.compedu.2012.03.001 (2012).
3. Cabero, J., Barroso, J.: The educational possibilities of Augmented Reality. *Journal of New Approaches in Educational Research*, 5(1), 44-50. doi: 10.7821/naer.2016.1.140, (2016).
4. García García, F., Gértrudix Barrio, F., Durán Medina, J., Gamonal Arroyo, R., Gálvez de la Cuesta, M. C.: Señas de identidad del 'nativo digital'. Una aproximación teórica para conocer las claves de su unicidad. *CDM Cuaderno de Documentación Multimedia*. V.22 ISSN: 1575-9733, (2011).
5. Prensky, M.: Digital Natives, Digital Immigrants P1. *On the Horizon*. V.9. pp. 1-6, (2001).
6. Salazar Mesía, N., Sanz, C., Gorga, G.: Experiencia de enseñanza de programación utilizando Realidad Aumentada. XXII Jornadas sobre la Enseñanza Universitaria de la Informática, Actas de las XXII Jenui. pp. 213-220. Almería, (2016).
7. Tamim R. M., Bernard, R.M., Borokhovski, E., Abrami, P. C., Schmid, R. F.: What Forty Years of Research Says about the Impact of Technology on Learning a Second-Order Meta-Analysis and Validation Study. *Review of Educational Research*, V.81, No. 1, pp. 4-28, (2011).
8. Roussou, M.: Learning by Doing and Learning Through Play: An Exploration of Interactivity in Virtual Environments for Children. *Computers in Entertainment (CIE) - Theoretical and Practical Computer Applications in Entertainment*, 2 (1), pp.1-23, (2004).
9. Sommerauer, P., Müller, O.: "Augmented reality in informal learning environments: a field experiment in a mathematics exhibition", *Computers & Education*, V.79, pp. 59-68, (2014).
10. Salamanca Díaz, D. M.: Creación de contenido educativo con realidad aumentada aplicando los principios de la teoría cognitiva del aprendizaje multimedia. Estudio comparativo para enseñar cómo volar un dron (cuadrícóptero). *Computing Colombian Conference (10CCC)*, 10th, Bogota, pp. 456-462, ISBN: 978-1-4673-9464 (2015).
11. Salazar Mesía, N., Gorga, G., Sanz, C.: EPRA: Herramienta para la Enseñanza de conceptos básicos de programación utilizando realidad aumentada. X Congreso de Tecnología en educación y Educación en Tecnología. ISBN 978-950-656-154-3, (2015).
12. Salazar Mesía, N., Gorga, G., Sanz, C.: Plan de evaluación del material educativo digital EPRA. Propuesta de indagación sobre la motivación intrínseca. XXI Congreso Argentino de Ciencias de la Computación CACIC. pp. 414-423. ISBN: 978-987-3724-37-4, (2015).
13. Adell, J., Castañeda, L.: Tecnologías emergentes, ¿pedagogías emergentes? En J. Hernández, M. Pennesi, D. Sobrino & A. Vázquez (Coords). *Tendencias emergentes en educación con TIC*. pp.18-63, Barcelona: Editorial espiral, (2012).

Estrategias Basadas en Robótica para Apoyar el Pensamiento Computacional

Maritza A. García¹, Claudia Deco², César A. Collazos³

¹ I.T.I. Pascual Bravo. Medellín, Colombia

²Universidad Nacional de Rosario, Rosario, Argentina

³Universidad del Cauca, Popayán, Colombia

magangarita@gmail.com, deco@fceia.unr.edu.ar, ccollazo@unicauca.edu.co,

Abstract. En la actualidad existen diferentes empresas públicas y privadas que apoyan al gobierno en la iniciativa de llevar los procesos de investigación e innovación a las aulas con el fin de generar un conocimiento que propicie grandes cambios en la sociedad y que a la vez ayude a mitigar algunas problemáticas nacionales. Esto genera grandes cambios al interior de la escuela y del estudiante, logrando en él, aprendizajes significativos en varias áreas del conocimiento como ambiental, física, electrónica, matemática, programación, mecánica, entre otras. En este artículo se presentan algunas estrategias, como son los clubes de robótica, y de cómo su metodología de enseñanza basada en problemas, colabora en el desarrollo de competencias de varios tipos y al logro del pensamiento computacional en los estudiantes. Además, se mencionan los resultados del proyecto desarrollado en el Instituto Técnico Industrial Pascual Bravo de la ciudad Medellín-Colombia.

Keywords: Robótica, Gamificación, Pensamiento computacional.

1 Introducción

En Colombia, así como en la mayoría de los países, es notable el bajo nivel académico en educación básica y secundaria como lo indican las pruebas PISA (www.oecd.org/pisa/home/) y las pruebas estatales ICFES (Instituto Colombiano para el Fomento de la Educación Superior) según Caro [3] en el documento “Resumen 2005 al 2009 del ICFES”, que obligan a repensar la educación de una forma diferente, donde es urgente cambiar el modelo tradicional por estrategias y herramientas que generen gran motivación en los estudiantes. Por su parte, Contreras [12] muestra diferentes formas de aplicar el concepto de robótica usando “el aprendizaje por proyectos” o “aprendizaje basado en problemas” que con las TIC podrían generar grandes avances en los procesos de enseñanza-aprendizaje. Para complementar esta idea, ya existen muchas investigaciones acerca del uso de éstas que de una forma sencilla y práctica generan grandes cambios en toda la comunidad educativa. Por su parte el Ministerio de Educación y otras empresas del sector público y privado, han emprendido diferentes estrategias como medida para incentivar la investigación y la innovación no sólo en las Universidades sino además en los

Colegios sin distinto alguno.

Una de estas estrategias consiste en clubes de robótica e innobótica, donde el Instituto Técnico Industrial Pascual Bravo participó con estudiantes de diferentes grados y especialidades. En las Secciones 2 y 3 se abordan definiciones importantes acerca del pensamiento computacional y la robótica educativa; así como algunas experiencias importantes con niños y adolescentes en la escuela donde se exponen algunos logros significativos que buscan una educación integral y competitiva. En la Sección 4 se describen las diferentes competencias que pueden ser desarrolladas en proyectos de este tipo. En la Sección 5 se describen varias estrategias de la Secretaría de Educación en donde el colegio ha participado. En la Sección 6, se detalla el proceso abordado en el proyecto de robótica y las etapas que a grandes rasgos deja ver la importancia de la robótica educativa y la competencia académica en la adquisición de conocimiento en áreas de programación y electrónica. Finalmente, se presentan las conclusiones del proyecto y de la importancia de trabajar con herramientas y/o estrategias para el trabajo en equipo y el desarrollo de competencias de todo tipo.

2 Generalidades sobre Robótica Educativa y Pensamiento Computacional

La robótica educativa según Acuña [1], hace su aparición en los años 60, cuando algunos investigadores como Symourt Papert, Marvin Minsky y Mitchael Resnik, propusieron la construcción de robots didácticos con el apoyo de la empresa LEGO, para que los niños interactuaran con éstos [4]. Cabe mencionar que esta empresa lidera el mercado de robots educativos, cuya variedad va desde robots para uso de niños hasta adultos. Luego, en los años 80 se popularizaron estos dispositivos, logrando incluirlos dentro de actividades en las escuelas por medio de proyectos importantes que permitieron que el estudiante obtuviera grandes competencias en áreas como matemática, informática, electrónica y mecánica; a la vez que se divierten y juegan al proponer soluciones a problemas específicos.

Por otro lado, el pensamiento computacional goza de gran importancia en nuestros días ya que permite el desarrollo de competencias y como expresa Wing ([6], [7]) es una habilidad fundamental para todo el mundo y no sólo para estudiantes y profesionales del área de la computación, ya que es un enfoque que permite resolver problemas de una forma innovadora. El pensamiento computacional genera en el estudiante la capacidad de realizar varios tipos de análisis como el ascendente-descendente, de ser creativos, de generar un pensamiento divergente-abstracto que permita abordar un problema de forma más sencilla y creativa. Este pensamiento es de alto nivel de complejidad ya que está relacionado con el desarrollo de ideas y con el pensamiento abstracto-matemático e ingenieril, pero a su vez puede ser usado en múltiples escenarios. Cabe mencionar que no depende de un lenguaje de programación o software específico sino de las habilidades que nos permiten enfrentar problemas con más claridad como lo menciona la autora en [7]. Existe ya un planteamiento de Papert [8] en los que se desea vincular las tecnologías de información con el construccionismo, ya que por ejemplo, el pensamiento computacional obliga no sólo a la apropiación de conceptos sino que además los objetos de estudio (programas, animaciones, videojuegos, robots) materializan de alguna forma el aprendizaje logrando un enriquecimiento del mismo.

La robótica educativa así como el pensamiento computacional en los últimos años han ido de la mano gracias a la incursión de ésta en el aula, quedando reflejado en muchas investigaciones a nivel no sólo de educación básica y secundaria, sino universitaria. Este tipo de proyectos, como dice [1], es común encontrarlos en programas nacionales o políticas educativas de ciudades o países. Por ejemplo existen iniciativas nacionales como en Costa Rica o a nivel local en algunas ciudades de países como México, Brasil, Chile, Colombia y Estados Unidos.

3 Competencias a desarrollar en Estudiantes de Educación Básica, Media y Técnica

Existe ya una gran cantidad de proyectos de investigación orientados a la adquisición de competencias gracias al uso de robots y lenguajes de programación de todo tipo. Estos proyectos proponen generar nuevas estrategias para que desde pequeño el estudiante pueda adquirir ciertas destrezas que lo ayuden más adelante en su proceso de formación universitaria. A continuación se describen algunos de estos proyectos:

- A. *Programación para Niños y Adolescentes.* Actualmente es posible encontrar varios entornos de programación para niños de los que podemos destacar Alice¹ y Scratch². Scratch por ejemplo, permite realizar proyectos personales como crear *juegos, contar historias y realizar animaciones. Éste se construye sobre las ideas* del constructivismo de Logo de Kafai and Resnick [4], Papert [8] y Etoys de Kay [2]. Uno de los aspectos más importantes de Scratch es que incentiva el aprendizaje autodidacta a través de la práctica personal y de la colaboración con otros, como afirman Maloney y Resnick [4]. Este proceso de aprendizaje haciendo ‘artefactos’ o ‘programas’ permite a los alumnos construir y comprobar, reconstruir, modificar, mejorar objetos de aprendizaje como plantea Papert [8].
- B. *Clubes de Robótica.* Por ejemplo en Costa Rica, como plantea Jiménez [9], se crean “Clubes de Robótica” para incentivar el estudio de las ciencias y la tecnología en algunas instituciones primarias y secundarias con sesiones de trabajo de al menos 40 horas. La metodología usada consistió básicamente según este mismo autor, en generar un ambiente de trabajo agradable, desafiante, generador de pensamiento crítico, que incentiva el trabajo en equipo y de tipo experimental, donde se contemplaban actividades de análisis, diseño, construcción y programación en Robolab 2.9 o NXT-G. Además el trabajo basado en proyectos ayuda a desarrollar varias competencias transversales como la planificación, la comunicación y la creatividad; donde algunos de los retos propuestos tenían que ver con la selección y clasificación de objetos o identificación y seguimiento a patrones de color, sonido, luz y distancia. En dicha investigación se analizaron tres dimensiones: desempeño de los estudiantes, habilidades en diseño y construcción y habilidades en programación,

¹ Lenguaje de programación educativo libre y abierto orientado a objetos con un entorno de desarrollo integrado (IDE), programado en Java, utiliza un entorno sencillo basado en «arrastrar y soltar» para crear animaciones mediante modelos 3D.

² Lenguaje de programación visual libre orientado a la enseñanza mediante creación de juegos.

donde se muestran resultados importantes que hacen pensar que esta estrategia goza de gran acogida en el estudiante.

- C. *Universidad de Cundinamarca y Robótica Educativa*. Como referencia Márquez [5], el grupo de investigación de Nano Ingeniería de la Universidad de Cundinamarca junto con colegios ubicados en la Sabana de Bogotá, decide crear un curso libre de robótica básica, que se programó de manera presencial, apoyado con información virtual, soportada en la plataforma Moodle, donde se facilitó material de temas de electrónica básica, que era complementado en clase. Los robots construidos por los estudiantes pertenecen a la categoría de “robots autónomos seguidores de línea”, que como su nombre lo indica deben seguir un circuito marcado por una línea donde se califica el recorrido en el menor tiempo posible. En total se construyeron seis robots analógicos cien por ciento funcionales, que demostraron la viabilidad de implementar la robótica en el aula. En cuanto a los resultados observados, son similares a la investigación presentada anteriormente, tanto en el aspecto de la construcción del robot, como en la motivación de los estudiantes que destaca el compromiso, la apropiación del conocimiento, el desarrollo y consecución de las tareas.

4 Algunas Experiencias en Latinoamérica

Para entender la importancia de los campeonatos de robótica en la educación secundaria, en la adquisición de competencias y de desarrollo de pensamiento computacional es indispensable revisar lo que Marqués [11] plantea como competencias tecnológicas indispensables a desarrollar como son el manejo de las plataformas virtuales, manejo de herramientas ofimáticas, de herramientas de comunicación como email, de diseño y construcción de sitios web en diversos contenidos y niveles escolares, así como la búsqueda y análisis de la información.

Además de estas competencias, también el Ministerio de Educación Nacional [10] ha fijado unas competencias que el Sistema Educativo Colombiano debe desarrollar en los estudiantes. Estas son de tres clases: básicas, ciudadanas y laborales.

Dentro de las *Competencias Básicas* se tienen las de pensamiento tecnológico que hacen alusión a la construcción de conceptos de tecnología, para ponerlos a prueba mediante el diseño y la construcción de prototipos. Junto a ésta está la técnica que permite el conocimiento de herramientas que posibiliten el desarrollo de actividades y por último la comunicativa que permea estas dos generando trabajo interdisciplinario en pro de los desarrollos. Por otro lado, se tienen las *Competencias Ciudadanas* que permiten una sana convivencia no sólo en la vida escolar sino en otros contextos. Para finalizar tenemos las *Competencias Laborales* que se subdividen en dos. Dentro de las competencias Laborales Generales están las *Intelectuales* referentes a la toma de decisión y solución de problemas; las *Personales* que tienen que ver con el comportamiento y actitudes esperados en los ambientes productivos; las *Interpersonales* que generan adaptación a los ambientes laborales como el trabajo en equipo, liderazgo, manejo de conflictos, entre otros. De la misma forma tenemos las *Organizacionales* que se refieren a la habilidad para aplicar el pensamiento estratégico en diferentes situaciones de la empresa, como la gestión de la información, orientación al servicio, entre otras. Por último tenemos las *Tecnológicas* que

permiten a los jóvenes identificar, transformar e innovar procedimientos; y las *Empresariales y para el emprendimiento* que son útiles para que se puedan crear, liderar y sostener unidades de negocio por cuenta propia.

Por último y específicamente en la especialidad de Programación y Mantenimiento de computadores, del Instituto Técnico Industrial Pascual Bravo, el estudiante debe desarrollar competencias en el manejo de lenguajes como C++, Java, Visual Studio lo que implica el desarrollo de lógica de programación y el uso de conceptos como el de programación orientada a objetos. De la misma forma, es indispensable que el estudiante conozca el proceso de modelado, diseño y creación de bases de datos y páginas web dinámica. Para finalizar en el grado undécimo el estudiante debe conocer la parte básica del hardware que hace referencia a la electrónica básica y a la digital donde los simuladores son herramientas indispensables en el proceso de diseño de desarrollos sencillos. Todas estas competencias podrían ser más fácilmente obtenidas por el estudiante si se usaran otras estrategias, herramientas o mecanismos en el aula, como los descritos anteriormente, que motiven al estudiante y lo hagan partícipe de su propio proceso educativo.

5 Competencias de Robótica: Caso de Estudio

El Instituto Técnico Industrial Pascual Bravo de Medellín, se ha venido vinculando en los últimos años en procesos de investigación implementados por Secretaría de Educación y empresas del sector público y privado; que pretenden brindarle al estudiante conocimientos en diferentes áreas que aporten a buscar soluciones a problemáticas de la ciudad o que apunten a la innovación. Una de estas iniciativas fue el Programa Ondas que es apoyado por el Estado y que ha querido orientar a los niños, para que aprendan a investigar de una manera sencilla y divertida. Este programa al igual que muchos otros, tienen unas etapas que son orientadas desde el comienzo por el capacitador del programa y que obedece claramente a una metodología de investigación propia de los proyectos de investigación y que son recorridas paso a paso en cada etapa de formación no sólo de los estudiantes sino del docente acompañante. En cuanto a este programa en el año 2012, el Instituto técnico se vinculó con cinco proyectos que fueron acompañados por las docentes del área de Sistemas. Estos proyectos se enfocaron básicamente en la parte ambiental, en la reconstrucción de la historia de la institución y un proyecto orientado al reconocimiento del software libre educativo.

Otra de las propuestas de esta ciudad, que incorpora a Ruta N y la alcaldía de Medellín, es el proyecto llamado “Aventura Horizontes”, en el cual se incluían varios programas para incentivar a los estudiantes a desarrollar sus ideas ya fueran de robótica, nuevas alternativas para cuidar y proteger el medio ambiente y en general creación de proyectos innovadores para solucionar problemáticas de ciudad. Los programas que se incluyeron en esta propuesta fueron:

- Innobótica: el cual pretendía incentivar a los jóvenes de las instituciones a crear y diseñar sus propios robots para competencias y para solucionar problemáticas existentes, entre otros fines. Esta iniciativa contó con el acompañamiento de la empresa Pygmalion Robotics, la cual enviaba a sus asesores para enseñar robótica en

los colegios, donde se daban unos kits con todos los implementos necesarios para abordar un proyecto sencillo de robótica.

- Ingenieríaⁿ: pretendía el desarrollo de proyectos y sus prototipos para ayudar al medio ambiente en diferentes problemas de ciudad que incluía un encuentro cada 15 días en la Universidad EAFIT, que posee excelentes laboratorios para el desarrollo de prototipos contando con impresoras 3d y diferentes máquinas de alta tecnología.
- Interchange: este programa pretendía hacer algo muy parecido a Ingenieríaⁿ, sólo que cambiaba su sede ya que se realizaba en el SENA (Servicio Nacional de Aprendizaje). Además pretendía llevar a los jóvenes a hacer intercambios a otros países, por sus ideas y desarrollo de proyectos.

Estas estrategias han dado sus frutos en los últimos años, con la participación en los diferentes programas, siendo el de robótica el de mayor continuidad por parte de un grupo de estudiantes que además de incentivarse por la adquisición de conocimiento de áreas como electrónica, sistemas, mecánica, entre otras; se motivaron al desarrollo de competencias que en un aula convencional serían imposibles de lograr (ver Figura 1).

La participación del Instituto en este proceso de forma pública se inició hace dos años, en un evento de robótica en Octubre del 2014, en la semana de la robótica y la innovación en su segunda versión, en Medellín. Esta competencia es organizada por la empresa de robótica *Pygmalion* de esta ciudad y la organización *RoboRave* International la cual es de Albuquerque - Nuevo México, Estados Unidos. Antes de este momento, la docente a cargo de los grupos de investigación, del área de Sistemas del Instituto, hace un llamado a toda la comunidad a participar en varios grupos de investigación y en diferentes propuestas de Secretaría de Educación en diferentes campos del saber. A partir de este momento, surge una gran motivación en los estudiantes por aprender acerca de robótica y sus áreas fundamentales. Por tal razón se aprovechan los programas de formación ofrecidos por *Ruta N* llamado *Innobotica*. Este programa era ofrecido a muchos colegios de Medellín sin distinción de estrato, ni número de participantes. Sólo se exigía la continuidad en el proceso para que los equipos pudieran competir en la semana de la robótica, donde los ganadores tenían la posibilidad de ir a competir a Albuquerque – Estados Unidos a la competencia mundial, en la cual competían equipos de todo el mundo. Para esta primera etapa el Instituto participó con 3 equipos con estudiantes de varios grados y de diferentes Técnicas o Especialidades (Programación, Electricidad, Mecánica entre otros) con las que cuenta dicho centro de formación; donde según el nivel de exigencia se decidió competir en dos retos, uno llamado “Seguidor de Línea más Pelotas” y el otro “Reto Pygmalion”, donde el seguidor de línea consistía en hacer un robot el cual siga una línea y supere dos intersecciones y lleve la mayor cantidad de pelotas posibles en un tiempo determinado para depositarlas en una torre (ver Figura 1).

En esta competencia el robot debía ser totalmente autónomo, razón por la cual el nivel alcanzado no fue el mejor debido a la complejidad del mismo. Obviamente por ser el primer año de competencia y por la poca capacitación recibida hasta el momento, el rendimiento no fue el mejor, pero sí logro motivar a los estudiantes de forma que crearon vínculos con otros grupos donde escucharon gran cantidad de consejos útiles para próximas competencias.



Fig. 1. Montaje de Reto Pygmalion

Vale la pena resaltar que en esta competencia se usó un kit llamado *Innobot* que nos daba la empresa Pygmalion por medio de Ruta N. Este paquete estaba conformado por una réplica de un integrado *Arduino* Leonardo, que contaba con 4 motores reductores, 4 llantas, 4 sensores (para seguir la línea) y un sensor ultrasónico para detectar objetos, entre otras; acompañada de una estructura hecha en plástico. En cuanto al software se refiere, el robot debía ser programado en lenguaje C por medio del software de Arduino. El “Reto Pygmalion” consistía en usar el mismo robot pero esta vez el robot debía seguir una pista octagonal, superando varias intersecciones, donde al mismo tiempo debían reconocer unos tótems y sacarlos de la pista hasta lograr que la pista estuviera libre de obstáculos según se muestra en la Figura 1.



Fig. 2. Base de orugas obsequiada por el equipo de EU

Como es de entender en esta prueba también se tuvieron algunas dificultades, en este caso con la parte electrónica, específicamente con los sensores del robot, lo que conllevó a que sólo pudieran competir una vez en los 5 días que duraba la competencia. Finalmente los ganadores de los tres retos con los que contaba la competencia (seguidor de línea, reto Pygmalion, Reto fuego) fue un equipo de Las Cruces – Nuevo México, de Estados Unidos, El San Ignacio de Loyola y una institución de Envigado. A pesar de no haber obtenido los primeros puestos, se lograron grandes vínculos entre la institución de Estados Unidos y los estudiantes del Pascual Bravo que fortalecieron el proceso académico y donde se intercambió información y dispositivos electrónicos para una posterior competencia. Gracias al equipo estadounidense, que obsequio al Pascual Bravo una “Base de oruga”, que a su vez contada con un puente H (circuito que permite a un motor DC girar en ambos sentidos, avance y retroceso) para el mejor funcionamiento de los motores en cuanto a potencia y capacidad de giro, se pudo iniciar un nuevo proyecto

contando ahora con mejores dispositivos eléctricos y electrónicos que le harían mejorar el rendimiento al robot (ver Figura 2). Es ahí que a partir de este momento se inicia un nuevo reto por representar mejor a la institución y a Medellín en dichas competencias.

2 Solución Propuesta

Antes de iniciar el proyecto es explicado con detalle a los estudiantes por parte de los capacitadores del proyecto, trazando una ruta del proceso para la realización del producto final. A continuación, se detallan algunos de los pasos llevados a cabo en este proceso para la participación de los eventos competitivos. Estos son:

1) *Capacitación en el uso de elementos electrónicos:* En esta fase un facilitador o docente, pertenecientes al programa que patrocina *Ruta n y Pygmalion*, asistía 2 horas semanales a los colegios donde existían equipos para enseñar a los estudiantes lo que se refiere a fundamentos de robótica, ambiente de programación del Arduino, partes mecánicas del robot como engranajes y poleas, así mismo capacitar en el manejo de la parte electrónica como manejo de baterías, resistencias, sensores, motores, entre otros.

2) *Reconocimiento del dispositivo Arduino:* Para esto fue necesario que el capacitador orientara un poco en las ventajas y desventajas de usar dispositivos Arduino o específicamente *Arduino Mega*. Algunas de estas ventajas son el hecho de tener una Plataforma Abierta, de que sus Tarjetas son fabricadas caseramente, además de su facilidad de programación y gran cantidad de información en la red. En el caso del Arduino Mega, tiene importantes características como son el número de pines de entrada y salida para el uso de diferentes sensores y motores para el control de giro del robot o la manipulación de objetos. Además de una buena capacidad de almacenamiento que lo hace indispensable a la hora de generar algoritmos predictivos que lo hagan más efectivos para cualquier prueba. También el tamaño del chip es ideal para este tipo de proyectos así como su bajo consumo de energía.

3) *Capacitación en la programación del Robot:* Se realizaron pequeñas pruebas como el giro de motores, avanzar y retroceder, reconocimiento de obstáculos, entre otros. Para esto, primero se debe instalar los drivers, luego se abre la aplicación y se selecciona el Arduino con el cual se va a trabajar. Luego se procede a programar y a descargar el código en el chip. Para esto el Arduino cuenta con funciones predefinidas, basadas en C, que hacen posible una programación intuitiva. Por ejemplo, en el código que se muestra a continuación es fácil ver que existen dos funciones, la primera hace referencia a la configuración del dispositivo (setup) y la segunda donde se implementa lo que debe realizar el integrado (loop), en este caso el led debe prender y apagar con pausas de 1 segundo.

```
int ledPin = 13;
void setup () {
  pinMode(ledPin, OUTPUT); }
void loop() {
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
```

```
delay(1000); }
```

Como se mostró con el código anterior, la programación de este dispositivo es muy sencilla. Lo realmente interesante es lograr la armonía entre el software y el hardware para lograr que los procesos se lleven a cabo con la mayor velocidad y exactitud.

4) *Análisis del proyecto*: En cuanto al uso de robótica educativa en la adquisición de competencias y desarrollo de pensamiento computacional, se puede decir:

a) Se observa grandes logros en los estudiantes que continúan el proceso en el club de robótica, sobretodo en la adquisición de conocimientos en la parte eléctrica y electrónica, así como en la parte mecánica y de programación, que se evidencia en el lenguaje usado y el desarrollo propio del proyecto. Con lo cual se puede afirmar que también desarrollaron competencias como las siguientes:

- **Intelectuales**: ya que se desarrollaron habilidades en la toma de decisiones y creatividad para la solución de este proyecto.
- **Personales e Interpersonales**: se resume en las competencias logradas en las ciudadanas.
- **Organizacionales**: en la gestión de los planos e información recolectada como fotografías y medidas, orientación a la construcción de la animación.
- **Empresariales y para el emprendimiento**: con la realización de este proyecto los estudiantes comprobaron que pueden realizar distintos tipos de proyectos en el ámbito de la robótica.

b) En cuanto a la motivación de los estudiantes para el desarrollo del proyecto, fue realmente buena. Todo el tiempo se observó disposición en todas las etapas del proyecto y trabajo en equipo, aspectos que los han llevado a tener apoyo por parte de diferentes entidades públicas y privadas, para competir en el primer semestre del 2016 en Estados Unidos.

c) En cuanto al desarrollo del proyecto, se nota en un estado muy avanzado, que les ha permitido participar en el concurso del presente año. Así mismo ya han representado al instituto en diferentes eventos logrando grandes resultados que han permitido poner a punto dicha máquina.

d) En cuanto a las relaciones docente/estudiante, docente/asesor y estudiante/estudiante, se observaron varios cambios de comportamiento, entre los que se destacan una conversación más cordial, respetuosa e incluso con uso de un vocabulario más técnico. Estas competencias se refieren a las ciudadanas y comunicativas en cuanto a competencias básicas. Por otro lado, se puede observar que con este proyecto se desarrollan competencias transversales en otras áreas, como es el caso del área de las matemáticas, electrónica, mecánica y programación.

7 Conclusiones

En esta investigación pudo observarse la motivación no sólo de estudiantes y docentes, sino de la comunidad educativa, que quedaron impresionados con la calidad y la dificultad del proyecto. De la misma manera se generan precedentes para que a través de estos clubes de ciencia o investigación, los estudiantes y docentes piensen en participar, para

que por medio de estos proyectos de tipo grupal potencien el trabajo autónomo y multidisciplinario. Para finalizar, lo que se pretende con este tipo de clubes es fomentar por medio de varias estrategias, que el estudiante se motive hacia la adquisición de conocimiento de una forma divertida, práctica, sencilla y diferente, que lo lleve a generar un aprendizaje sólido y transversal en diferentes áreas del conocimiento. Actualmente se trabaja en una investigación acerca de la posibilidad de utilizar software libre en la enseñanza- aprendizaje del área de tecnología e informática y en la utilización de programas como Scratch para el desarrollo de la lógica en estudiantes de programación.

Referencias

1. A. Acuña, Robótica y aprendizaje por Diseño en Educación, 2004 [Online]. Available: <http://www.educoas.org/portal/bdigital/lae-duccion/home.html>.
2. A. Kay, Squeak toys, children, and learning, 2010 [Online]. Available: <http://www.squeakland.org/resources/articles/>
3. B. Caro, Examen de Estado de la educación media, 2011: Resultados del período 2005/2010 [Online]. Available: www.icfes.gov.co
4. J. Maloney & M. Resnick & N. Rusk, The Scratch Programming Language and Environment, 2010 [Online]. Available: <http://doi.acm.org/10.1145/1868358.1868363>
5. J. Márquez & J. Ruiz, Robótica Educativa aplicada a la enseñanza Básica secundaria, Revista Didáctica, Innovación y multimedia, 2014.
6. J. Wing, Computational thinking. Communications of the ACM, 2006 [Online]. Available: <http://dx.doi.org/10.1145/1118178.1118215>
7. J. Wing, Research notebook: Computational thinking—What and why? The Link Magazine, Spring. Carnegie Mellon University, Pittsburgh, 2011 [Online]. Available: <http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why/>
8. S. Papert, Situating constructionism. In I. Harel & S. Papert (Eds.), Constructionism. 1-11. Norwood, NJ: Ablex, 1991.
9. M. Jiménez & R. Cerdas, La Robótica educativa como agente promotor del estudio por la ciencia y la tecnología en la región atlántica de Costa Rica, Congreso Iberoamericano de Ciencia, Tecnología, Innovación y Educación, 2014.
10. MEN, Articulación de la educación con el mundo productivo: Competencias laborales generales, 2007.
11. P. Marqués, Los Docentes: Funciones, Roles, Competencias Necesarias, Formación, Departamento de Pedagogía Aplicada, Facultad de Educación, UAB, 2000 [Online]. Available: http://www.uaa.mx/direcciones/dgdp/defaa/descargas/Docentes_funciones.pdf
12. R. Contreras & J. Eguia, Gamificación en aulas universitarias, Universidad Autónoma de Barcelona, 2016 [Online]. Available: http://incom.uab.cat/download/eBook_incomuab_gamificacion.pdf
13. W. Ruiz & A. Aguilera & C. Fúquene, Aprende jugando: el uso de técnicas de gamificación en entornos de aprendizaje, Universidad de La Salle, Colombia [Online]. Available: <http://revistas.lasalle.edu.co/index.php/im/article/viewFile/2708/2465>.
14. Y. Kafai, Learning design by making games: Children's development of design strategies in the creation of a complex computational artifact, 1996.

Herramientas Educativas para la Enseñanza de la Lógica

Maria Virginia Mauco¹, Laura Felice¹, María Carmen Leonardi¹

¹Facultad de Ciencias Exactas
Universidad Nacional del Centro de la Provincia de Buenos Aires
Tandil, Argentina
{vmauco, lfelice, cleonard}@exa.unicen.edu.ar

Resumen. El software educativo ayuda a motivar y mejorar el proceso de enseñanza/aprendizaje. En este artículo se presenta un proyecto de desarrollo de herramientas didácticas, interactivas y visuales para cursos de Lógica dictados en los primeros años de carreras de Informática. Las herramientas fueron desarrolladas por alumnos de 2º año de la carrera como Trabajo Final de dos materias, una que introduce conceptos de Lógica para Computación y otra que aborda las bases de las técnicas de diseño de algoritmos, permitiéndoles integrar y profundizar en los contenidos de ambas.

Palabras clave: Software Educativo; Lógica Proposicional; Lógica de Predicados de Primer Orden; Técnicas de Diseño de Algoritmos

1 Introducción

La mayoría de las carreras de Computación incluyen el dictado de cursos básicos de Lógica. En estos cursos, los alumnos deben resolver gran cantidad de ejercicios para adquirir práctica en el manejo de formalismos. El uso de herramientas didácticas que acompañen el proceso de enseñanza/aprendizaje, sin requerir demasiado tiempo de estudio para su utilización, puede ser un gran aporte.

El plan de estudios de la carrera de grado Ingeniería de Sistemas de nuestra Facultad incluye un curso introductorio en Lógica Proposicional (LP) y Lógica de Predicados de Primer Orden (LdPPO) [1, 2] que se dicta en el primer semestre del segundo año de la carrera [12]. En un principio, la ejercitación práctica de los temas enseñados en este curso se realizaba con papel y lápiz, siendo difícil, y en muchos casos tedioso para los estudiantes detectar y corregir errores. En consecuencia, los estudiantes consideraban a estos cursos menos atractivos, y en algunos casos más difíciles que otros en los que interactuaban con la computadora. Como el feedback inmediato, la interacción y la visualización pueden ayudar a motivar y mejorar el proceso de enseñanza/aprendizaje, se consideró la posibilidad de brindar a los estudiantes herramientas didácticas que se ajustaran a los contenidos, la notación y la metodología seguidos en el curso, y que fueran sencillas e intuitivas en su uso.

También, el plan de estudios de la carrera incluye en el primer semestre del segundo año un curso de Análisis y Diseño de Algoritmos (AyDA) [11] que brinda a los estudiantes conceptos fundamentales tales como Programación Recursiva, Tipos de

Datos Abstractos (TDA), Complejidad Computacional, y Técnicas de Diseño de Algoritmos: Backtracking, Divide y Conquista, Programación Dinámica, Método Greedy [4]. Al finalizar este curso, los estudiantes deben realizar un proyecto integrador, aplicando los principales temas aprendidos. Se consideró entonces beneficioso realizar un proyecto que desarrollara una herramienta didáctica para la materia de la carrera que introduce los conceptos fundamentales de LP y LdPPO.

Teniendo en cuenta lo anteriormente mencionado, se propuso definir un proyecto de desarrollo de herramientas educativas para asistir en el proceso de enseñanza/aprendizaje de los temas abordados en el curso de Lógica. Cada una de estas herramientas fue desarrollada por un equipo de 2 ó 3 estudiantes del 2º año de la carrera, como trabajo final de los cursos de AyDA y de Lógica. En este marco, se desarrollaron varias herramientas didácticas, visuales e interactivas que se ajustan a los contenidos, la notación y la metodología seguidos en el curso de Lógica, y que son fáciles e intuitivas en su uso [3, 5, 6, 9, 10, 17, 18]. Estas herramientas son utilizadas como soporte en el proceso de enseñanza/aprendizaje, tanto por docentes como por estudiantes del curso de Lógica.

En este trabajo se describen brevemente algunas de las herramientas desarrolladas. Cada herramienta permite experimentar con algunos de los contenidos del curso de Lógica, y a la vez para algunos contenidos se ha desarrollado más de una herramienta con el objetivo de abordarlos con un “enfoque” complementario, y en algunos casos, diferente.

Este artículo está estructurado de la siguiente manera. En la Sección 2 se describen los contenidos de las materias en cuestión: Análisis y Diseño de Algoritmos y Ciencias de la Computación II. En la Sección 3 se presentan las características principales de las herramientas desarrolladas y se describe brevemente, por razones de espacio, la funcionalidad de cada una. La Sección 4 describe algunas experiencias en el uso de las herramientas. Finalmente, se presentan las conclusiones sobre el trabajo realizado en la Sección 5.

2 Materias Involucradas en el Desarrollo de las Herramientas

En esta sección, se describen brevemente los contenidos de las dos materias involucradas en el desarrollo de las herramientas. Ambas materias se dictan en el primer semestre del 2º año de la carrera.

2.1 Ciencias de la Computación II

Es un curso introductorio a la LP y a la LdPPO que se dicta en el primer semestre del segundo año de la carrera. Los contenidos del curso están orientados a brindar a los estudiantes fundamentos de lógica para computación. Para esto, se introducen la LP y la LdPPO, abarcando por un lado los conceptos más clásicos de la lógica (sintaxis, semántica, métodos de deducción y demostración, resultados de corrección y completitud, satisfacibilidad y validez, formalización de especificaciones en lenguaje natural), y por otro, los fundamentos para algunas aplicaciones de la lógica en computación: verificación de software, programación lógica, bases de datos, etc.

Las clases se dictan en la modalidad teórico-práctica. Se presenta semanalmente un núcleo temático, utilizando filminas elaboradas por los docentes, acompañadas de ejercicios desarrollados en pizarrón con la participación de los estudiantes. Los estudiantes desarrollan los ejercicios propuestos, consultando a los auxiliares de la materia, o con la asistencia de las distintas herramientas didácticas que fueron desarrolladas para ser usadas en la materia por otros estudiantes de la carrera [12].

2.2 Análisis y Diseño de Algoritmos I

Entender un problema en términos computacionales e intentar escribir un programa que lo resuelva es el primer escollo que el estudiante debe confrontar al abordar problemas del mundo real. En general, ellos se encuentran con que la descripción de este tipo de problemas es algo acotada, y los primeros obstáculos surgen al intentar separar los detalles necesarios de los innecesarios para obtener una vista abstracta o modelo del problema.

El objetivo del curso de Análisis y Diseño de Algoritmos está centrado en la introducción de las bases para el desarrollo de software como lo es la especificación de TDAs, la integración de especificaciones e implementaciones, la construcción de librerías de componentes y la reusabilidad del software desarrollado. Para lograr este objetivo, es necesario abstraer el diseño de una manera apropiada, eficiente e implementable para resolver los problemas. Esta tarea sin buenas prácticas de desarrollo puede resultar engorrosa, es necesario que el programador acceda a los aspectos principales del conocimiento: las técnicas de diseño. Los buenos diseños incluyen técnicas básicas como la búsqueda Exhaustiva, Divide y Conquista, el Método Greedy, la Programación Dinámica y otras heurísticas junto con estructuras de datos adecuadas.

Para este curso se considera que implementar soluciones para problemas relacionados con el desarrollo de herramientas para el aprendizaje, es un buen tópico para la enseñanza de estas técnicas. Así, los estudiantes pueden aplicarlas en proyectos reales, académicos, didácticos y también atractivos generando desafíos para lograr buenas soluciones. Se ha trabajado durante los últimos años en este tipo de proyectos integradores significando para el alumno un desarrollo a gran escala, donde se aplica la metodología propuesta para resolver un problema basado en una situación concreta y real [11].

3 Las Herramientas Desarrolladas

Todas las herramientas que se describen en esta sección fueron desarrolladas por estudiantes de la carrera al finalizar 2° año como Proyecto Final de las materias que involucran contenidos de lógica y análisis y diseño de algoritmos. Son herramientas didácticas, visuales e interactivas que se ajustan a la notación, metodología y contenidos de Ciencias de la Computación II, lo que hace que sean muy fáciles e intuitivas de usar para los estudiantes. Las herramientas son software libre bajo

licencias GNU GPL[7], siendo posible su descarga para uso, estudio o modificación del código fuente.

Las herramientas fueron implementadas en el lenguaje de programación C++, siguiendo un diseño orientado a clases. Mediante la identificación de TDAs (Tipos Abstractos de Datos), se abstraieron características y comportamientos propios de los mismos que posteriormente se plasmaron en clases de C++. La implementación de cada herramienta tuvo en cuenta los aspectos necesarios para lograr la mayor eficiencia en cuanto a la complejidad computacional de los algoritmos desarrollados.

Para la interfaz gráfica de todas las herramientas se usó el framework Qt [16], que brinda un entorno completo y documentado que permite la implementación de interfaces gráficas intuitivas y amigables. Las interfaces fueron desarrolladas inicialmente en español, pero pueden ser traducidas al inglés fácilmente usando un conjunto de herramientas provistas por Qt. El desarrollo de estas interfaces resultó tan desafiante como productivo ya que no forma parte del contenido de las materias en cuestión.

Todas las herramientas cuentan con Ayuda on-line para el usuario y mediante mensajes de error brindan al usuario la información necesaria ante la presencia de un error.

En las siguientes subsecciones se describe brevemente la funcionalidad de cada una de las herramientas desarrolladas y, por razones de espacio, se muestran solamente algunas interfaces de las mismas.

3.1 SAT y DP Solver

La herramienta SAT (Problema de la satisfacibilidad booleana) fue desarrollada para determinar la satisfacibilidad de un conjunto de cláusulas de LP de manera eficiente y para mostrar la importancia de la LP, ya que la mayoría de los estudiantes en sus primeros años carece del conocimiento de la cantidad de problemas que pueden ser resueltos mediante el uso de la lógica. Esta herramienta permite al usuario experimentar con dos problemas reales, como son el coloreo de un grafo y el de las N reinas, verificando que pueden resolverse como un problema de satisfacibilidad booleana [8, 18]. En la Figura 1 se puede observar una de las ventanas para el problema de las N reinas.

DPSolver se pensó como una herramienta didáctica e interactiva que sirviera como apoyo a la hora de afianzar conceptos de LP, particularmente la aplicación del algoritmo de Davis Putnam a partir de un conjunto de cláusulas para determinar su satisfacibilidad. DPSolver también permite calcular la forma normal conjuntiva para cualquier fórmula de LP dada, y la posibilidad de trabajar con consecuencias semánticas [3].

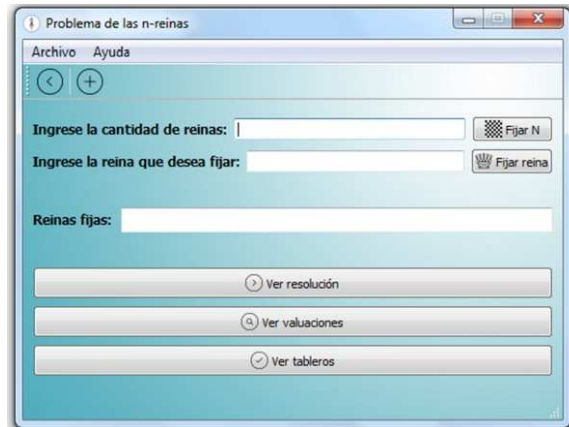


Figura 1. Satisfacibilidad booleana aplicado al problema de las N-reinas

3.2 YAT3

La herramienta YAT3 (Yet Another Truth Table Tool) permite trabajar con Árboles de Refutación tanto en LP como en LdPPO [17]. Dentro del marco práctico de un curso de lógica, es habitual que se requieran construcciones de estos árboles a partir de fórmulas lógicas para poder determinar consecuencias lógicamente válidas y validez de fórmulas. De esta manera, la construcción de un árbol se desarrolla aplicando una serie de reglas hasta determinar si cada rama del mismo se puede cerrar o no. La herramienta considera tres modos distintos de construir un árbol de refutación. El primero consiste en ofrecer, a partir de una fórmula ingresada por el usuario, el Árbol de Refutación construido. El mismo se va mostrando en pantalla automáticamente hasta completarse. El segundo modo, consiste en ofrecer al usuario la posibilidad de generar el árbol paso por paso, una regla a la vez. De esta manera, se puede ir verificando el desarrollo del mismo y proceder a un análisis más lento de los pasos (reglas) que se ejecutan. El tercer modo, y de más interacción con el usuario, consiste en que él mismo decida la forma en que se genera el árbol, logrando así algo más cercano a lo que puede desarrollar con lápiz y papel.

3.3 FOLST y LogicChess

FOLST [10, 14] y LogicChess [9] son dos herramientas que permiten la evaluación sintáctica y semántica de fórmulas de LdPPO en modelos construidos por el usuario en el dominio provisto por cada herramienta.

FOLST permite definir modelos sobre dos dominios: Mundo y Granja. La Granja (Figura 2) consiste en una imagen de una granja en la que distintos animales (por ejemplo, vacas y gallinas) se pueden agregar en distintos lugares (el corral, el bosque, etc.). Cada animal tiene atributos (su especie, ubicación y si está durmiendo) y relaciones que permiten formalizar información real en este contexto. Se proveen once

predicados unarios, dos predicados binarios y una función. El Mundo provee un mapa dividido en continentes sobre el cual se pueden agregar ciudades (capitales o no) que se pueden conectar. Se definieron seis predicados unarios, cinco predicados binarios y una función.

LogicChess permite trabajar sobre un conjunto finito de modelos. Cada modelo representa un tablero de Ajedrez y está formado por Fichas, Números y Vacío (referencia a un casillero que no contiene fichas). El usuario puede crear y manejar de forma fácil estos modelos quitando, agregando y modificando cada componente. En LogicChess se definieron un total de quince relaciones clasificadas en: relaciones de identificación, relaciones de posición relaciones de distancia.



Figura 2. Mundo Granja definido por FOLST

3.4 Clausula y Clprover

Clausula permite a estudiantes y docentes experimentar con conjuntos arbitrarios de cláusulas de la LdPPO con el objetivo de determinar su (in)satisfacibilidad [6,13]. Para esto, Clausula implementa métodos clásicos y fundamentales de la LdPPO como el método de Resolución de cláusulas. Además, la herramienta brinda la posibilidad de calcular el unificador más general (umg) de un par de literales, usando el Algoritmo de Unificación de Robinson, y aplicar estrategias de simplificación al conjunto inicial de cláusulas, tales como eliminación de cláusulas tautológicas, eliminación de literales por idempotencia y simplificación de cláusulas por literales puros. Con respecto a la determinación de la satisfacibilidad de un conjunto de cláusulas, la herramienta detecta si se trata de una Lógica de Programas, un conjunto de cláusulas de Horn o un conjunto arbitrario para aplicar la estrategia pertinente en cada caso.

Clprover es la otra herramienta desarrollada para determinar (in)satisfacibilidad de cláusulas de la LdPPO, agregando la posibilidad de trabajar también a partir de fórmulas arbitrarias. Además de brindar la funcionalidad provista por Clausula, Clprover permite simplificación por subsunción de cláusulas y factorización de cláusulas [15]. En la Figura 3 se muestra una captura de pantalla de la herramienta.

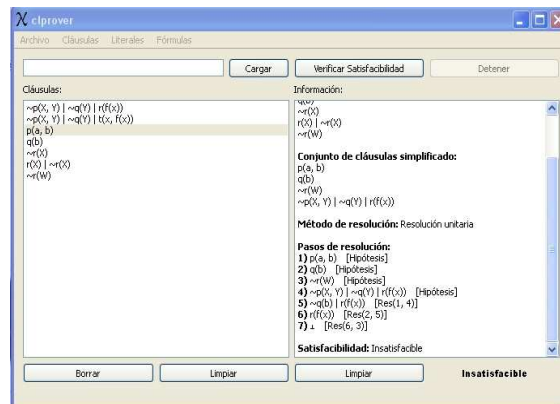


Figura 3. Método de Resolución en Clprover

3.5 TheM

La herramienta THEM (Teaching Herbrand Models) permite determinar la satisfacibilidad de un conjunto de cláusulas de la LdPPO a través de la existencia de Modelos de Herbrand [5]. Esto resulta importante ya que cualquier fórmula de esta lógica puede llevarse a forma clausular, y luego aplicar la herramienta para conocer su validez.

Con la intención de que la herramienta sea didáctica y de utilidad para los estudiantes, se implementó un algoritmo que muestra lo más claro posible el procedimiento que se realiza de forma manual para resolver este tipo de problemas. En primera instancia, se genera el conjunto S , es decir, el conjunto de cláusulas definidas y luego se definen el Universo y la Base de Herbrand. Luego de expresar la suposición de que existe un Modelo de Herbrand que satisface cada una de las cláusulas, se busca probar la existencia de éste. Para hacerlo se realiza la selección cláusula por cláusula de literales ya instanciados, evitando contradicción con aquellos que pertenezcan a la solución parcial. En caso de insatisfacibilidad, se realiza un procedimiento similar, utilizando los números de pasos para mostrar la contradicción que lleva a tal resultado. La Figura 4 muestra los posibles modelos para el conjunto de cláusulas ingresado.

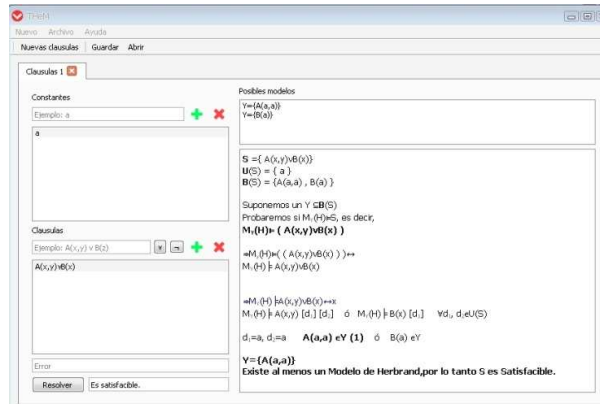


Figura 4. Modelos de Herbrand con THEM

4 Experiencia en el Uso y Desarrollo de las Herramientas

Las herramientas presentadas en la sección anterior se han utilizado en la materia Ciencias de la Computación II como software de apoyo, tanto por los docentes para introducir y ejemplificar algún tema durante las clases, como por los estudiantes para realizar la ejercitación práctica correspondiente a la materia. Los estudiantes han reportado experiencias positivas en el uso de las distintas herramientas (la materia tiene entre 100 y 120 estudiantes por año), manifestando que son fáciles de usar, y que les permiten aplicar los conceptos estudiados en la materia y comprobar sus resultados, participando así más activamente en su proceso de aprendizaje, y autoevaluando sus conocimientos con anterioridad a las instancias de evaluación.

En la actualidad, se dispone de al menos una herramienta para asistir en el proceso de enseñanza/aprendizaje de todos los contenidos de la materia de lógica: DPSolver y SAT para LP y sus aplicaciones; YAT3 para árboles de refutación en LP y LdPPO; FOLST y LogicChess para semántica en LdPPO; THEM para Modelos de Herbrand; Clausula y Clprover para experimentar con resolución en LdPPO. YAT3, THEM, Clausula y Clprover permiten además abordar, indirectamente, los conceptos de consecuencia semántica y validez de fórmulas en LdPPO.

En cuanto al desarrollo de las herramientas involucra la puesta en marcha de no sólo los programas diseñados con las técnicas adecuadas en cada caso e implementados en lenguaje C++, sino también la vinculación de este desarrollo con el entorno gráfico. El diseño independiente de la solución y la interfaz gráfica favorece al mantenimiento del software desarrollado y también al reuso del mismo. En muchos casos, se han abordado nuevos desarrollos, por ejemplo extensiones en cuanto a la funcionalidad de alguna herramienta, basados en implementaciones anteriores, resultando un trabajo ordenado y sistemático para el estudiante.

5 Conclusiones

En este trabajo se describió un proyecto de desarrollo de herramientas didácticas como trabajo final de dos materias del 2º año de la carrera.

El objetivo de estas herramientas es asistir a los estudiantes de la materia Ciencias de la Computación II en el proceso de enseñanza/aprendizaje de los distintos temas abordados en la misma. Las herramientas usan la misma notación y metodología seguidas en la materia, y son utilizadas como software de apoyo para los estudiantes al realizar la ejercitación práctica correspondiente a cada tema, y por los docentes en las clases para introducir y ejemplificar temas en los que cada herramienta brinda asistencia. Desde el punto de vista pedagógico, faltaría una evaluación más sistemática y formal del aporte de cada una de estas herramientas en el proceso de enseñanza/aprendizaje de los contenidos que cada herramienta aborda.

Para los estudiantes que desarrollaron estas herramientas, el aporte involucró dos aspectos: por un lado, haber tenido una primera experiencia de un desarrollo de software completo, desde la etapa de requisitos hasta la implementación, aprendiendo nuevas tecnologías y aplicando temas aprendidos en la materia Análisis y Diseño de Algoritmos, y por otro lado, la integración y profundización de los temas de lógica involucrados en la funcionalidad de la herramienta. Otro de los aportes para estos estudiantes, fue que tuvieron la posibilidad de presentarla herramienta desarrollada durante una clase de la materia Ciencias de la Computación II a los estudiantes de la misma. De esta forma, los desarrolladores tuvieron una primera experiencia de presentación oral a un grupo numeroso, permitiendo un intercambio con los estudiantes del curso, quienes se sintieron motivados al ver que estudiantes sólo un poco más avanzados que ellos fueron capaces de desarrollar un software completo con la formación adquirida hasta ese momento de la carrera. Además, los docentes de ambas materias alentaron a los estudiantes a presentar las herramientas desarrolladas en concursos de trabajos estudiantiles nacionales y en congresos nacionales e internacionales (en la descripción de cada herramienta se halla la referencia correspondiente). Esto permitió a los estudiantes tener una primera experiencia en la preparación de un artículo para un congreso así como también en la presentación del mismo.

Las herramientas tienen otra ventaja para la enseñanza: son software libre. Cuando se utiliza software libre en los procesos de enseñanza/aprendizaje, se da la posibilidad a los estudiantes de utilizar y compartir los recursos que éste les ofrece. Pero por sobre todo, permite potenciar las capacidades didácticas de los programas de computadora, permitiendo a los estudiantes explorar el software y analizar las implementaciones reales de los algoritmos propuestos. De esta forma, además de utilizarlas como un instrumento de aprendizaje, el estudiante puede ser partícipe en la construcción de sus propias herramientas.

Se considera que el desarrollo de un proyecto transversal a dos materias como el que se presentó en este artículo, permite que los estudiantes integren y profundicen en los contenidos de ambas, y no visualicen a las mismas como cursos aislados e independientes, sino que forman parte de un proceso que de manera incremental los irá formando en su carrera. Es por esta razón, que ambas cátedras van a seguir apostando a este tipo de propuestas integradoras.

Referencias

1. Arenas Alegría, L.: *Lógica Formal para Informáticos*. Ediciones Diaz de Santos (1996).
2. Ben-Ari, M.: *Mathematical Logic for Computer Science*. Springer-Verlag London (2012).
3. Cicconi, D., Fernández Cocirio, M. DPSolver: Una herramienta interactiva para la implementación del algoritmo de Davis Putnam. Simposio de Trabajos Estudiantiles, 2º CoNaISI, Argentina, (2014).
4. Cormen, T.; Lieserson, C.; Rivest, R. *Introduction to Algorithms*. Ed. The MIT Press. 2009.
5. Dahl, J., Fujii, D. THeM: Una Herramienta Didáctica para Modelos de Herbrand. Aceptado para ser presentado en Concurso de Trabajos Estudiantiles, 45ºJAIIO, Argentina(2016).
6. Ferrante, E. Clausula: Herramienta Didáctica para la Enseñanza de Lógica de Predicados de Primer Orden. Concurso de Trabajos Estudiantiles, 38ºJAIIO, Argentina(2009).
7. GNU General Public License (GPL). <http://www.gnu.org/licenses/gpl.html>
8. Kelley, J. *The Essence of Logic*, Prentice Hall (1997).
9. Kiehr, A., Re Medina, M. LogicChess: Herramienta Didáctica para la Ejercitación en Lógica de Predicados de Primer Orden. Concurso de Trabajos Estudiantiles, 41ºJAIIO, Argentina(2012).
10. Maggiori, E., Gervasoni, L. FOLST: Una Herramienta Didáctica para la Lógica de Predicados de Primer Orden. Primer Premio en el Concurso de Trabajos Estudiantiles, 41ºJAIIO, Argentina(2012).
11. Materia Análisis y Diseño de Algoritmos I. <http://aydalgor.alumnos.exa.unicen.edu.ar>
12. Materia Ciencias de la Computación II. <http://ccomp2.alumnos.exa.unicen.edu.ar>
13. Mauco, M.V., Ferrante E.: Clausula: A Didactic Tool to Teach First Order Logic. In: Proceedings de ISECON 2009, Information Systems Education Conference, v26: §4142. WashingtonDC. USA. (2009).
14. Mauco, M.V., Maggiori, E., Gervasoni, L., Ferrante, E., Felice, L. FOLST: A Didactic Tool to Support First Order Logic Semantics Learning. In Proceedings of International Conference on Future Computers in Education. Shangai. China. Lecture Notes in Information Technology, Vols.23-24, 302-307 (2012).
15. Mauco, M.V., Moauro, L., Felice, L. Una Herramienta Didáctica para la Enseñanza de Lógica de Predicados de Primer Orden. En Anales del Congreso Iberoamericano de Educación Superior en Computación (CIESC 2010), Paraguay (2010).
16. Qt: Documentación oficial sobre el entorno gráfico. <http://doc.qt.io>
17. Ruau, K., Tosini, J.M. Yet.anotherTruth.Tree.Tool: Una herramienta didáctica sobre Árboles de Refutación. Concurso de Trabajos Estudiantiles, 44ºJAIIO, 63-71, Argentina(2015).
18. Santillán Cooper, M., Horquin, E., Covelli, T. SAT: Herramienta para la resolución del Algoritmo de Davis Putnam LogemannLoveland y problemas computacionales que pueden ser resueltos a partir de la lógica proposicional. Reporte Interno UNCPBA (2015).

Estilos de Aprendizaje y Rendimiento Académico de los Alumnos Ingresantes a la Lic. en Sistemas de la UNRN

Edith Lovos¹, Tatiana Gibelli¹,

¹ CIEDIS, Sede Atlántica Universidad Nacional de Río Negro,
8500 Viedma Río Negro, Argentina
{elovos,tgibelli}@unrn.edu.ar

Abstract. El trabajo busca identificar los estilos de aprendizajes que predominan en los alumnos que ingresan a la Licenciatura en Sistemas de la Universidad Nacional de Río Negro, siguiendo el modelo propuesto por Felder y Silverman. Asimismo, se analiza la relación entre el rendimiento académico de los mismos en dos asignaturas del primer año, con la intención de diseñar estrategias docentes que posibiliten acompañar a los alumnos a lo largo de la cursada. Se presentan los resultados alcanzados hasta el momento y líneas de acción a futuro

Keywords: ingresantes, estilos de aprendizaje, informática.

1 Introducción

Se han publicado diversos artículos en relación a los problemas que se presentan durante el aprendizaje, en los cursos introductorios a las carreras vinculadas a la ciencia informática. Estos resaltan la importancia de estos cursos en los trayectos curriculares de las diferentes carreras y la falta de habilidades para la resolución de problemas que se observa en los estudiantes que ingresan a las mismas [1,2].

Cada individuo tiene su propia forma de adquirir, procesar y utilizar la información, determinando un estilo de aprendizaje. En el ámbito académico, éste se refleja en las diferentes habilidades, intereses, debilidades y fortalezas académicas del estudiante [3]. En este sentido, y cómo docentes de primer año de la Licenciatura en Sistemas de la Universidad Nacional de Río Negro (UNRN), nos hemos propuesto analizar los estilos de aprendizaje de los alumnos ingresantes. La intención es conocer cuales son los estilos que predominan en los mismos y, de esta forma, analizar si existe una relación entre el rendimiento académico de los alumnos en los distintos cursos con sus formas de aprendizaje. Entendemos que es a partir de este conocimiento que se deben diseñar estrategias de enseñanza y aprendizaje que posibiliten acompañar a los alumnos a lo largo de la cursada.

El trabajo que se desarrolla a continuación se enmarca en el proyecto de investigación “La mediación de las tecnologías de la información y la comunicación en procesos educativos. Innovaciones para mejorar los procesos de enseñanza y aprendizaje” (40-C-486), acreditado por la UNRN. A través del mismo, lleva adelante una investigación que permitirá diseñar metodologías y estrategias innovadoras que

favorezcan los procesos educativos mediados por TIC, con la intención de potenciar los procesos de enseñanza y aprendizaje (analizando posibilidades y barreras encontradas), y el desarrollo de habilidades (pensamiento crítico, trabajo en equipo, etc) necesarias para el desarrollo en la sociedad del conocimiento.

2 Estilos de Aprendizaje

El aprendizaje en un contexto estructurado puede pensarse como un proceso de dos fases: recepción de la información a través de los sentidos y/o adquirida en forma introspectiva y procesamiento de la misma: como la memorización, el razonamiento inductivo o deductivo, la reflexión o acción, la introspección o interacción con otro [4]. Y este proceso, cada individuo lo lleva adelante de diversas maneras, es decir cada persona tiene su propio estilo de aprendizaje.

Un modelo de estilos de aprendizaje, posibilita clasificar a los estudiantes de acuerdo a las formas en las que reciben y procesan la información. Si bien existen diferentes modelos teóricos para caracterizar los estilos de aprendizaje, para esta investigación hemos optado por el modelo de Felder-Silverman [4], el cual fue diseñado para su aplicación con estudiantes de ingeniería, pero luego fue utilizado en otras disciplinas y que ha sido probado en diferentes experiencias [3,5,6].

El modelo de estilos de aprendizaje propuesto por Felder and Silverman [4], contiene 44 preguntas cuyas respuestas solo admiten dos opciones (a o b) y es tipo escala. Las preguntas permiten conformar 5 dimensiones de análisis: sensorial – intuitivo (percepción), visual – auditivo (entrada), activo – reflexivo (procesamiento), secuencial - global (comprensión) y deductivo-inductivo (organización). Más adelante Richard Felder [7] realizó cambios al modelo, eliminando la dimensión inductiva - deductiva y cambiando la dimensión visual - auditiva a visual - verbal con la intención de incluir tanto palabras escritas como habladas. De acuerdo a este modelo, los estudiantes se clasifican en los estilos de aprendizaje que se presentan en la tabla 1.

A su vez, el modelo permite calificar a cada estilo en tres categorías: Fuertes, Moderados y Equilibrados, de acuerdo a la preferencia de los estudiantes por uno u otro de los polos de la dimensión analizada (visual o verbal, activo o reflexivo, etc). El modelo sugiere tendencias en el comportamiento y no predicciones de comportamiento. En este sentido, su uso provee a los docentes de un instrumento con el cual determinar los diferentes estilos de aprendizajes que se encuentran en un curso y en función de ellos realizar un el diseño instruccional que se ajuste a las necesidades de sus alumnos. Por otra parte, desde el punto de vista de los alumnos, conocer su estilo de aprendizaje, les permite entender su modo de apropiación del conocimiento, identificado sus debilidades y fortalezas para luego trabajar en las habilidades asociadas con sus estilos menos preferidos [8].

Tabla 1. Dimensiones y estilos de aprendizaje del modelo Felder y Silverman

Dimensión	Estilos	Características
Procesamiento	Activos	se sienten más a gusto con la experimentación activa, es decir necesitan explicar, aplicar o discutir la

		información.
	Reflexivos	necesitan examinar y manipular la información en forma introspectiva. Tienen una tendencia al trabajo en soledad.
Comprensión	Secuenciales	al resolver problemas siguen un proceso de razonamiento lineal. De esta forma aprenden mejor cuando el material se presenta en forma progresiva constante de complejidad y dificultad
	Globales	son mejores con el pensamiento divergente y la síntesis. La escuela es en general una experiencia difícil para este tipo de estudiantes.
Entrada	Visuales	retienen mejor lo que pueden ver (diagramas, cuadros, demostraciones, líneas de tiempo).
	Verbales	recuerdan mejor lo que escuchan o aprenden de una discusión. Aprenden mejor cuando pueden explicar a otros.
Organización	Sensorial	prefieren los hechos, datos y la experimentación. Son cuidadosos pero más lentos.
	Intuitivos	prefieren los principios y teorías. Gustan de la innovación y no de la repetición.

3 Metodología

El proceso de indagación propuesto en este artículo, estuvo dirigido a los alumnos que cursan el primer año de la Licenciatura en Sistemas de la Sede Atlántica de la UNRN, carrera que tiene el trayecto curricular que se muestra en la tabla 2.

Tabla 2. Trayecto curricular de los alumnos ingresantes a la Lic. Sistemas

Cuatrimestre	Asignatura	Correlativas
Primero	Organización de Computadoras (OCI)	-
	Programación de Computadoras I (PCI)	-
	Razonamiento y Resolución de Problemas (RRP)	-
	Introducción a la Lectoescritura Académica (ILEA)	-
Segundo	Arquitectura	OCI
	Programación de Computadoras II (PCII)	PCI
	Matemática I	RRP

Tanto RRP como ILEA corresponden al trayecto de ingreso que ha fijado la UNRN para todas sus carreras. RRP, asignatura vinculada al área de matemáticas, es transversal y en algunas carreras es correlativa a algunas materias del primer año. En

este trabajo se tuvieron en cuenta a los datos sobre el rendimiento de los alumnos en el cursado de las materias PCI y OCI que son las asignaturas específicas de la carrera.

En relación a las formas de enseñanza, tanto en PCI como en OCI, las clases presentan conceptos, ejemplos y situaciones problemáticas a través de las cuales los alumnos internalizan los conceptos. En el caso particular de PCI, se realizan además actividades prácticas de laboratorio, que buscan promover el trabajo en equipos y en forma colaborativa haciendo uso de recursos TIC (aulas virtuales, laboratorios de programación virtuales, entre otros) como mediadores del proceso.

Para conocer los estilos de aprendizaje de los alumnos se optó por el cuestionario de Felder y Silverman, y se adicionó una encuesta que permitía conocer para cada alumno aspectos relacionados con su edad, género, situación académica al momento de ingresar, procedencia, etc. Para llevar adelante el seguimiento de los alumnos, el cuestionario no es totalmente anónimo. El mismo se aplicó al inicio del primer cuatrimestre, en formato papel durante una clase práctica de la asignatura PCI. El tiempo para responder al mismo fue de 30 minutos.

Con los datos recogidos se realizó un análisis cuantitativo. Para la mayoría de las variables se aplicó un análisis estadístico de tipo descriptivo (cálculo de frecuencias absolutas y relativas). Así mismo, para el caso de los estilos de aprendizaje se aplicó un análisis de correlación lineal (con cálculo del coeficiente de r de Pearson y su respectivo valor de probabilidad p).

4 Resultados

A continuación se muestran algunos de los resultados obtenidos a partir del análisis de los datos recolectados.

4.1 Caracterización de la muestra

Se obtuvieron en forma completa para el análisis, un total de 60 encuestas sobre datos personales respondidos por estudiantes que cursaron la materia PCI durante el primer cuatrimestre de 2016. De esta muestra se obtuvieron los siguientes datos:

- respecto al género, mayormente (93%) son varones.
- la edad promedio es de 21 años
- el 46% terminó de cursar el último año de nivel medio, durante el año 2015.
- un 15% de los ingresantes adeudan materias del nivel medio y solo el 13% proviene de escuelas técnicas.
- para el 55%, la carrera es su primera experiencia universitaria.
- el 70% de los estudiantes, manifestó no tener experiencias previas en programación.
- Solo el 29% de los ingresantes trabajan.

4.2 Estilos de aprendizaje

Se utilizaron como datos un total de 52 cuestionarios sobre estilos de aprendizaje (según se describió en el apartado anterior) respondidos en forma completa, descartando del análisis algunos cuestionarios recibidos en forma incompleta. Se realizó un análisis de frecuencia de cada tipo de aprendizaje distinguiendo además en cada uno la intensidad leve, moderada o alta (en tabla 3 frecuencias relativas y en gráfico 1 frecuencias absolutas). A partir de este análisis puede observarse que predominan los tipos de aprendizaje: activo, sensorial, visual y secuencial, con tendencias moderada y leve principalmente.

Tabla 3. Frecuencias relativas de los estilos de aprendizaje observados

Estilo de aprendizaje		Alto	Moderado	Leve	Total
Procesamiento	Activo	0,08	0,47	0,45	0,73
	Reflexivo	0,00	0,21	0,79	0,27
Percepción	Intuitivo	0,00	0,20	0,80	0,29
	Sensorial	0,11	0,46	0,43	0,71
Entrada	Verbal	0,00	0,00	1,00	0,10
	Visual	0,11	0,51	0,38	0,90
Comprensión	Global	0,06	0,35	0,59	0,33
	Secuencial	0,03	0,29	0,69	0,67

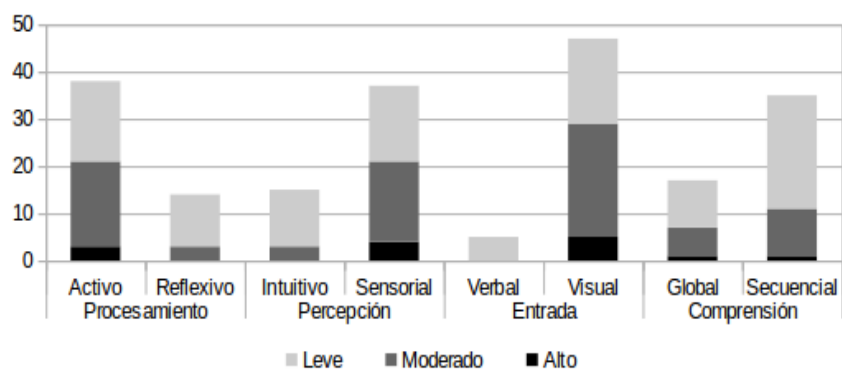


Fig. 1. Frecuencias absolutas de los estilos de aprendizaje observados.

Se realizó además un análisis de correlación (en tabla 4 puede verse el coeficiente de correlación de Pearson versus el valor de probabilidad asociado). Se observa que tienen correlación positivamente y significativa ($p < 0.01$) los ejes: procesamiento (activo-reflexivo) con comprensión (secuencial-global) por un lado ($r = 0,37$) y percepción (sensorial-intuitivo) con entrada (visual-verbal) por el otro ($r = 0,38$). La relación entre entrada (visual-verbal) y comprensión (secuencial-global) presenta una tendencia negativa ($r = -0,24$) aunque no llega a ser significativa ($p = 0,09$). Es decir, un aprendizaje más activo que reflexivo estaría relacionado con un aprendizaje

principalmente secuencial (por sobre el global); mientras que un aprendizaje más sensorial que intuitivo se asociaría con un aprendizaje prioritariamente visual (por sobre verbal). Además un aprendizaje donde predomina lo visual sobre lo verbal se asociaría a un aprendizaje más global que secuencial.

Tabla 4. Análisis de correlación.

Coef./Prob.	Dif. Act-Ref	Dif. Sen-Int	Dif. Vis-Ver	Dif. Sec-Glob
Dif. Act-Ref	1	0,592	0,543	0,007
Dif. Sen-Int	0,076	1	0,005	0,973
Dif. Vis-Ver	0,086	0,381	1	0,091
Dif. Sec-Glob	0,372	-0,005	-0,237	1

4.3 Relación con rendimiento académico

Los resultados de rendimiento académico del total de los alumnos inscriptos a las materias del primer año (PCI y OCI) pueden verse en la tabla 5.

Tabla 5. Resultados académicos de los alumnos ingresantes

	Aprobaron	Desaprobaron	Abandonaron
OCI	0,27	0,33	0,41
PCI	0,2	0,24	0,56

Se observa que la tasa de abandono en ambas materias supera el 40% de los inscriptos, con un desgramiento mayor en OCI. Esta diferencia puede estar asociada al esquema de exámenes que se aplican en ambas materias. En OCI se aplican en dos partes: a mitad y a fin del cuatrimestre con una instancia de recuperación para cada uno. En cambio en PCI, la acreditación se realiza a través de un examen parcial con dos instancias de recuperación, que se aplican sobre el final del cuatrimestre. Sumado a esto, durante el cuatrimestre se realizan actividades de evaluación (trabajos grupales) cuya acreditación se obtiene con el desarrollo y entrega de los mismos. En cualquiera de los casos, el nivel de desgramiento, afectará el avance en la carrera. (ver Tabla 2)

Respecto a los estilos de aprendizaje de los alumnos que cursaron estas materias, se han obtenido las frecuencias relativas que se indican en la tabla 6. Sobre la forma de procesamiento, es posible indicar que son mayormente activos. En el caso particular de la asignatura PCI, se observa que los alumnos que no han logrado aprobar la misma, tienen un estilo reflexivo, que requiere de situaciones de aprendizaje en forma más individual que grupal. En este sentido, la propuesta de trabajo en equipo usada en la asignatura, les ha requerido de un esfuerzo adicional para ajustar su estilo de aprendizaje a la misma.

En relación a la comprensión, se observa que los alumnos que no tuvieron un rendimiento académico necesario para aprobar ambas asignaturas, existe un equilibrio entre ambos polos de la escala (secuencial - global). En tanto, en el grupo en general, se observa un predominio del estilo de aprendizaje secuencial, que coincide con la

metodología docente que se utiliza para la enseñanza de los contenidos de estas materias, los cuales tienen una marcada estructura secuencial.

Tabla 6. Frecuencias relativas de los estilos de aprendizaje y los resultados académicos en OCI y PCI

Estilos de aprendizaje		Resultado OCI			Resultado PCI		
		Aprob.	Desap.	Libres	Aprob.	Desap.	Libres
Procesamiento	Activo	0,59	0,67	0,67	0,62	0,33	0,75
	Reflexivo	0,41	0,33	0,33	0,38	0,67	0,25
Percepción	Intuitivo	0,29	0,17	0,00	0,23	0,33	0,25
	Sensorial	0,71	0,83	1,00	0,77	0,67	0,75
Entrada	Verbal	0,06	0,00	0,00	0,08	0,00	0,00
	Visual	0,94	1,00	1,00	0,92	1,00	1,00
Comprensión	Global	0,35	0,50	0,33	0,38	0,50	0,38
	Secuencial	0,65	0,50	0,67	0,62	0,50	0,63

5 Conclusiones

La investigación que se presenta en este trabajo, resulta innovadora en el contexto de la UNRN y sus resultados posibilitan pensar en propuestas didácticas para las asignaturas de primer año, que en palabras de Felder [7] *“logren acercarse más y mejor a los estilos de aprendizaje de cada estudiante”*

En este sentido, del análisis de los resultados, se puede concluir respecto a los estilos de aprendizaje de los alumnos ingresantes a la Lic. en Sistemas de la UNRN, que existe un predominio de los estilos: activo, sensorial, visual y secuencial. Estos resultados coinciden con lo observado por Tocci [5] en una asignatura de primer año de la facultad de ingeniería de la Universidad Nacional de La Plata y por Cataldi específicamente en estudiantes de la especialidad informática de la facultad de la Universidad de Buenos Aires [3]

Así mismo, se observa en los alumnos de primer año de las asignaturas analizadas, que la preferencia de lo visual por sobre lo verbal, no afecta el rendimiento académico. Esto permite continuar avanzando en la inclusión de recursos TIC como simuladores y animaciones que complementen el proceso de enseñanza y aprendizaje de las asignaturas introductorias a la carrera.

Por otra parte, resulta importante reforzar aquellos estilos de aprendizaje vinculados a un mejor rendimiento académico en las asignaturas introductorias a la Licenciatura en Sistemas de la UNRN, sin que ello implique descuidar los demás. En este sentido, si tenemos en cuenta, que la tasa de inscriptos a la carrera es muy baja, comparada con otras instituciones de nivel nacional que ofrecen la misma titulación, conocer los estilos de aprendizajes de los alumnos, puede ayudar a generar estrategias de enseñanza que permitan mantener el interés de los alumnos.

Felder y Silverman [7] recomiendan el uso de técnicas de enseñanza que permitan estimular los distintos estilos, entre ellas:

- balancear el contenido de los materiales enfatizando en métodos de resolución de problemas (estilos sensorial/activo) con aquellos que acentúan la comprensión (estilos intuitivo/reflexivo)
- usar materiales visuales antes, durante y después de la presentación de los contenidos en forma verbal (estilos sensorial/visual).
- diseñar actividades que promuevan la colaboración entre los estudiantes (estilo activos)
- animar las soluciones creativas, aún cuando no sean correctas (estilos intuitivo/global)
- que los alumnos conozcan sus estilos de aprendizaje, les dará una idea de como aprenden en forma más eficiente y los ayudará a rediseñar sus estrategias de aprendizaje, más allá de cualquier estilo.

Como trabajo a futuro, se propone considerar los estilos de aprendizaje de los alumnos para la conformación de los grupos de trabajo de las actividades de laboratorio, en particular en el caso de las asignaturas de programación correlativas a PCI, teniendo en cuenta que existen investigaciones [9] que utilizan reglas de asociación para determinar relaciones entre los estilos de aprendizaje y los roles de los alumnos dentro de un grupo con el desempeño de los grupos de trabajo.

Referencias

1. Lorenti, L., Villagarcía Wanza, H. A., & Giacomantone, J. (2012). HARN: herramienta para el aprendizaje de sistemas de representación numérica. In *XVIII Congreso Argentino de Ciencias de la Computación*.
2. De Giusti, A. E., Madoz, M. C., Gorga, G., Feierherd, G. E., & Depetris, B. O. (2003). Enfoques y herramientas en la enseñanza de un primer curso de computación (CS1). En IX Congreso Argentino de Ciencias de la Computación
3. Figueroa, N., Cataldi, Z., Méndez, P., Rendon Zander, J., Costa, G., Salgueiro, F. A., & Lage, F. J. (2005). Los estilos de aprendizaje y el desgranamiento universitario en Carreras de informática. In *I Jornadas de Educación en Informática y TICs en Argentina*.
4. Felder, R. M., & Silverman, L. K. (1988). Learning and teaching styles in engineering education. *Engineering education*, 78(7), 674-681.
5. Tocci, A. M. (2015). Caracterización de estilos de aprendizaje en alumnos de ingeniería según el modelo de felder y silverman. *Fall Edition Vol. 8 No. 16*, 8(16), 101.
6. Costaguta, R., & Gola, M. (2009). Identificación de estilos de aprendizaje dominantes en estudiantes de informática. In *XV Congreso Argentino de Ciencias de la Computación*.
7. Felder, R. M. (1993). Reaching the second tier. *Journal of college science teaching*, 23(5), 286-290.
8. Felder, R. M., & Spurlin, J. (2005). Applications, reliability and validity of the index of learning styles. *International journal of engineering education*, 21(1), 103-112.
9. Nicoletti, M., Balmaceda, J. M., Schiaffino, S., & Godoy, D. (2013). Influencia de estilos de aprendizaje y roles de equipo en el aprendizaje colaborativo soportado por computadoras. In *Anais do Simpósio Brasileiro de Informática na Educação* (Vol. 24, No. 1, p. 557).

Aplicación del enfoque descendente para la enseñanza de Redes de Computadoras, un estudio comparativo

Nelson Rodríguez¹, Daniel Arias Figueroa², María Murazzo¹, Jorge Mercado¹,
Marcelo Moreno¹, Ernesto Sanchez², Alvaro Gamarra², Gustavo Gil²

¹ Universidad Nacional de San Juan, Departamento e Instituto de Informática, Complejo Universitario Islas Malvinas, Rivadavia. San Juan, Argentina

² Universidad Nacional de Salta, Centro de Investigación y Desarrollo en Informática Aplicada – C.I.D.I.A., Salta, Argentina

nelson@iinfo.unsj.edu.ar, daaf@cidia.unsa.edu.ar, maritemurazzo@gmail.com,
jorgenmp@gmail.com, mpmoren@gmail.com,
esanchez@cidia.unsa.edu.ar, gdgil@unsa.edu.ar

Abstract. El enfoque ascendente para la enseñanza de redes fue el que habitualmente se utilizó en las carreras tanto de informática como de ingeniería. Los libros más citados en la bibliografía siguen este enfoque. Algunos autores como Jim Kurose y Keith Ross cuestionaron esto y propusieron el sentido descendente, el cual comienza por las capas superiores para luego adentrarse en los detalles de las capas más bajas, tratando de aprovechar los conocimientos previos de los estudiantes en el uso de aplicaciones de red. En las carreras tradicionalmente se ha utilizado en enfoque ascendente y en particular en la UNSJ, UNSa y UCSAL actualmente se utiliza este nuevo enfoque. Por ello el presente trabajo presenta las experiencias, conclusiones, sugerencias, junto con un estudio comparativo de otras carreras fuera del país, con respecto al modelo de enseñanza de asignaturas sobre Redes de Computadoras.

Keywords: Teaching Computer networks, Computer Science Education, Computer Networks, Teaching methods

1 Introducción

Redes de Computadoras es un área muy compleja de la informática, que involucra muchas tecnologías, conceptos y protocolos entrelazados de una manera complicada de abordar. Sin embargo docentes e investigadores han aplicado diversas estrategias para lograr que los estudiantes puedan apropiarse de estos contenidos y puedan aplicarlos adecuadamente en las prácticas y en su futura vida profesional. Se puede citar el trabajo de Anders Berglund [1], que propone estrategias para la enseñanza de protocolos de Internet y en particular TCP. En el presente trabajo se aplica el enfoque descendente para la enseñanza de esta temática y además se analizan diversos cursos de universidades extranjeras de referencia, donde se aprecia en su mayoría esta estrategia.

El enfoque ascendente es el habitual heredado del origen de las redes de computadoras que son las telecomunicaciones, con una impronta ingenieril muy fuerte. Luego al transformarse estas redes de comunicaciones en redes de datos, la fusión generó una serie de especialistas nuevos y también una jerga que es muy precisa. Acrónimos y siglas por doquier que permiten referenciarse a un objeto sin tener que describir todas sus características como: RJ45, WLAN, IPv6, host, backbone, etc.

Sin embargo, los estudiantes de informática presentan ciertas dificultades para comprender el enfoque ascendente, además no tiene mucho sentido dedicar tiempo a clases de temas que en su quehacer profesional va a realizar esporádicamente.

Por otro lado, el enfoque descendente parte de la capa aplicación, que es la nuestro estudiante que recién comienza a cursar redes ha utilizado. De esta manera, se introduce a la materia por lo que conoce, las aplicaciones de red, para luego ir descendiendo en las capas que conforman el modelo y llegar a los detalles de cómo se implementan esas aplicaciones. En este punto, el estudiante comprende que para mejorar, proteger las aplicaciones, servicios y datos es necesario comprender lo que pasa debajo y de esta forma se lo introduce en las capas subsiguientes.

El enfoque ingenieril ascendente también tiene esta premisa, solo que el estudiante de telecomunicaciones ha tenido materias de física, teoría de la señal o similares y el concepto de señal le es familiar y no así el concepto de protocolo por ejemplo. Con lo cual se introduce al estudiante desde la capa física, pero en casos poco habituales (como materias optativas) llega a estudiar la capa transporte.

2 Enfoque tradicional

Habitualmente se enseña redes, siguiendo el modelo de capas de OSI, TCP o híbrido (como la mayoría de los libros de textos: por ejemplo Computer Networks de Andrew Tanenbaum y David Wetherall [3], Data and Computer Communications de Williams Stallings [4], Computer Networks: a Systems Approach de Larry Peterson y Bruce Davie [5], TCP/IP Illustrated de Richard Stevens [6], Computer Networks and Internets de Douglas Comer [7] y Data Communications, Computer Networks, and Open Systems de Fred Halsall ; para la parte práctica: Advanced Programming in the UNIX Environment de Richard Stevens y Stephen Rago), parten de una introducción general , luego se abordan detalles de la capa física, capa de enlace y así sucesivamente hasta llegar a la capa aplicación. A este enfoque se lo conoce como ascendente, dado que en la jerga de comunicaciones, la capa física se la referencia como capa 1, enlace 2 y así hasta llegar a la capa aplicación (7 en el modelo OSI). Algunas estrategias de enseñanza se basan en el modelo OSI de 7 capas, otras en el modelo TCP/IP de 3 capas, a las que usualmente se le agregan la capa enlace y capa física, dado que estas involucran el funcionamiento de las redes LANs, WLANs, y protocolos de acceso a la red como PPP.

Lo más ventajoso de este esquema es la variedad de libros de textos que siguen el mismo, y la madurez que ha alcanzado, debido al tiempo que se viene aplicando en la enseñanza de redes.

3 Ventajas del Enfoque descendente

La enseñanza de redes bajo este enfoque comienza por la capa aplicación y se trabaja hacia abajo terminando por la capa de enlace y los conceptos necesarios de la capa física para comprender el funcionamiento de los protocolos de redes LAN y WLAN. Según Kurose y Ross [8], la retroalimentación que recibieron de profesores y estudiantes, han confirmado que este enfoque de arriba hacia abajo tiene muchas ventajas y en realidad funciona muy bien pedagógicamente. En primer lugar, se hace hincapié en la capa de aplicación (un "área de alto crecimiento" en redes de computadoras). De hecho, muchos de los temas que están revolucionando las redes y el impacto de IT en el mundo de los negocios incluyen la Web, el intercambio de archivos peer-to-peer, y la transmisión de contenido multimedia, SaaS en Cloud Computing, forman parte de la capa aplicación, los cuales habitualmente son abordados en otras asignaturas.

Colocar el énfasis inicial en cuestiones de capa aplicación se diferencia de los enfoques adoptados en la mayoría de los textos tradicionales, que tienen sólo una pequeña cantidad de material en aplicaciones de red, sus requisitos, los paradigmas de capa de aplicación (por ejemplo, cliente-servidor y peer-to-peer), y las interfaz de programación de aplicaciones.

Por otra parte, según lo documentado por Kurose y Ross en el libro: Computer Networking: A Top-Down Approach [8], de acuerdo con su experiencia y la de otros instructores que aplican el enfoque descendente, comprobaron que la enseñanza de aplicaciones de red al comienzo del curso es una poderosa herramienta de motivación. Los estudiantes están encantados de aprender acerca de cómo trabajan las aplicaciones de red que utilizan cotidianamente, como el correo electrónico y la Web. Una vez que un estudiante entiende las aplicaciones, puede entonces comprender los servicios de red necesarios para soportar estas aplicaciones.

El estudiante puede entonces, a su vez, examinar las diversas formas en que dichos servicios pueden estar formulados y soportados por las capas inferiores. Cubriendo los conceptos y fundamentos detrás de las aplicaciones en una etapa temprana de la enseñanza de redes, se proporciona una motivación para el resto del curso.

Además, un enfoque de arriba hacia abajo permite a los profesores introducir el desarrollo de aplicaciones de red en una etapa temprana. Los estudiantes no sólo perciben cómo las aplicaciones y protocolos populares trabajan, sino también aprenden lo fácil que es crear sus propias aplicaciones de red y protocolos de nivel de aplicación.

Con el enfoque de arriba hacia abajo, los estudiantes obtienen la exposición temprana a las nociones de programación de sockets, modelos de servicio, y los conceptos de protocolos importantes que son necesarios en todas las capas subsecuentes. Proveyéndose ejemplos en algún lenguaje de programación de sockets como Java o Python, se destacan las ideas centrales sin confundir a los estudiantes con el código complejo. Los estudiantes universitarios de sistemas y ciencias de la computación no deberían tener dificultades para seguir y comprender este código.

Desde una perspectiva pedagógica, se observa que el enfoque descendente parece más adecuado en base a la experiencia. La presentación y abordaje de cada tema en un nivel de abstracción conceptual, sin necesidad de llegar a los detalles intrincados de cada protocolo. Por otro lado, el uso de Internet permite al estudiante comprender

rápidamente el funcionamiento de la red de redes Internet, y prepararlo para afrontar cualquier tecnología de redes.

4 Currículas Analizadas

4.1 Currículas Internacionales

Se han analizado diversos programas de estudio de distintas universidades. Para ello se revisaron los cursos de ejemplo que presenta la currícula ACM – IEEE 2013 [9], y además se tomó en cuenta el ranking elaborado por Centro de Universidades de Clase Mundial de Shanghai Jiao Tong University (CWCU) sobre las universidades de todo el mundo, que tiene en cuenta artículos indexados por SCIE (Science Citation Index-Expanded), los criterios otorgados para la construcción del ranking pueden ser consultados en el sitio [10]. Teniendo en cuenta este ranking para Computer Science y los tres ejemplos de la currícula IEEE-ACM que aparecen en el documento, se seleccionaron además las cuatro universidades de Estados Unidos mejor posicionadas y siete fuera de Estados Unidos. De todas estas universidades, se realizó un análisis de las planificaciones de cursos de Redes de Computadoras.

El reporte de la ACM-IEEE, presenta tres modelos de contenidos sobre Redes, los cuales son analizados a continuación:

Computer Networks I, Case Western Reserve University Cleveland, Prof. Vincenzo Liberatore. (pág. 318). La organización por semana del curso se inicia por una unidad introductoria, luego la capa de aplicación y siguiendo el enfoque descendente hasta llegar a redes LAN y Ethernet. Finaliza con una unidad de sobre wireless y redes móviles.

La Universidad de Stanford ofrece el curso CS144: Introduction to Computer Networking, a cargo de Phil Levis and Nick McKeown (pág.320) (cs144.stanford.edu). También expone un enfoque descendente, comenzando con una unidad de introducción, luego aplicaciones de red, entrega confiable, routing, redes LAN, finalizando con interface de comunicación, agrega además después unidades de desarrollo teórico como máquina de estado finito, confiabilidad y asignación de recursos.

Computer Networks, Williams College Williamstown, Massachusetts Thomas Murtagh. (pág. 323 del informe). ([http:// www.cs.williams.edu/~tom/courses/336](http://www.cs.williams.edu/~tom/courses/336)). Esta Universidad presenta un enfoque mixto, dado que comienza con HTTP, en la segunda semana continúa con redes de difusión y redes Ethernet, luego capa de red, capa de transporte, seguridad y finaliza con la capa física.

De las Universidades citadas en ranking y mejor posicionadas de USA, se analizó:

Instituto tecnológico de Massachusset (MIT) [11]: presenta el curso Computer Networks (otoño 2002). Está estructurado en tres módulos: Internetworking y ruteo, administración de recursos y servicios de red. Fundamentalmente los temas son IP y

TCP. Tiene un enfoque mixto, dado que analiza red y transporte para finalizar con inalámbrico, MAC y multicast. No sugiere ningún libro de texto pero sí numerosos artículos para analizar.

Universidad de Berkeley [12] ofrece el curso Computer Networks (2014). Presenta como bibliografía los libros Peterson, Davie[5], Kurose, Ross[8] y Stevens[6]. Este curso solo trata temas de interconexión y ruteo, no es una asignatura de redes completa. Se divide en las siguientes cuatro partes: Internetworking, Infraestructura e Intra-domain, Datacenters and SDN y finalmente Herramientas, Técnicas y Redes no tradicionales. Si bien sigue un enfoque descendente, no se puede destacar debido a que no es un curso sobre todas o gran parte de las capas.

Universidad de Princeton: el curso presentado es Computer Networks [13]. Incluye los temas: capa de red y protocolos, descubrimiento y ruteo, asignación distribuida de recursos, Aplicaciones de Internet y casos de estudio sobre redes. Como libro principal recomienda Peterson, Davie [5]. Presenta un enfoque no orientado a capas sino a la funcionalidad, empezando por protocolos (para capas de enlace, capa de red, capa de transporte y capa de aplicación), luego asignación de recursos, ruteo y descubrimiento, Aplicaciones de Internet (CDN, P2P, overlay networks), estudio de casos (redes para centro de datos, inalámbricas, Ad-hoc, Backbone) y seguridad.

Universidad de Harvard: presenta el curso Computer Networks [14], que también desarrolla el curso de forma transversal y no por capas incluyendo los temas: Conceptos básicos de redes, Internet de las cosas, redes de bajo consumo (BLE), Redes para Centro de datos, Redes definidas por el software, Redes a escala Web, Redes de entrega de Contenido, Seguridad, Redes inalámbricas, Machine Learning Assisted Networking y Redes vehiculares y Cyber-physical.

Universidad de Toronto: el curso Computer Networks [15]. Es introductorio y su contenido es: sistemas de packet switching, programación de sockets, protocolos software y hardware de red, naming y direccionamiento, control de congestión, redes definidas por el software, seguridad en red y redes inalámbricas. El énfasis del curso es la programación de aplicaciones y de red. Libro principal sugiere: Peterson, Davie [5], y como libros recomendados: Stevens, Rago [27] y Stevens [6].

El Instituto de Tecnología Federal Suizo de Zurich [16]: presenta la materia Communication Networks (primavera 2016). Utiliza como libro principal de texto a Kurose, Ross [8]. Solo se enfoca en Internet, capa de Red y transporte, ruteo, UDP y TCP, SDN y seguridad. No trata la capa aplicación (salvo en seguridad) y no tiene ningún concepto de la capa física. Sigue un enfoque descendente aunque no es curso completo de redes.

Technion Israel Institute of Technology [17]: presenta el curso Introduction to Computer Networks. Como libros de texto indica a los autores: Halsall [28] y Tanenbaum, Wetherall [3]. Utiliza un enfoque ascendente comenzando por introducción, capa de enlace, teoría de colas, protocolos MAC, LANs, Broadband, LAN bridging y TCP/IP.

Universidad de Cambridge [18]: presenta el curso Computer Networking. Utiliza aproximación descendente y es un curso completo de redes con los siguientes contenidos: introducción, capa de enlace, wireless, IP, ruteo, capa de transporte (UDP/TCP), capa de aplicación, Redes Multimedia y Redes para centro de datos. Los libros recomendados por los profesores son: Peterson [5], Kurose[8], Comer [7] y Stevens, Rago [27].

Universidad Nacional de Taiwan [19]: ofrece el curso Computer Networks, Como libro de texto sugiere a los autores Kurose y Ross [8], y como libros de referencia a Peterson y Davie [5], y Stevens [6]. Cubre los principios básicos de redes con un enfoque descendente y énfasis en los protocolos, sus implementaciones y aspectos específicos de Internet. El contenido del curso incluye: Redes de Computadoras e Internet, Capa Aplicación, Transporte, Red, redes inalámbricas y móviles, redes Multimedia, Seguridad en Redes y Administración.

Universidad Tsinghua (China): presenta el curso Advanced Computer Network. Trata los principios del diseño de redes de computadoras, la arquitectura de Internet, protocolos de red. Cubre los tópicos: LAN y Ethernet, Internetworking e IPV6, ruteo, multicast, TCP y control de congestión, P2P y redes superpuestas, móvil e inalámbrico, seguridad y futuro de Internet. Sugiere los libros de Stevens [6] y Stevens, Rago [27]. Utiliza un enfoque ascendente en la mayoría de las unidades.

De las universidades de habla hispana, la Universidad de Granada es la primera que figura en el ranking. Ofrece el curso Fundamentos de Redes. Se imparte por el Depto. Teoría de la Señal, Telemática y Comunicaciones. Sigue el modelo descendente, y sugiere como libros de texto: Pedro G. Teodoro, Jesús D. Verdejo y Juan M. L. Soler. Transmisión de datos y redes de computadores, Ed. Pearson, 2007 y Kurose Ross [8]. Comienza el curso con: Introducción a los Fundamentos de Redes, luego sigue con Servicios y Protocolos de Aplicación en Internet, posteriormente la Capa de Transporte en Internet y finaliza con Redes Conmutadas e Internet. Los aspectos de seguridad están distribuidos en las distintas teorías como son: Protocolos seguros, cortafuegos y VPNs.

4.2 Universidad Nacional de San Juan, Salta, Católica de Salta y Nacional de Santiago del Estero

El presente trabajo se realizó dentro del marco del protocolo de colaboración entre Consejo de Investigación de la Universidad Nacional de Salta (CIUNSA) y la Facultad de Ciencias Exactas Físicas y Naturales de la UNSJ [2].

4.2.1 UNSJ

En la Facultad de Ciencias Exactas Físicas y Naturales, se dicta la materia Redes en tercer año, para las carreras de Licenciatura en Sistemas de Información y Licenciatura en Ciencias de la Computación (la misma materia para ambas carreras).

Es una sola materia en el plan de estudios vigente, que trata todos los contenidos sobre redes de computadoras, según la resolución 786 [23] y que se han actualizado conforme han surgido recomendaciones curriculares por parte de la red UNCI [22].

La evaluación de los estudiantes se realiza por medio de 2 parciales y diversos prácticos áulicos y en laboratorio, para ello se utiliza software como simuladores de redes y analizadores de tráfico entre otros [24].

En el año 2013, se realizó el cursado con un enfoque mixto, con algunas unidades en orden descendente y otras ascendente. A partir de los resultados obtenidos y la experiencia pedagógica, se decidió a partir del ciclo 2014, cambiar el orden del dictado de la asignatura y utilizar el libro de Kurose, Ross [8]. Para planificar adecuadamente el nuevo curso, se tuvo que redefinir la unidad introductoria (de conceptos generales) la cual permitió también rediscutir el alcance de cada una de las unidades.

Un número reducido de estudiantes que recurieron a la materia fueron entrevistados por el equipo de cátedra y afirmaron que les resultó más fácil de entender los contenidos con el nuevo enfoque. Si bien, los estudiantes al recurrir ya tenían conocimientos de redes, al haber cursado anteriormente la asignatura, sus afirmaciones deben ser tenidas en cuenta. Asimismo, se redujo el porcentaje de desaprobados en ambos parciales con respecto a años anteriores cuando no se aplicaba el enfoque descendente.

Los datos relevados indican que durante 2012 y años anteriores (con el enfoque tradicional), el porcentaje de estudiantes que obtuvieron la regularidad estaba entre un 45 y 60%, en 2013 con un enfoque mixto aprobaron un 75,87% y en 2014 ya con el enfoque descendente hubo un 85,24% y en 2015 76,19%.

4.2.2 UNSa

En la Universidad Nacional de Salta se imparte la carrera Licenciatura en Análisis de Sistemas. En su plan de estudios vigente del año 2010, los contenidos sobre redes de computadoras están incluidos en dos asignaturas: Redes de Computadoras I y Redes de Computadoras II en el cuarto año, primer y segundo cuatrimestre respectivamente. Dichos contenidos fueron actualizados conforme a las recomendaciones curriculares de la red UNCI [22]. Ambas asignaturas siguen el nuevo enfoque descendente desde su implementación en el año 2013. La evaluación de los estudiantes en cada asignatura, se realiza por medio de cuatro exámenes parciales (uno para cada tema del programa). En la práctica se utilizan simuladores y como guía el libro desarrollado por la Cátedra denominado Redes de Computadoras I con Packet Tracer [26], que acompaña cada tema del libro de base de Kurose y Ross [8], con ejercicios prácticos resueltos con el simulador.

Cuando el estudiante llega a cursar la asignatura Redes de Computadoras I, trae consigo un bagaje de conocimiento sobre el manejo de los servicios más destacados en Internet y su funcionalidad desde el punto de vista del usuario, esto es, navegadores, clientes FTP, clientes e-mails, servicios DNS, aplicaciones de telefonía IP y muchos otros. El enfoque descendente parte de este conocimiento para primeramente tratar los protocolos de capa de aplicación que implementan estos servicios, protocolos HTTP, FTP, SMTP, DNS. Este primer acercamiento a los

fundamentos sobre protocolos, se centra en la simplicidad de protocolos como HTTP y SMTP, para luego analizar los protocolos más complejos como FTP y DNS, siempre resaltando los conceptos necesarios sobre la capa de transporte y la interfaz de programación Sockets. Posteriormente se tratan los fundamentos de capa de transporte y los protocolos UDP y TCP. En este punto también es importante introducir el concepto de dirección IP y el de servicio no fiable que brinda IP. Finalmente se estudia la capa de red, el protocolo IP y los protocolos asociados como son protocolos ICMP, DHCP y protocolos de ruteo, en esta instancia también se hace necesario introducir algunos conceptos de la capa de enlace.

La asignatura Redes de Computadoras II, continuación de la anterior, trata inicialmente la capa de enlace y redes de área local para luego adentrarse en temas más avanzados sobre seguridad de redes y gestión de redes.

Como se puede advertir, en el enfoque descendente se hace necesario, cuando se trata una determinada capa del modelo TCP/IP, introducir algunos conceptos de la capa inferior, por ejemplo, cuando se explica que los datos son encapsulados en la estructura de datos de la capa inferior. Quizás, esto sea lo único criticable de este enfoque, sobre todo para los que se formaron con el enfoque tradicional, se podría decir, que el tratamiento de una capa en particular “no es tan limpio como en el enfoque tradicional”. Aunque, finalmente esto parece ser más una percepción de los que estudiaron con el enfoque ingenieril que de los estudiantes que están muy gustosos y motivados con el nuevo enfoque descendente.

La mayor ventaja que se observa del enfoque descendente, es su dirección pedagógica y didáctica en lo que respecta al aprendizaje de redes de computadoras. La presentación y abordaje de cada tema en un nivel de abstracción conceptual, sin necesidad de llegar a los detalles intrincados de cada protocolo, lo que quizás sea más aplicable al libro de los autores Kurose y Ross que al propio enfoque.

Realizando una comparación la asignatura Conectividad y Teleinformática perteneciente al viejo plan de estudios del año 1997, se pudo observar en el período 2013-2015 un mayor porcentaje de estudiantes regulares en cada cursada. Cabe aclarar que esta asignatura a la fecha, ya se encuentra extinta.

En la experiencia en la Universidad Nacional de Salta, este enfoque permitió abordar temas como seguridad y gestión de redes que nunca antes, con el enfoque tradicional ascendente, se había podido brindar a los estudiantes por cuestiones de tiempos.

4.2.3 UCASAL

A raíz del convenio existente entre la Universidad Nacional de Salta y la Universidad Católica de Salta [2], se pudo realizar la transferencia necesaria para la implementación del nuevo enfoque en la UCASAL. Actualmente y desde el año 2015, la asignatura Redes I del tercer año de la carrera Ingeniería en Telecomunicaciones de la UCASAL, se basa en un enfoque mixto. Esta asignatura abarca todas las capas del modelo TCP/IP. Si bien el libro de base utilizado es el de Kurose y Ross y la parte teórica sigue el nuevo enfoque, la práctica todavía se basa en el enfoque tradicional ascendente. Esta mezcla dificulta poder abarcar toda la temática que se estudia en la teoría, en la práctica, ya que esta comienza con temas más vinculados a la capa de

enlace y protocolos como ARP, para luego analizar casos concretos de configuración sobre redes IP, VLANs, etc. La asignatura Redes II, es la continuación de la primera y abarca la temática sobre redes WAN.

En la planificación, se prevé para el próximo año 2017, realizar la implementación total del enfoque descendente en la primera asignatura Redes I.

4.2.4 UNSE

En la Universidad Nacional de Santiago del Estero, Facultad de Ciencias Exactas y Tecnológicas, se imparte la carrera Licenciatura en Sistemas de Información. En su plan de estudios del año 2011, los contenidos sobre redes de computadoras se desarrollan en dos asignaturas Redes I [25] en el tercer año y Redes II en el cuarto año. Si bien, como parte de la bibliografía de base se propone el libro de los autores Kurose y Ross [8], ninguna de las dos asignaturas sigue el enfoque descendente. Los contenidos de la asignatura Redes I, sigue el diseño tradicional ascendente, comenzando con una introducción al modelo de comunicación, transmisión de datos, enlace de datos y redes de área local LAN. Se han iniciado conversaciones entre responsables de asignaturas de la UNSa, UNSJ y UNSE con el objeto de redactar convenios de cooperación en diversos aspectos del dictado de redes entre ellos la aplicación del enfoque descendente y la utilización de herramientas de simulación.

5 Conclusiones

El enfoque descendente permite a los estudiantes colocar el énfasis inicial en la capa aplicación, que prepara los fundamentos hacia la computación distribuida, donde se están produciendo los cambios más importantes y notables de los últimos años como son: Cloud Computing, computación ubicua, Redes Sociales, Redes Superpuestas y Big Data (desde el punto de vista de la comunicación de los componentes) entre otros.

En un futuro cercano ciertos conceptos como SDN (Redes Definidas por el Software), virtualización de red, Redes definidas por las aplicaciones, IoT serán claves para la Internet del futuro.

Por otro lado, los estudiantes se entusiasman debido a que encuentran los conceptos iniciales, o sea la capa aplicación, más “familiares” y fáciles de entender, y luego van apropiándose de los conceptos de TCP/UDP, Internetworking e IP (donde comprenden la complejidad de las redes) y luego los LAN, WLANs, WANs y tecnologías para acceso a la red. La seguridad es sumamente importante y es conveniente analizarla en conjunto con las otras unidades a medida que son desarrolladas, tanto como sea posible y no como una unidad inconexa.

De los catorce cursos internacionales analizados se obtienen los siguientes datos: seis de los mismos siguen el enfoque descendente. Dos de ellos un enfoque mixto (algunas capas de forma descendente y otras ascendente), dos cursos solo tratan IP y TCP (no son cursos completos de redes) y al igual que las dos que incluyen un enfoque transversal (no orientado a capas) se complementan con otros cursos para la enseñanza de la capa aplicación o temas faltantes.

6 Referencias

1. Berglund Anders. What Is Good Teaching of Computer Networks?. 33rd ASEE/IEEE Frontiers in Education Conference. (2003).
2. Protocolo de colaboración entre el Consejo de Investigación de la Universidad Nacional de Salta (CIUNSa) y la Facultad de Ciencias Exactas Físicas y Naturales de la UNSJ.
3. Andrew Tanenbaum, David Wetherall. Computer Networks 5th Edition. Pearson New International Edition (2013).
4. William Stallings. Data and Computer Communications, Tenth Edition. Pearson Prentice Hall. (2014).
5. Larry L. Peterson and Bruce S. Davie. Computer Networks a systems approach. Fifth Edition. Morgan Kaufmann Publishers - Elsevier. (2012).
6. W. Richard Stevens. TCP/IP Illustrated, Vol. 1: The Protocols 2da Edition. (Addison-Wesley Professional Computing Series). (2012).
7. Douglas Comer. Computer Networks and Internets (6th Edition). Pearson Prentice Hall (2014).
8. J.F. Kurose, K.W. Ross. Computer Networking: A Top-Down Approach (6th Edition). Pearson Education. (2015).
9. Computer Science Curricula 2013. The Joint Task Force on Computing Curricula ACM - IEEE Computer Society: <http://www.acm.org/education/CS2013-final-report.pdf>
10. Academic Ranking of World Universities. <http://www.arwu.org/>
11. MIT: Computer Networks. <http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-829-computer-networks-fall-2002/>
12. Universidad de Berkeley: <http://people.eecs.berkeley.edu/~sylvia/cs268-2014/syllabus.html>
13. Universidad de Princeton: <http://www.cs.princeton.edu/courses/archive/spr14/cos461/>
14. Universidad de Harvard: <https://locator.tlt.harvard.edu/course/colgsas-118418>
15. Universidad de Toronto: <http://www.cdf.toronto.edu/~karaman/csc458/CourseInfo.pdf>
16. Swiss federal Institute of Technology Zurich: <https://www.inf.ethz.ch/studies/bachelor.html>
17. Technion Israel Institute of technology: <https://webcourse.cs.technion.ac.il/236334/Spring2016/>
18. Universidad de Cambridge: <https://www.cl.cam.ac.uk/teaching/1314/CompNet/>
19. National Taiwan University: <http://www.cmlab.csie.ntu.edu.tw/~rod24574575/CN2015/>
20. Universidad Tsinghua: <http://thu-cmu.cs.tsinghua.edu.cn/curriculum/ACN/FIA.htm>
21. Universidad de Granada: http://dtstc.ugr.es/it/gitt_It/index.php
22. Documentos curriculares Red UNCI, <http://redunci.info.unlp.edu.ar/documentos.html>
23. Resolución 786/2009. Ministerio de Educación. Educación superior. Junio 2009.
24. Arias Figueroa, D. (UNSa), Díaz, J. (UNLP), Rodríguez, N. (UNSJ) et al.: Estudio de la influencia del uso de simulación en la enseñanza de redes de computadoras en el nivel universitario (resultados parciales). In: XVIII WICC (2016).
25. Programa Analítico de la asignatura Redes I de la Universidad Nacional de Santiago del Estero. <http://fce.unse.edu.ar/sites/default/files/pdf/asignatura/REDES%20I.pdf>
26. Daniel Arias Figueroa. Redes de Computadoras I con Packet Tracer. Editorial de la UNSa - EDUNSa. ISBN 978-987-633-132-6-1; 1a ed. - Salta - E-Book. 2015.
27. Richard Stevens y Stephen Rago.: Advanced Programming in the UNIX Environment, 3rd Edition. Pearson Education Inc. (2013).
28. Fred Halsall Data Communications, Computer Networks, and Open Systems 4th Edition. Addison Wesley (1996)

ALGOLIPSE: una herramienta educativa para mejorar la comprensión de algoritmos y estructuras de datos

Laura Fava¹, Alejandra Schiavoni¹, Jorge Rosso¹, Facundo Falcone¹,
Lisandro Ronconi¹,

¹ LINTI - Laboratorio de Investigación en Nuevas Tecnologías Informáticas.
Facultad de Informática. Universidad Nacional de La Plata
Calle 50 esq. 120, 2^{do} Piso. Tel: +54 221 4223528
{lfava, ales, jrosso}@info.unlp.edu.ar, faku_f@hotmail.com, lisandro.ronconi@gmail.com

Resumen. En Ciencias de la Computación la enseñanza de las estructuras de datos es de suma importancia puesto que representan la base para el desarrollo de toda clase de aplicaciones. El proceso de enseñanza-aprendizaje de estos conceptos acontece en los primeros años de las carreras, por lo que resultan mas complejos de aprehender debido, entre otras cosas, a la falta de abstracción observada en los estudiantes. En este contexto, surgieron numerosas herramientas de visualización de algoritmos, atractivas tanto para educadores, como para estudiantes, pero que no han sido ampliamente adoptadas debido al esfuerzo que implica su uso. En este trabajo, se presenta una extensión para Eclipse que genera automáticamente visualizaciones de estructuras de datos y algoritmos a partir de código JAVA implementado por los estudiantes y no intervenido. Esta propuesta ayuda a reducir la abstracción que conlleva el estudio de estos temas sin el esfuerzo extra que implica aprender una nueva herramienta, lenguaje o librerías especiales para la visualización de algoritmos.

Palabras claves: visualización de estructuras de datos, visualización de algoritmos, recursión, programación orientada a aspectos (POA), JAVA.

1 Introducción

En Ciencias de la Computación el estudio de las estructuras de datos, es fundamental dado que las mismas son utilizadas en el desarrollo de casi todo tipo de software. Estudiar estructuras de datos implica estudiar las diferentes maneras de almacenar y organizar datos para facilitar el acceso y modificación [1]. El propósito principal de la mayoría de los programas no es procesar información sino almacenarla y recuperarla, en lo posible de manera rápida. Las estructuras de datos, en términos generales, organizan o estructuran una colección de datos, requieren de un determinado espacio para almacenarlos y de una cantidad de tiempo para accederlos. En consecuencia, cuando se necesita resolver un problema, comúnmente se necesita seleccionar una determinada estructura de datos

considerando las restricciones de espacio y de tiempo. La introducción de estos temas complejos, a través de explicaciones teóricas y mediante su refuerzo con actividades prácticas, muchas veces no alcanza para que los estudiantes sean capaces de comprender el funcionamiento de todas las estructuras, poder determinar cuál es la mejor y codificar los algoritmos que representan una solución.

Los temas relacionados con la algorítmica y el uso de las estructuras de datos aparecen en asignaturas de los primeros años de las carreras de Ciencias de la Computación, lo que aumenta la complejidad de su enseñanza-aprendizaje. En particular, en la Facultad de Informática de la Universidad Nacional de La Plata, la materia que comprende estos contenidos es Algoritmos y Estructuras de Datos, asignatura obligatoria que corresponde a segundo año de las carreras de grado: Licenciatura en Informática, Licenciatura en Sistemas y Analista Programador Universitario. Además, en segundo año de la carrera de Ingeniería en Computación se dicta la materia Programación 3, también de carácter obligatorio, que incluye los temas mencionados. En estas asignaturas se abordan los temas relacionados con el uso y aplicación de estructuras de datos básicas y avanzadas, algoritmos que permiten la manipulación de dichas estructuras y también el análisis de la eficiencia de los mismos.

Para que un curso sobre estructuras de datos alcance el éxito deseado debe contar con una componente fuerte de programación de las diferentes estructuras que se estudian [2], por ello, cada una de las estructuras enseñadas tiene su parte práctica para reforzar la comprensión de las mismas. En muchos casos se observa que los estudiantes, aun conociendo las estructuras donde se guardan los datos y comprendiendo su organización, les resulta difícil entender el funcionamiento de un algoritmo con sólo leerlo. Donald Knuth, menciona que la mejor forma de aprender y entender lo que hace un algoritmo es probarlo mediante un ejemplo [3].

Con el fin de mejorar la comprensión de algoritmos, en los últimos años han aparecido numerosas herramientas de visualización y animación que permiten complementar el análisis, resultan muy útiles en procesos de auto-formación y favorecen la comprensión del problema en sí. Sin embargo, algunas se basan en contextos estáticos, no permitiéndole a los alumnos ejecutar su propio código, otras requieren esfuerzos extras para ser usadas, lo que desalienta su uso. En su mayoría, las propuestas resultan atractivas para los educadores quienes las consideran como ayudas pedagógicas, sin embargo no han logrado alcanzar popularidad para la enseñanza en ciencias de computación. Algunas razones podrían deberse a las escasas mejoras de aprendizaje observadas en comparación con el esfuerzo que implica usarlas. Muchas de ellas, como veremos más adelante en este artículo requieren que los estudiantes aprendan lenguajes o librerías específicas para escribir los algoritmos, otras incluyen códigos estáticos donde los estudiantes sólo pueden ver la animación de esos códigos y a menudo los aprendizajes son pasivos, es decir, el estudiante no puede interactuar con las estructuras y algoritmos.

Este artículo describe la arquitectura e implementación de una herramienta educativa para facilitar el aprendizaje de algoritmos de acceso a las estructuras de datos, que intenta paliar estas dificultades. Este entorno podría ser usado como un complemento novedoso de las clases teóricas y prácticas en las que el alumno incorpora los conceptos y los aplica

en el desarrollo de sus propios algoritmos. En las próximas secciones se describirán las características más relevantes de algunas herramientas existentes con objetivos similares a la propuesta en este artículo. Luego, se expondrá la arquitectura y funcionalidad del entorno propuesto, y por último, la implementación de la herramienta mencionando la metodología y las tecnologías que se utilizaron.

2 Estado del arte: Análisis de herramientas existentes

Desde hace tiempo se vienen estudiando mecanismos complementarios al material teórico-práctico usado tradicionalmente para la enseñanza de las estructuras de datos y algoritmos. A continuación se describirán brevemente algunas herramientas seleccionadas que dan cuenta de distintos enfoques, todas con el fin de mejorar el aprendizaje de las estructuras de datos.

La primera herramienta que se describe es VisuAlgo [4], un software interactivo y web para la visualización de algoritmos clásicos. Es interactivo en el sentido que los estudiantes pueden ingresar datos y ejecutar ciertos algoritmos predefinidos. Durante la ejecución se despliega un panel que visualiza un script predefinido y fijo con la lógica del mismo. VisuAlgo permite visualizar el manejo de las estructuras básicas y algoritmos asociados a través de una interfaz de usuario gráfica, moderna y unificada, que incluye desde algoritmos básicos de manejo de listas hasta algoritmos complejos vinculados con grafos.

Otra herramienta analizada provee animación de algoritmos simples definidos por el usuario pero a través del uso de un lenguaje propio llamado JavaMy [5]. Para la escritura del código, cada usuario puede usar el editor de textos provisto o importarlo de un archivo. El uso de esta herramienta le implica al usuario el esfuerzo de aprender un lenguaje específico, que si bien tiene una sintaxis similar a Java, presenta algunas diferencias en cuanto a la definición y organización de los archivos fuente.

CADILAG es otra herramienta que está siendo desarrollada en la Universidade Estadual Paulista, en Presidente Prudente, Brasil [6]. Fue desarrollada en JavaFX, permitiendo que sea utilizada como aplicación de escritorio o basada en web. Ofrece un conjunto limitado de estructuras de datos con las cuales trabajar, y a partir de la selección de una de ellas, la herramienta permite visualizar el comportamiento de sus operaciones básicas. Si bien, es posible interactuar de manera amigable con las estructuras de datos y comprender cómo funcionan las operaciones, no es posible que los estudiantes verifiquen el funcionamiento de su propio código.

Otra herramienta analizada es jGRASP, el último entorno de desarrollo implementado por el grupo de investigación GRASP de la Universidad Auburn en Alabama, USA [7]. Es un ambiente de desarrollo integrado, de libre acceso, que provee visualizaciones generadas automáticamente para mejorar la comprensión de las estructuras de datos. En relación al enfoque de nuestro análisis, esta herramienta cuenta con visualizadores dinámicos que intentan reconocer automáticamente las estructuras a partir del código Java

y graficarlas en forma adecuada. Si bien la herramienta es muy interesante al momento de visualizar las estructuras, representa un ambiente de desarrollo específico para este fin, que no acompaña de una manera amigable la escritura del código de los algoritmos, dado que no brinda ayudas contextuales, respecto a paquetes, clases y métodos disponibles.

Finalmente, sintetizamos las funcionalidades provistas por otra herramienta analizada denominada Algorithm Visualizer [8]. Esta herramienta es basada en web y provee un conjunto de algoritmos implementados sobre distintos temas: recorridos sobre árboles y sobre grafos, algoritmos de ordenación, algoritmos clásicos de programación dinámica y de backtracking, entre otros. Esta herramienta es muy completa y ofrece también la posibilidad de ingresar un código propio, pero usando un API propia llamada Tracer API.

Cada herramienta descrita tiene su atractivo. En particular, la herramienta que se propone en este artículo tiene un enfoque similar a jGRASP, puesto que trabaja sobre código Java elaborado por los estudiantes sin ningún tipo de restricción y propone una visualización automática a partir del mismo, sin embargo, nuestra propuesta está basada en la extensión de las capacidades de Eclipse y no en la implementación de una herramienta independiente como es JGrasp. En las siguientes secciones se describen la arquitectura y las funcionalidades de la herramienta propuesta, y la implementación realizada.

3 Diseño de la herramienta propuesta

La herramienta propuesta está orientada a ofrecer una representación visual de la ejecución de algoritmos sobre estructuras de datos y la visualización de los llamados de algoritmos recursivos, que involucra una forma distinta de pensar y encarar la solución a problemas [9, 10]. Dado que JAVA es el lenguaje utilizado para la implementación de las estructuras de datos en las asignaturas Algoritmos y Estructuras de Datos y Programación 3, se tuvieron en cuenta soluciones basadas en este lenguaje. En las actividades prácticas, los estudiantes utilizan Eclipse como ambiente de desarrollo, motivo por el cual se decidió extender este entorno en lugar de desarrollar una herramienta diferente. De esta manera, los estudiantes seguirán usando Eclipse, un entorno de amplia aceptación en la comunidad de desarrolladores JAVA, con el agregado de un nuevo plug-in denominado Algolipse.

La definición de la herramienta involucra principalmente la creación de un plug-in para la visualización de las estructuras de datos y de los llamdos recursivos y el uso de programación orientada a aspectos para interceptar la ejecución del código.

3.1 Arquitectura del plugin Algolipse en el entorno Eclipse

Eclipse es un framework de código abierto para la creación de entornos de desarrollo integrados utilizando un conjunto de herramientas y componentes GUI pre-construidas.

Desde su lanzamiento, ha generado gran interés en la comunidad de desarrolladores JAVA tanto por sus funcionalidades como por sus capacidades de extensión. El workbench de Eclipse se compone de editores, vistas y perspectivas.

Los *editores* son áreas de edición asociados con tipos de documentos específicos. Por ejemplo hay editores de archivos xml, .java, JSP, etc. brindando ayudas específicas para cada tipo archivo.

Las *vistas* son paneles que apoyan a los editores y en su conjunto, conforman una perspectiva.

Las *perspectivas* son combinaciones de vistas específicas y editores que dan soporte al usuario durante el desarrollo de una aplicación.

La implementación del plug-in¹ [11] propuesto proveerá a Eclipse nuevas *Vistas*, dedicadas a visualizar la ejecución de algoritmos sobre una estructura de datos particular y en el caso de algoritmos recursivos, también se visualizará el estado de las variables en cada llamado recursivo.

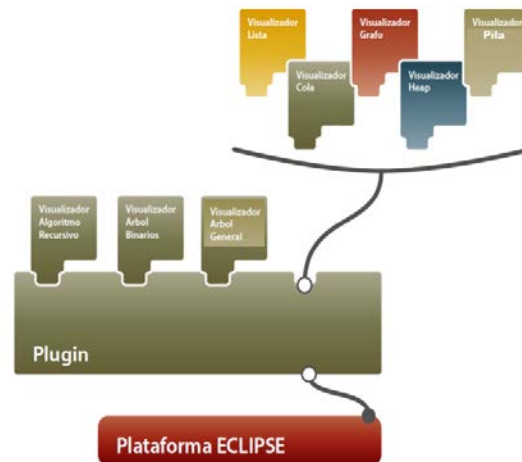


Fig. 1. Arquitectura de Eclipse

La Fig. 1 muestra de manera simplificada la arquitectura de Eclipse, donde se inserta el nuevo plug-in con varias visualizaciones. Puede observarse que es posible agregar nuevas funcionalidades mediante la incorporación de nuevas visualizaciones como Visualizador para Listas, Colas, Pilas, Grafos y otras estructuras.

¹ Un plug-in es una componente de software, es la unidad funcional más pequeña de la plataforma Eclipse que puede ser desarrollada de manera separada e integrada al entorno Eclipse para potenciar su funcionalidad.

3.2 Visualizadores de las Estructuras de Datos

Como ya se describió en el punto anterior, el plug-in definirá nuevas *Vistas* destinadas a visualizar las estructuras de datos y el estado de las llamadas de los algoritmos recursivas. Para interfaces gráficas la plataforma estándar de JAVA, provee librerías con componentes gráficas como Abstract Window Toolkit (AWT) y Swing [12], sin embargo, para el desarrollo de plugin, se recomienda el uso de una librería de componentes gráficas o widgets alternativa denominada Standard Widget Toolkit (SWT) [13]. Esta librería combina lo mejor de AWT y Swing, no es parte de la plataforma estándar JAVA pero se implementó para usar con ella y en especial para Eclipse. SWT fue desarrollada por IBM, y actualmente es mantenida por el Eclipse Fundación en conjunto con el IDE de Eclipse. Para mostrar elementos visuales, la implementación de SWT accede a librerías nativas de GUI del sistema operativo usando JNI (Java Native Interface) en una manera similar a AWT, es decir usando APIs específicas del sistema operativo, pero a diferencia de esta, cuenta con una rica librería de componentes como Swing. Esta librería está licenciada bajo la Eclipse Public License, una licencia open source aprobada por el Open Source Initiative.

La Fig. 2 muestra la arquitectura de la librería de componentes de interfaz de usuario estándar AWT/Swing y SWT. Las primeras se presentan juntas ya que Swing se construyó sobre AWT para proveer componentes más sofisticadas, flexibles y con un *look&feel* independiente de la plataforma, entre otras cosas. A la derecha se muestra las capas de SWT donde se observa como capa superior JFace, compuesto por un conjunto de herramientas de interfaz de usuario de alto nivel que utiliza los *widgets* de SWT para proporcionar componentes basados en modelos. JFace depende de SWT, pero no oculta los widgets SWT, dando la capacidad de elegir entre el desarrollo de interfaz de usuario basado en modelos o mediante manipulación de widgets en crudo.

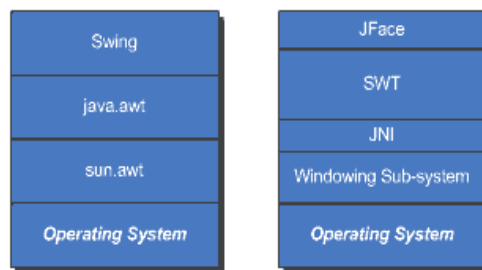


Fig. 2. Arquitecturas de AWT/Swing y SWT

En síntesis, SWT intenta hacer lo mejor de ambas librerías Swing y AWT, define una API portable y proporciona implementaciones para todas las plataformas. Siempre que sea posible, las implementaciones usan widgets nativos. Esto permite que se reflejen los aspectos de la interfaz de usuario gráfica del sistema operativo subyacente.

3.3 Programación Orientada a Aspectos en el plugin Algolipse

Uno de los objetivos principales de la herramienta, fue poder visualizar las diferentes estructuras de datos, sin afectar o intervenir el código desarrollado por los estudiantes. Esto significa que la herramienta debe conocer las clases y métodos que permitan describir la composición de cada estructura y los datos contenidos en ella, sin imponer restricciones. Es decir, el código de los estudiantes no tendrá que cumplir con convención de nombres, ni incluir anotaciones JAVA para indicar de qué tipo de estructura se trata y cuáles son sus métodos de acceso a la información. Por otro lado, utilizar archivos descriptores (con algún formato estándar como XML o JSON) asociados con las estructuras utilizadas, resultaría en una solución un tanto estática.

De esta manera llegamos a la conclusión de que la Programación Orientada a Aspectos nos permitiría resolver este tema de una manera más elegante y dinámica. Con AspectJ [14] es posible indicar puntos de un programa en los cuales uno desea examinar los objetos y realizar alguna determinada acción sin modificar el código original. La especificación de estos puntos de intervención (Pointcuts) y las acciones que se llevarán a cabo se escriben en un aspecto (Aspect).

El momento en que se entrelaza (weaving) el código origen con los aspectos se puede dar en tres momentos: en compilación (compile-time weaving), durante el empaquetado de un archivo JAR, o en ejecución (load-time weaving). La Fig. 3 muestra una esquematización del funcionamiento de AOP para entremezclar los aspectos con los programas.

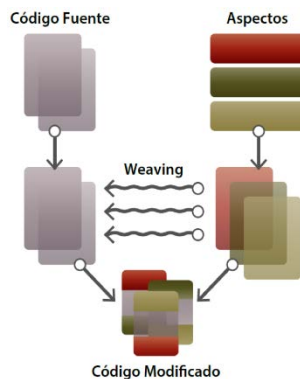


Fig. 3: Weaving de Aspectos con el Código Fuente

Como puede observarse, los aspectos que definen qué código sería interceptado, son independientes del código fuente implementado por los estudiantes. El uso de programación orientada a aspectos nos permite no ser intrusivo en la definición de las estructuras de datos y los algoritmos implementados por los estudiantes.

4 Implementación de la herramienta propuesta

Como se ha mencionado, muchas herramientas no han alcanzado demasiada popularidad debido a los beneficios obtenidos en relación al esfuerzo que implica su uso. Por este motivo, se tomaron dos decisiones importantes de diseño tendientes a evitar la complejidad del uso: la primera está relacionada con mantener a los estudiantes en el entorno de desarrollo habitual para probar sus códigos, por ello se ha implementado un plugin para ECLIPSE y la segunda determinación está relacionada con no obligar a los estudiantes a aprender un nuevo lenguaje, pseudocódigo o API para obtener la visualización de sus producciones, por el contrario, pueden codificar normalmente sus algoritmos en JAVA con la API estándar.

4.1 El plugin Algolisp

El plugin Algolisp fue desarrollado siguiendo los lineamientos para desarrollo de cualquier plugin, su instalación es simple y automáticamente las nuevas Vistas estarán disponibles para usar en cualquiera de las Perspectivas de Eclipse.

Una vez instalado el plugin y agregada la nueva vista en alguna de las perspectivas, Eclipse mostrará como ilustra Figura 4. La nueva *Vista* de nombre *Algolisp* mostrará tres paneles internos: el visualizador de las estructuras de datos, selector de métodos de la clase seleccionada y que pueden ser inspeccionados y si el algoritmo es recursivo, aparecerán ventanas en cascada que representan cada uno de los llamados recursivos con el estado de las variables y la visualización de los algoritmos en ese llamado.

El código que se ejecuta, como ya mencionamos es JAVA. Los alumnos escriben sus clases como ListaEnlazada, ArbolBinario, ArbolAVL, ArbolGeneral, etc., sin ninguna restricción y pueden hacerlo además, con Tipos Genéricos, de manera que una implementación les sirva para instanciar estructuras que mantengan diferentes tipos de datos. Una vez que codifican las operaciones de sus estructuras, pueden visualizar su funcionamiento, indicando mediante la ventana de configuración, qué clase quieren visualizar y que algoritmo o algoritmos. Una vez elegida la clase, en el panel de la derecha aparecen los algoritmos que pueden visualizarse. En el caso de la Figura 4, se seleccionó la clase ArbolGeneral y el método printPreorden(). También podrían seleccionarse más métodos de la lista de algoritmos que pueden aplicarse a la estructura de datos ArbolGeneral.

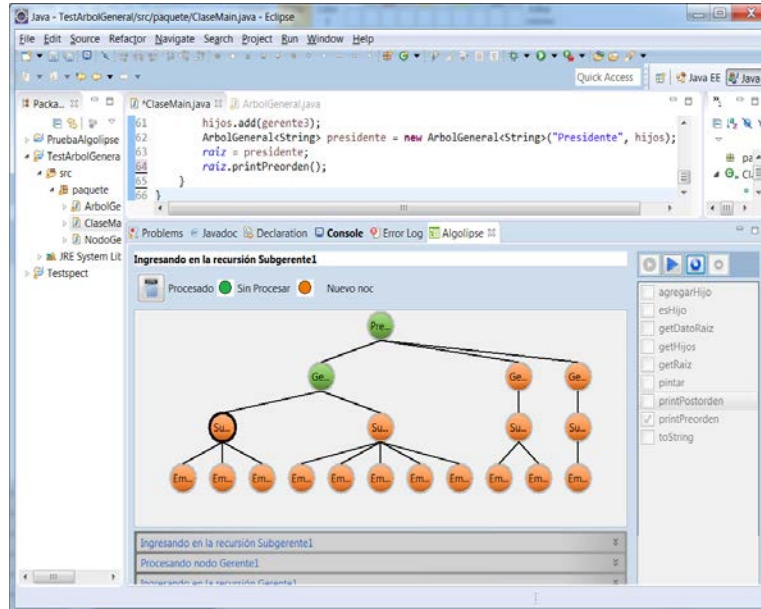


Fig. 4: Vistas del IDE Eclipse durante la ejecución del algoritmo printPreorden

Para ejecutar la aplicación, sólo es necesario crear una clase ejecutable, es decir, una clase que incluya el método main(String[] args). A continuación se muestra, a modo de ejemplo, segmentos código donde se crea el árbol de la Fig. 4 y finalmente se invoca al método printPreorden() que se quiere visualizar.

```

public class ClaseMain {
    public static ArbolGeneral<String> raiz;

    public static void main(String[] args) {
        LinkedList< ArbolGeneral<String> > hijos =
            new LinkedList< ArbolGeneral<String>>();
        hijos.add(new ArbolGeneral<String>("Empleado1"));
        hijos.add(new ArbolGeneral<String>("Empleado2"));
        hijos.add(new ArbolGeneral<String>("Empleado3"));
        ArbolGeneral<String> subGerente =
            new ArbolGeneral<String>("Subgerente1", hijos);
        . . .
        ArbolGeneral<String> presidente =
            new ArbolGeneral<String>("Presidente", hijos);
        raiz = presidente;
        raiz.printPreorden();
    }
}

```

4.2 El uso de AOP para visualizar código JAVA

El segundo aspecto que se tuvo en cuenta al momento de diseñar la herramienta está relacionado con el código JAVA escrito por los estudiantes. Durante la cursada de las asignaturas mencionadas, los alumnos desarrollan prácticas donde definen sus propias estructuras de datos, según cierta especificación UML, como Listas, Pilas, Colas, Arboles y Grafos y abundantes algoritmos asociados. Para permitir la visualización de los algoritmos a partir de código JAVA no modificado, utilizamos programación orientada a aspectos. El uso de aspectos permite generar visualizaciones automáticamente a partir de código JAVA sin necesidad de utilizar una API especial, metadatos como anotaciones o archivos de configuración XML. Los estudiantes codifican, compilan y ejecutan para visualizar.

La manera más apropiada que encontramos para interceptar la ejecución de un método, elegido en tiempo de ejecución, es decir durante la ejecución de códigos compilados, fue el uso de programación orientada a aspectos. De esta manera, cuando se alcanza una llamada a un método previamente seleccionado desde la interface de usuarios gráfica, el aspecto lo detectará para comenzar a actuar. Cabe destacar que para proveer mayor *feedback*, la visualización además muestra mediante diferentes colores que nodos han sido procesados y cuáles no. Para lograr esto también se ha utilizado aspectos.

5 Conclusiones y trabajos futuros

En este artículo presentamos una herramienta destinada a mejorar el aprendizaje de algoritmos y estructuras de datos, destinada a los alumnos de los primeros años de las Carreras de Ciencias de la Computación. Los estudiantes pueden escribir sus propios algoritmos en JAVA dentro de ECLIPSE como lo hacen habitualmente y observar la visualización de los mismos. Lo destacable de la herramienta es que la animación de los algoritmos depende en forma directa del código que está ejecutando el alumno. El otro aspecto a resaltar, es la posibilidad que brinda la herramienta para realizar el seguimiento y análisis de los llamados recursivos, pudiendo inspeccionar el estado de las variables en cada llamado.

Atendiendo al uso permanente que hacen los jóvenes de las nuevas tecnologías y a la fuerte incorporación de las TICs en educación, consideramos que esta herramienta va a ser un complemento efectivo a la enseñanza tradicional. A partir de los conocimientos adquiridos de las clases teóricas, la herramienta asistirá a los estudiantes en la creación de sus propios algoritmos, ayudando a disminuir la abstracción inherente al estudio de las estructuras de datos y algoritmos de acceso.

En esta primera versión, se implementó el Visualizador para Algoritmos Recursivos y los Visualizadores para Listas, Pilas, Colas, Arboles Binarios y Arboles Generales. Actualmente se están comenzando a hacer las primeras pruebas de usabilidad de la herramienta para evaluar diferentes aspectos relacionados con la facilidad de uso, claridad

de la interfaz y eficacia en el funcionamiento. En una próxima etapa, se prevé realizar tests con alumnos de las materias mencionadas a fin de evaluar el impacto en el aprendizaje.

Se espera poder incorporar Algolipse como complemento innovador a las clases teóricas, explicaciones prácticas y textos recomendados por las cátedras. Asimismo esto permitirá retroalimentar la aplicación con sugerencias tanto por parte de los docentes como de los mismos estudiantes.

Referencias

1. T. Cormen, C. Leiserson, R. Rivest, C. Stein; Introduction to Algorithms, Second Edition, ISBN 0-07-013151-1 (McGraw-Hill), 2001.
2. C. Shaffer, Data Structures and Algorithm Analysis, Edition 3.2 (Java Version), Department of Computer Science, Virginia Tech, Blacksburg, VA 2406, January 2, 2012, Update 3.2.0.3.
3. D. Knuth, The Art of Computing Programming, Volume 1: Fundamental Algorithms, Third Edition, ISBN 0-201-89683-4 AddisonWesley Longman, 1997.
4. S. Halim, Z. Koh, V. Loh, F. Halim, Learning Algorithms with Unified and Interactive Web-Based Visualization, Olympiads in Informatics, 2012, Vol. 6, 53–68, Vilnius University, 2012.
5. T. Chen and T. Tarek Sobh; A Tool for Data Structure Visualization and User-defined Algorithm Animation, Department of Computer Science and engineering, University of Bridgeport, USA. Accesible en: <http://www1bpt.bridgeport.edu/~risc/pdf/jp29.pdf>
6. G. Cardim, I. Marcal, C. de Sousa, D. de Campos, C. Marin; A. do Carmo, D. Toledo, A. Saito, R. Correia, R. Garcia, Teaching and Learning of Data Structures Supported by Computer: An Experience with the CADILAG tool. Information Systems and Technologies (CISTI), 2012 7th Iberian Conference, Madrid, p:1-5. ISBN 978-1-4673-2843-2
7. J. Cross, D. Hendrix; Workshop jGRASP: An Integrated Development Environment with Visualizations for Teaching Java in CS1, CS2, and Beyond, 36th ASEE/IEEE Frontiers in Education Conference, ISBN 1-4244-0256-5, 27 -31 Oct. 2006.
8. Algorithm Visualizer. Disponible en GitHub: <https://www.producthunt.com/tech/algorithm-visualizer>
9. J. Zhang, M. Atay, E. Smith, E. Caldwell, E. Jones, Using a Game-Like Module to Reinforce Student Understanding of Recursion, Frontiers in Education Conference (FIE), IEEE, ISBN: 978-1-4799-3921-3, Madrid, Spain, 22 – 25 Oct, 2014.
10. A. Chaffin, K. Doran, D. Hicks, T Barnes, Experimental Evaluation of Teaching Recursion in a Video Game, Proceedings of the 2009 ACM SIGGRAPH, Symposium of Video Games, 79-86.
11. A. Blewitt, Eclipse 4 Plug-in Development by Example: Beginner's Guide. ISBN 978-1-78216-032-8, Packt Publishing, 2013.
12. C. Haase and R. Guy, Filthy Rich Clients: Developing Animated and Graphical Effects for Desktop Java Applications, ISBN-10: 0132413930, Addison Wesley, 2007.
13. M. Scarpino, S. Holder, St. Ng, L. Mihalkovic, SWT_JFace in action, ISBN: 1-932394-27-3, Manning Publications Co, 2009.
14. J. Gradecki, N. Lesiecki, Mastering AspectJ: Aspect-Oriented Programming in Java. ISBN 0-471-43104-4, Wiley, 2003.

Una paralelización del método de Householder

Oswaldo Sposito, Gastón Procopio, Fabio Quintana, y Hugo Ryckeboer

Universidad Nacional de la Matanza
Departamento Ingeniería e Investigaciones Tecnológicas

Resumen La transformación de matrices a la forma de Hessemberg o triangular superior mediante el método de Householder es uno de los métodos más utilizados para obtener autovalores y autovectores de una matriz.

Cuando se intenta utilizarlo en un ambiente de multiprocesadores es conveniente que la matriz se pueda almacenar repartida en las memorias rápidas de los procesadores disponibles para contribuir al cálculo de las transformaciones $(I - \omega \omega^T) A$ que por razones de eficiencia se calculan como $A - \omega (\omega^T A)$. Distribuyendo a cada procesador los trozos de vector que se necesitan en cada operación (una contribución lineal), se alimenta una cantidad cuadrática de operaciones.

Cuando el tamaño de la matriz es tal que no cabe en la memoria se debe cargar y guardar sucesivos bloques de la misma en los procesadores disponibles, lo que baja su productividad por la cantidad cuadrática de ciclos de memoria lenta involucrados.

Se propone un replanteo del método de Householder que permite aprovechar la presencia de un bloque en la memoria para realizar múltiples tandas de actualización.

1. Introducción

Transformar los cálculos matriciales a esquemas de computación paralela se ha encarado hace tiempo [1,2]. Estos cambios se han enfocado de dos modos y en este orden: a) buscar operaciones del esquema de cálculo secuencial que se presten para un desarrollo paralelo. Los ciclos se prestan especialmente para estas transformaciones. b) Un segundo camino ha buscado un replanteo de los algoritmos de modo tal de poder expresar la meta por operaciones sobre grandes estructuras cada una de las cuales admite una paralelización ya conocida.

El primer enfoque se acerca mucho más a la arquitectura subyacente, para justificar y aumentar el ahorro de tiempo. El segundo enfoque se mantiene en un nivel más matemático ya que la paralelización la consigue recurriendo a subprogramas ya optimizados programados en todos sus detalles por los proveedores de ambientes y bibliotecas de cálculo paralelo como ser el MATLAB, ScaLAPACK, CUBLAS, OPENCL, etc. Se tarda meses en emular la eficiencia y flexibilidad de adaptación a una diversidad de estructuras que estos productos han conseguido.

Los métodos de alto nivel se pueden encuadrar bajo un denominador común: la división en bloques.

La transformación de esquemas de cálculo pensados para escalares introduce inevitablemente muchos productos entre bloques y sus derivados, que es una de las operaciones mejor implantadas en arquitecturas paralelas. A las divisiones entre escalares le suelen corresponder inversiones de bloques. Aunque los tamaños de los mismos pueden ser variables, los elementos de la diagonal principal ocupan bloques cuadrados. Rotella y Zambettakis [3] lograron adaptar el método de Householder a una descomposición en bloques.

Anderson, Ballard, Demmel, Keutzer [4] realizan un desarrollo orientado a matrices de muchas filas y pocas columnas, mostrando la posibilidad de lograr un uso intensivo de cálculo en cada procesador.

El método aquí propuesto está orientado a matrices cuadradas de gran tamaño, usuales en el método de recuperación de información LSI. Procede por etapas. La cantidad de etapas depende de la relación entre el tamaño de la matriz y la memoria total disponible entre todos los procesadores. Analíticamente se basa en una relectura de la transformación de Householder y tiene en vista que nunca deba ser cargado durante una etapa, en la memoria de registros de los procesadores, un mismo elemento de la matriz. Como la cantidad de operaciones es igual a la del método original salvo términos de menor orden, resulta que el esquema propuesto mantiene, a diferencia de una implantación directa de aquel, una muy alta relación entre cálculos y transferencias.

Teniendo en cuenta lo indicado, el método ocupa respecto de la clasificación inicial una posición intermedia.

2. Revisión de la transformación de Householder

La transformación de Householder permite transformar un vector en otro, con cierta limitación, por una reflexión respecto de un plano. Householder al plantearlo tenía en mente anular los elementos por debajo de una diagonal.

La transformación se puede describir con una matriz unitaria, la cual, por su número de condición unitaria, mantiene baja la propagación de los errores del cálculo.

La reflexión es una operación que no se limita a las necesidades del método tal como lo usa Householder.

En el planteo habitual se aprovecha para transformar un vector columna en un vector de ceros por debajo de la primera diagonal o subdiagonal.

Sin embargo esto no es una exigencia de la transformación sino de una estrategia para lograr en pocos pasos la deseada forma ya sea triangular superior o de Hessenberg.

Supongamos tener un vector u que se quisiera transformar por reflexión en un vector v . La única exigencia para que esto sea posible es que v tenga el mismo tamaño. De aquí surge una primera ecuación $\|u\|_2 = \|v\|_2$.

Hablar de tamaño con sentido geométrico no es más que tomar normas euclídeas. Cuando el método se aplica con la intención de conseguir ciertos valores en determinados elementos es necesario disponer de otro, un comodín, al que se le hará asumir el valor necesario para igualar módulos. Si los valores nuevos

superan a los viejos, el objetivo se hace imposible. En el uso que propuso Householder de esta transformación, que es la de anular elementos, ello es siempre posible. También lo será en esta propuesta. Tenemos aquí una primera ecuación: $\|u\|_2 = \|v\|_2$ que contiene una incógnita: el nuevo valor del comodín S , pudiendo elegir su signo.

Utilizando el método para triangular una matriz, el comodín pertenece a la diagonal y si es para llegar a la forma de Hessemberg el comodín pertenece a la subdiagonal.

No se pretende repetir íntegra la deducción que está reproducida en textos de cálculo numérico [5,6], sino llamar la atención a las características a tener en cuenta para comprender la modificación propuesta:

- El término comodín se sustituye por el largo del subvector armado por su antiguo valor y los términos que se quieren anular. Esta operación, una distancia euclídea, culmina con una raíz cuadrada cuyo signo se puede elegir según la conveniencia de la estabilidad numérica.
- El vector ω tiene componentes nulos en correspondencia a los elementos de u que no cambian.
- Los elementos que se anulan con la transformación ingresan tal cual en $u - v$.
- El comodín es el único que aporta una real diferencia. Como hay interés en tener un valor elevado se elige el signo de la raíz cuadrada como opuesto al signo del antiguo valor para que haya una suma de valores absolutos en lugar de una diferencia.
- Finalmente para tener ω es necesario escalar el vector diferencia de modo tal que su módulo sea 1 para satisfacer $\omega^\top \omega = 1$.

La aplicación de la transformación a la matriz A modifica todas sus columnas. Eligiendo el orden de los productos indicados se obtiene un esquema eficiente de cálculo:

$$(I - 2\omega\omega^\top)A = A - 2\omega(\omega^\top A) \quad . \quad (1)$$

Al producto $\omega^\top A$ en esta ecuación lo designaremos $\tilde{\omega}^\top$. Si el producto externo $2\omega\tilde{\omega}^\top$ se calcula por columnas no es necesario tener la totalidad de $\tilde{\omega}^\top$ calculado, sino que el cálculo de este vector puede avanzar conforme se lo necesita. Esto permite procesar matrices donde la cantidad de columnas es tal que no se puedan tener íntegras en los procesadores.

En el método propuesto el vector v tendrá algunos ceros más que u , consiguiendo en sucesivas etapas la anulación del triángulo inferior deseado.

Un desarrollo similar se efectúa en sucesivas etapas sobre las columnas siguientes.

El desarrollo propuesto está orientado a matrices cuyo tamaño impide tenerlas íntegras en las memorias rápidas de los procesadores disponibles.

Si se aplicara la fórmula 1 a estas matrices se cargaría bloques de A y un correspondiente subvector de ω^\top , con ello se puede computar la contribución del bloque al producto de la derecha y luego, esperar hasta tener el producto completado entre los diversos procesadores que contribuyen con las columnas del

bloque considerado. Luego cargar el subvector correspondiente al bloque y efectuar el producto externo con ω para modificar a A . A continuación almacenar el bloque en la memoria grande para poder procesar a otro bloque. Las transferencias que son lentas suman: 2 veces el bloque completo: carga y descarga, y 4 de tamaño lineal. La operaciones flotantes dos veces el tamaño del bloque, lo cual constituye un esquema de baja eficiencia.

3. Subdivisión de la matriz en franjas para su proceso

Se presupone una arquitectura en la cual cada procesador dispone de una memoria exclusiva que además de rápida es el lugar en el cual deben residir los operandos. La matriz dato y resultado residen en una memoria compartida la cual es también el vínculo para intercambiar mensajes. Finalmente se supone que la matriz dato es demasiada voluminosa para tenerla íntegra en la memorias rápidas. Un algoritmo para ser eficiente debe realizar gran número de operaciones con los elementos almacenados en su memoria rápida antes de sustituirlos por nuevos datos. Los mensajes a intercambiar son los vectores ω . La suma de todos ellos es una fracción del espacio que ocupa la matriz.

El algoritmo procesará por etapas y en cada etapa hará un nuevo reparto de la matriz A de $n \times n$ a transformar.

Para poder hacer un uso intensivo de partes de la matriz, se la supone dividida en franjas horizontales. En cada subetapa se tendrá en memoria la primera y una de las siguientes.

La primera con p filas, las siguientes con q , la última posiblemente con fracción de q . Los tamaños p y q , de ser conveniente se pueden redefinir en cada etapa, pero están influenciados por los recursos disponibles.

Los elementos de cada franja se almacenarán en sendos procesadores, si por insuficiente número de procesadores no se pudieran almacenar las franjas de una subetapa completas, bastaría con tener completas un grupo de columnas y una vez procesadas estas pasar al grupo siguiente.

Si el objetivo es triangular, la primera franja abarcará las filas 1 a p . Si es llegar a la forma de Hessenberg, como la primera fila no cambia, tomará de la 2 a la $p+1$. Cada etapa quedará dividida en subetapas. Una subetapa corresponde al proceso de una de las franjas de tamaño q .

Para unificar ambos objetivos de aquí en más la numeración de filas se referirá a lo almacenado.

3.1. Primera subetapa

La primera subetapa agrupará la franja de ancho p y la primera de tamaño q , ambas instaladas en las memorias de sendos microprocesadores, manteniendo en procesadores independientes los elementos de ambas franjas.

Procedemos a aplicar sobre las columnas 1 a p la transformación de Householder anulando hasta la fila $p+q$.

Las operaciones a efectuar serán descritas para una columna genérica $k \in 1 \dots p$. Se trata de anular en esa columna los elementos de las filas $k + 1 \dots p + q$. El cálculo a efectuar para determinar ω es idéntico al que se efectuaría si se tratara de una matriz de tamaño $p + q$.

El vector ω paralelo a $u - v$ tiene ceros donde esta diferencia es cero, los que se corresponden a los elementos que no sufren cambios que son las posiciones $1 \dots k - 1$ y $p + q + 1 \dots n$. Los elementos no nulos están en los lugares $k \dots p + q$. El elemento k es el que oficia de comodín y mantiene la igualdad de módulos.

El siguiente paso es analizar que sucede en el cálculo de $A - 2\omega(\omega^\top A)$.

La fila ω^\top debe construir un producto interno con las sucesivas columnas de A . Los elementos de la columna a los que corresponden ceros en el primer factor pueden ser ignorados. O sea, de cada columna de A sólo necesitamos acceso a las filas $k \dots p + q$ y estas pertenecen a las dos franjas almacenadas. El resultado de estos sucesivos productos internos es un vector fila $\tilde{\omega}^\top$, posiblemente completo a partir del lugar k . Como las columnas se procesan en orden, en las columnas anteriores a k ya hay ceros y se puede omitir un cálculo que daría cero.

Si la distribución de elementos de A por procesador se hizo por bloques, recibiendo una cantidad lineal de datos se organiza un cálculo cuadrático y una emisión lineal de resultados.

Más adelante se analiza las dimensiones ideales de tales bloques. En el último paso el vector columna ω multiplica al vector fila recién obtenido $\tilde{\omega}^\top$.

Cada elemento de A se actualiza con el esquema $a_{ij} = a_{ij} - 2\omega_i \tilde{\omega}_j$. En las filas donde ω_i se anula no hay nada que procesar, sufriendo cambios solamente las filas almacenadas en la memoria. Este cálculo exige distribuir a cada procesador los subvectores de ω y de $\tilde{\omega}$ que necesita para su actualización local. Trabajo que con una transferencia lineal organiza también un cálculo de volumen cuadrático.

Un mismo procesador dedicado a contener una submatriz de la franja realiza estos dos trabajos cuadráticos p veces lo que justifica el trabajo cuadrático de su carga y descarga.

3.2. Siguiendo subetapas

Se conserva el contenido de la primera franja y se sustituye la segunda franja con las siguientes q filas de la matriz A . Para no complicar las fórmulas se supone en lo que sigue que esta es la tercera subetapa, segunda después de la inicial.

El propósito es anular ahora los elementos del bloque determinado por las filas $p + 2q + 1 \dots p + 3q$ y columnas $1 \dots p$.

Nuevamente se procesa en orden las columnas. Para la columna k la situación y proceso es el siguiente:

- A las filas $1 \dots k - 1$ pertenecientes al triángulo superior no hay interés en cambiarlas
- La fila k volverá a cumplir el papel de comodín para igualar normas.
- Ya tiene ceros en las filas $k + 1 \dots p + 2q$ y no se quiere cambiarlas.
- Las filas $p + 2q + 1 \dots p + 3q$ deben anularse.
- Las filas $p + 3q + 1 \dots n$ no sufren cambios.

El vector ω de esta subetapa, construido por diferencia entre el antes y el después tiene elementos no nulos solamente en las posiciones k y $p + 2q + 1 \dots p + 3q$. Al efectuar el producto $\omega^T A$ solamente intervienen estas filas de A . La k , solitaria, ubicada en la primera franja; las otras en la segunda franja de la memoria que en este momento contiene la cuarta franja de A .

Ese producto genera un vector fila casi completo desde el lugar k y ceros a su izquierda, porque ya hay ceros en las columnas anteriores.

La modificación final de los elementos de A sigue el mismo esquema discutido para la primera subetapa.

Los elementos de la primera franja trabajan menos, un elemento para cada transformación de A , para cada k uno distinto con lo cual todos intervienen al menos una vez por subetapa.

Lo mismo sucede en el producto externo que actualización a A .

3.3. Etapas posteriores

Al finalizar la primera etapa la matriz se encuentra en el mismo estado en que hubiera estado después de p etapas del método de Householder. Las primeras p columnas y p filas de A no vuelven a intervenir y ya tienen sus valores definitivos. Luego hay que procesar la submatriz restante, cuadrada de tamaño $n - p$, lo que requiere un replanteo de los tamaños p y q .

4. Reflexiones sobre los tamaños

Hay varios tamaños por discutir: los parámetros p y q y la configuración de cada procesador. Estos elementos interactúan, pero conviene entender a cada uno por separado.

4.1. La elección de p y q

Hay que tener presente que múltiples características de la arquitectura de proceso disponible influyen en la elección de los parámetros. Las variables a concretar son discretas de modo tal que modelos continuos sólo sirven de orientación en el diseño.

La precisión se ve afectada si las franjas son demasiado pequeñas. Las publicaciones citadas omiten comentarios sobre la incidencia de los esquemas que proponen en la precisión. La experiencia numérica indica que la cantidad de franjas de tamaño q no debiera pasar de 10. Si se conserva los S^2 de cada etapa el valor del elemento a_{kk} al final de una etapa coincide con el valor del método original.

El volumen de cálculo sólo se ve afectado en un término menor. Para efectuar una transformación es necesario un cálculo preparatorio, obtener ω el cual depende de dos normas. Pero en el método estándar estas normas se calculan sobre vectores del orden de $n - k$ y en este reordenamiento, en la primera subetapa

con vectores de tamaño $p + q - k$, y en las siguientes subetapas con vectores de tamaño q con lo cual el trabajo total es comparable con el del método directo. Sólo está el trabajo extra de reinicialización de ciclos en cada subetapa, que frente a otras magnitudes parece ser despreciable.

Cada reducción genera una matriz ortogonal $H = I - \omega\omega^\top$, habiendo r reducciones por columna en lugar de una, su cantidad aumenta r veces. Conceptualmente, su producto es la matriz ortogonal que junto con la triangular superior podrían reconstruir la matriz original A . Hay que tener presente que los vectores ω que los engendran tienen $q + 1$ elementos no nulos, a excepción de la primera que tiene $p + q - k$, de modo tal que multiplicar por ellas cuesta r veces menos. que operar con las ω extensos del método original. Almacenar los ω correspondientes a un mismo valor de k para postergar estos productos insume $p + q - k$ para la primera subetapa y $q + 1$ para cada una de las $r - 1$ posteriores, su suma $p + q - k + (r - 1)(q + 1) = p + rq - k + r - 1 = n - k + r - 1$ contra $n - k$ de un ω grande del planteo original de Householder evidencia que insume sólo $r - 1$ lugares extras.

Cada transformación afecta a una fila de la primera franja y a todas las filas de la que esté ocupando la segunda franja presente. Cuando se procesa nuevamente para un mismo valor de k la segunda franja en memoria ya no es la misma. Si en total la matriz A fue descompuesta en una franja de ancho p y r de ancho q , la primera subetapa opera por cada valor de k sobre $p + q - k$ filas y las siguientes sobre $q + 1$, en total se procesan $p + q - k + (r - 1)(q + 1) = p + rq - k + r - 1 = n - k + r - 1$ contra $n - k$ del planteo original, o sea $r - 1$ filas extras. El aumento del proceso se corresponde con el aumento de almacenamiento y en ambos casos representa un término de menor orden.

Beneficio de un q pequeño. Mantener q bajo permite aumentar el valor de p , lo que implica menos etapas y por consiguiente menos lecturas en proporción al cálculo.

Inconveniente de un q demasiado pequeño. Pero, un q bajo implica que los procesadores que contienen bloques, submatrices de la matriz a factorizar, tienen pocas filas y consiguientemente más columnas. Una gran desproporción entre filas y columnas baja el rendimiento del procesador, expresado como cociente entre flops y transferencias.

4.2. Dimensionamiento óptimo de un bloque

Se puede tomar como indicador la relación entre volumen de las lecturas y volumen del cálculo, cuanto más bajo, mejor. Simplifica los análisis posteriores analizar primero las dimensiones óptimas de un bloque.

El cálculo se mide en flops, la operación típica del álgebra lineal, que abarca un producto y suma de reales.

Forma óptima de un bloque. Un bloque, o submatriz, se define por una cantidad a de filas y una cantidad b de columnas. Estas cantidades no son independientes sino que su producto es el tamaño B que ocupa el bloque en la

memoria, una fracción de la memoria rápida M de la cual dispone el procesador. Fijado B sabemos que esto dará lugar a un volumen $2B$ de transferencias.

Por cada valor de k se repite el mismo proceso: lectura de un vector columna de tamaño a y varias acciones de transferencia de vectores fila de tamaño b . Como esta cantidad dependerá de decisiones posteriores se parametrizará con t .

El volumen de transferencias que se hacen para cada valor de k es: $a + tB/a$, expresión que se minimiza con $a = \sqrt{tB}$ y de lo cual se deduce $b = \sqrt{B/t}$. La cantidad total de transferencias con formato óptimo resulta ser $2\sqrt{tB}$.

Primer diseño de q . Para minimizar la comunicación entre bloques es conveniente que las q filas ocupen un sólo bloque, lo que equivale a decir $a = q$. Cada procesador asignado a la franja se ocupa exclusivamente de una cantidad b de columnas.

Cada bloque (salvo los que contienen las primeras p columnas) interviene en las siguientes operaciones:

Carga del bloque Son ab lecturas.

Elaboración de $\tilde{\omega}^T$ El procesador debe recibir a elementos y efectúa ab flops con lo cual construye un vector de tamaño b . Éste se debe combinar con el cómputo de la fila k . Lo puede hacer de dos formas, envía su vector al procesador que tenga la fila k y recibe la suma terminada o recibe la contribución de la fila k , realiza la suma y la devuelve. En ambos casos son dos transferencias de tamaño b y puede emprender el siguiente cómputo.

Actualización de los elementos de A que están en el bloque La cantidad de operaciones es nuevamente ab

Grabación del bloque Son ab grabaciones.

Gracias al tratamiento propuesto la segunda y tercera operación se realizan p veces. La relación de a y b ya fue discutida y del análisis surge que la cantidad de transferencias t de dimensión b que se efectúa es 3. Lo que permite establecer:

$$\text{Relación de E/S a Flops} = \frac{2p\sqrt{3B} + 2B}{2pB} = 1/p + \sqrt{3/B} . \quad (2)$$

que señala la conveniencia de p y B grandes. El tamaño de B será una porción de la memoria M que la arquitectura provee a cada procesador, después de restar el espacio para las transferencias de vectores de tamaños a y b hacia la memoria compartida.

Compensación del desbalanceo de la carga Hasta aquí se ha ignorado a los procesadores que contienen los elementos de las primeras p filas. Estos computan de un modo similar pero en lugar de recibir un factor izquierdo de a elementos sólo reciben uno efectuando en total $2b$ flops y $3b + 1$ transferencias por cada k . Siendo sus transferencias comparables con la de un procesador de la franja inferior, no tiene retardo por ello, pero tiene capacidad de cálculo ociosa.

De aquí surge la conveniencia de estudiar si un sistema mixto no ofrece mejor productividad. Teniendo $p \gg q$ como el cálculo precedente ha sugerido tenemos muchos procesadores con baja ocupación. Sea f un coeficiente que exprese la relación que expresa $p/q + 1$. Por cada procesador activo al máximo hay $f - 1$ poco cargados.

Tratando de tener una carga homogénea se puede redistribuir la carga de modo tal de que se almacene en un procesador por cada fila de la segunda franja $f - 1$ de la primera. Para simplificar las cuentas y tener un esquema homogéneo de trabajo supondremos f una potencia de dos.

El tamaño del bloque se ve reducido a $B = M/f$. La cantidad de procesador que contienen las mismas columnas es f . El cálculo de $\tilde{\omega}^\top$ exige integrar en una única suma los aportes de cada uno lo que exige $2 \log_2 f$ intercambios de mensajes. Lo que sí se ha ahorrado es la comunicación por cada valor de k del aporte de la fila k ; alguno de los procesadores la tiene. Esto da $t = 1 + 2 \log_2 f$. El tamaño óptimo de a en cada procesador es $a = \sqrt{tB} = \sqrt{(1 + 2 \log_2 f)B/f}$. El volumen total de transferencias en cada procesador es $2p\sqrt{(1 + 2 \log_2 f)B/f} + 2B/f$ y el cálculo ocupa $2pB/f$.

En el tiempo en que trabaja un procesador para procesar un bloque de tamaño B/f trabajan f procesadores y completan el proceso de B el bloque que inicialmente hubiera estado solo en un procesador.

El inconveniente del reparto es el aumento de $q = fa$, y fija un valor de $p = (f - 1)q = f(f - 1)a$.

Para poder comparar las distintas soluciones, cuyas unidades de procesamiento no son iguales, se propone tomar el esfuerzo de procesar un bloque de tamaño B sometido a una reducción. Lo que da en transferencias $2\sqrt{(1 + 2 \log_2 f)B/f} + 2M/(f^2(f - 1))$.

El cuadro 1 muestra los valores de las variables en este esquema de carga distribuida. La parte inferior del cuadro estudia el comportamiento si se mantienen la primera y segunda franja separadas en dos grupos de procesadores. Se observa que resulta ventajoso el esquema de ubicación compartida ya que aprovecha con una carga más homogénea la capacidad de cálculo de todos los procesadores.

Por otra parte bajan las transferencias y cálculos que realiza cada procesador en paralelo al aumentar su número.

Para cantidades de procesadores que no sean las tabuladas hay que diseñarlo a medida porque los procesadores no intercambian la misma cantidad de mensajes.

También es legítimo apartarse ligeramente de los valores de a y b sugeridos ya que cerca del óptimo los cambios tienen poca incidencia.

Los procesadores asignados a las primeras p columnas tienen menos trabajo ya que no necesitan calcular los ceros que la teoría asegura que producen.

5. Experimento

Se ha supuesto que cada una de 4 unidades aritméticas tuviera 128K de memoria de trabajo y que, después de planear su uso se tuviera 100K disponibles lo que equivale a 12.500 números reales.

Con estos valores, aplicando las fórmulas de la segunda columna $p = 1644$, $q = 548$. Para que el proceso fuera por franjas plenas se construyó con azar una matriz simétrica con $N = 3288$. Se la trianguló en la forma habitual y siguiendo el esquema propuesto. Todo el cálculo se hizo en doble precisión. El error medio

Cuadro 1. Valores de los parámetros para diversas configuraciones

Procesadores asignados	unidad	2	4	8	16
Fracción de memoria franja 2		0,500	0,250	0,125	0,063
Transferencias de dimensión a		1	1	1	1
Transferencias de dimensión b		3	5	7	9
a óptimo	\sqrt{B}	1,225	1,118	0,935	0,750
b óptimo	\sqrt{B}	0,408	0,224	0,134	0,083
Ancho de la 2da franja	\sqrt{B}	2,449	4,472	7,483	12,000
Ancho de la 1ra franja	\sqrt{B}	2,449	13,416	52,383	180,000
Transferencias de vectores	\sqrt{B}	2,449	2,236	1,871	1,500
Transferencias carga y descarga	\sqrt{B}	0,408	0,037	0,005	0,001
Flops	B	1	0,5	0,25	0,125
Relación entre E/S y proceso	$\sqrt{1/B}$	2,858	4,547	7,502	12,006
Ancho de la 2da franja	\sqrt{B}	1,732	1,732	1,732	1,732
Ancho de la 1ra franja	\sqrt{B}	1,732	5,196	12,124	25,981
Transferencias de vectores	\sqrt{B}	3,464	3,464	3,464	3,464
Transferencias carga y descarga	\sqrt{B}	1,155	0,385	0,165	0,077
Flops	B	2	2	2	2
Relación entre E/S y proceso	$\sqrt{1/B}$	2,309	1,925	1,815	1,771

cuadrático fue $1,2740e-12$ con un máximo de $1,1642e-10$, ubicado en la primera fila.

Referencias

1. Dianne P. O’leary, G. W. Stewart Data-flow algorithms for parallel matrix computations Comm. ACM 28, 1985, 840-853.
2. Dianne P. O’leary, G. W. Stewart: Assignment and scheduling in parallel matrix factorizations, Linear Algebra Appl. 77, 1986, 275-299.
3. F. Rotella, F., Zambettakis Block Householder transformation for parallel QR factorisation. Appl. Math. Lett. Vol 12, Issue 4, May 1999, 29-34
4. Communication-Avoiding QR Decomposition for GPUs Michael Anderson, Grey Ballard, James Demmel, Kurt Keutzer Technical Report No. UCB/EECS-2010-131, University of California at Berkeley, October 8, 2010 <http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-131.html>
5. Gene H. Golub, Charles F. Van Loan Matrix Computations 4th Edition, Johns Hopkins University Press, 2013
6. William H. Press, Saul A. Toukolsky, William T. Vetterling, Brian P. Flannery Block Householder transformation for parallel QR factorisation. Appl. Math. Lett. Vol 12, Issue 4, May 1999, 29-34

Evaluación y Seguimiento Durante el Proceso de Enseñanza y Aprendizaje del Modelado UML en Entornos Colaborativos

Nicolás Battaglia, Carlos Neil, Darío Cardacci,
Marcelo De Vincenzi, Roxana Martinez
Facultad de Tecnología Informática - Universidad Abierta Interamericana
{nicolas.battaglia, carlos.neil, dario.cardacci,
medvincenzi, roxana.martinez}@uai.edu.ar

Resumen. Los avances de las nuevas TIC en materia de educación impulsaron fuertemente la colaboración en el proceso de enseñanza y aprendizaje. Los contextos académicos que brindan espacios necesarios para el desarrollo del aprendizaje centrado en el alumno, proveen las herramientas necesarias para que la interacción con docentes sea cada vez más fluida. Estos contextos ayudan a promover los espacios mixtos, para que el trabajo colaborativo aporte todo su potencial para mejorar el aprendizaje en aquellas áreas de conocimiento donde el uso de la tecnología cumpla un papel primordial, como es el caso de la Ingeniería de Software (IS). El objetivo de éste trabajo es diseñar un proceso de evaluación colaborativa en el marco de una herramienta CASE. Esto permite la integración del docente utilizando un conjunto de herramientas que brinden al alumno un panorama diferente al momento de obtener los resultados del proceso de modelado de sistemas dentro de la enseñanza de la IS

Palabras Clave: Trabajo Colaborativo, Ingeniería de Software Asistida por Computadora, Educación a Distancia

1 Introducción

La utilización de modelos para crear software utilizando lenguajes, técnicas y herramientas dentro de la IS (Ingeniería de Software) es una práctica común durante las etapas de análisis y diseño. Esto conlleva al desarrollo de un producto con calidad de Ingeniería para la resolución de un problema. Llegar a ser competentes en estas dos áreas requiere no sólo la identificación de problemas y el diseño de modelos sino también analizar, discutir y defender propuestas dentro de un equipo de personas involucradas [1].

Existe una gran variedad de herramientas que brinden soporte a lo planteado anteriormente. Estas herramientas denominadas CASE (Computer Aided Software Engineering) en su mayoría soportan algún lenguaje de Modelado estándar como UML (Universal Modelling Language). Permiten simplificar no sólo el entendimiento del problema planteado sino también mejorar la comunicación dentro de un equipo de trabajo y la correcta utilización de una metodología que permita llegar al objetivo con mayor facilidad [4][8].

El aprendizaje de la ingeniería de software debe estar acompañado con experiencia práctica y, principalmente, trabajo en equipo. Es muy común que los proyectos a nivel académico realizados por un grupo de alumnos persigan los siguientes objetivos: 1) Diseño, Validación, Implementación y Mantenimiento de sistemas Software 2) Entender procesos y modelos y 3) Obtener y mejorar el trabajo en equipo [4].

La integración de una herramienta CASE en un entorno académico brinda soporte necesario al proceso de Enseñanza y Aprendizaje en aquellas áreas del conocimiento dentro de la IS que requieren de medios externos para su ejecución y práctica profesional.

Basándonos en estudios anteriores podemos destacar que:

- La colaboración en la etapa de análisis y diseño de software es común, y muchas herramientas proponen espacios para que dos o más colaboradores puedan realizar el trabajo de forma simultánea, pero carecen de la integración de un docente como parte del grupo de trabajo, haciendo que carezcan de las herramientas necesarias para su enseñanza y aprendizaje [2].
- La dinámica de los contenidos en cursos de Ingeniería de Software proponen grandes desafíos para los docentes desde el punto de vista de planificación y ejecución, ya que esto puede provocar descuidos sobre los principios fundamentales del desarrollo de sistemas de software [6].

Estos puntos anteriormente mencionados, destacan la necesidad de 1) Contar con un espacio de participación docente dentro de una herramienta CASE, que permita integrar su potencial con un aparato de evaluación que se integre con los contextos colaborativos. 2) La necesidad de utilizar el concepto de ubicuidad para brindar un entorno de evaluación y seguimiento con el objetivo de optimizar la participación activa del docente dentro del proceso de evaluación.

1.1 Entornos Colaborativos y *Awareness*

Los contextos colaborativos se basan en aplicaciones para grupos de trabajo que comparten objetivos en común y surge de la unión de computadoras con tecnologías de comunicación. Son tecnologías diseñadas para trabajo en grupo y se pueden usar para comunicar, cooperar, coordinar y resolver problemas, entre otras aplicaciones [3].

Existen diferentes tipos de aplicaciones colaborativas, según el tiempo y lugar de ocurrencia. Pueden ser actividades en el mismo tiempo y lugar (Sincrónica Centralizada), mismo tiempo y distinto lugar (sincrónica distribuida), distinto tiempo y mismo lugar (Asincrónica Centralizada) y por último distinto tiempo y distinto lugar (Asincrónica Distribuida) [9].

Durante una actividad realizada dentro de un entorno colaborativo, surge información inherente a la misma, con el objetivo de entender actividades de otros y contextualizar la propia. Esta información, denominada *awareness*, describe el estado de la colaboración en relación a la presencia de usuarios, ubicación dentro del sistema, actividades realizadas, acciones, etc. [9]. También simplifica el trabajo dentro de un equipo de colaboración, ya que los usuarios van a tener una percepción más detallada de lo que está sucediendo dentro del entorno colaborativo [3].

1.2 Entornos Colaborativos de Enseñanza y Aprendizaje

Los métodos de aprendizaje colaborativo comparten la idea de que los estudiantes trabajan juntos para aprender y son responsables del aprendizaje de sus compañeros tanto como del suyo propio. Todo esto trae consigo una renovación en los roles asociados a profesores y alumnos, Las herramientas colaborativas deben enfatizar aspectos como el razonamiento y el autoaprendizaje y el aprendizaje colaborativo. El concepto de “aprendizaje colaborativo” describe una situación en la cual se espera que ocurran formas particulares de interacción, que conllevarán a mecanismos de aprendizaje, pero no hay una garantía total que estas condiciones ocurran efectivamente.

Para que existan garantías efectivas de aprendizaje, es necesario contar con interdependencia genuina entre los alumnos. También es necesario que el docente participe activamente en el proceso, ya que gran parte del aprendizaje se formaliza cuando se soluciona un problema. En este formato es el docente quien defina las condiciones iniciales del trabajo, plantee los objetivos académicos, defina los conocimientos mínimos que deben ser adquiridos durante el proceso de enseñanza, definir las tareas a realizar, definir los mecanismos de evaluación y monitorear el aprendizaje de los alumnos. Esta participación implica diseñar ambientes de aprendizaje donde existan muchas oportunidades para que los estudiantes puedan acceder a todo el contenido planteado de forma individualizada [5].

2 Modelado de Sistemas: Enseñanza y Aprendizaje Utilizando Herramientas CASE

En un mundo rico de información no es suficiente con que la información esté disponible en cualquier momento, lugar y forma. La información que llega al educando de forma ubicua debe ser pertinente con el grado de necesidad y acorde al contexto en que se encuentre el individuo y en nuestro caso el estudiante. Las actuales plataformas de formación mediadas por la tecnología están destinadas a facilitar las tareas del docente y del alumno pero carecen de herramientas que permitan la interacción entre ambos en tiempo real [7].

Según [1], en el modelo pedagógico centrado en el docente los estudiantes juegan un rol pasivo, en donde solo adquieren la experiencia en función de los conocimientos brindados por el docente. Es por esto que en los contextos colaborativos, en donde el alumno juega un rol predominante, es sumamente necesario que el docente pueda participar activamente dentro del proceso de enseñanza. Es importante que los alumnos no pierdan el foco y no solo puedan aprender de experiencias propias y de otros alumnos sino que también el docente pueda brindarles el conocimiento necesario.

Esto sucede también en aquellas áreas de conocimiento dentro de la IS que requiere el uso de herramientas CASE como es el modelado de sistemas.

En [2] se plantea la carencia de herramientas CASE con entornos colaborativos dedicados a la enseñanza de temas relacionados. Utilizar este tipo de herramientas presenta algunas limitaciones: Por un lado, no habrá espacio de interacción entre docentes y alumnos, esto significa tener que combinar distintos recursos para llevar a

cabo la realización de un proyecto. Por otro lado el alumno no dispondrá de la información o *awareness* inherente al proceso colaborativo de aprendizaje que se está llevando adelante. Esta información es importante ya que facilita el trabajo teniendo percepción de lo que está sucediendo con él y con los otros usuarios del ambiente. El docente tampoco contará con el entorno necesario para poder hacer un seguimiento y evaluación de los proyectos llevados a cabo por un alumno o grupo de alumnos. Esto implica tener que utilizar herramientas externas que permitan poder trazar las correcciones realizadas e inclusive, herramientas que coordinen la comunicación.

3 Diseño de Una Actividad Colaborativa

Es común encontrar procesos que requieran diferentes actividades con diferentes tipos de complejidad, a la vez que requieren personas con diferentes conocimientos para poder llevarlas adelante. Estos procesos se agrupan y consolidan para alcanzar un objetivo determinado, lo que conlleva a requerir que muchas personas participen en forma conjunta para cumplirlo [12].

En el apartado anterior, se plantea la necesidad de contar con un entorno colaborativo de interacción entre docentes y alumnos dentro de una herramienta CASE. El uso de este tipo de entornos tiene como objetivo favorecer al proceso de enseñanza y aprendizaje de aquellas áreas de conocimiento de la Ingeniería de Software que requieran el uso de este tipo de herramientas. Para esto es necesario describir la interacción y colaboración entre los diferentes roles (docente y alumno) que participen en las actividades de enseñanza y aprendizaje mencionadas.

Se identificó en [11] que existen varios problemas que hay que abordar antes de diseñar un proceso colaborativo. El modelado conceptual de grupos de trabajo tiene como característica la ausencia de modelos teóricos que permitan especificar las actividades de trabajo colaborativo y la dificultad de modelar aspectos de interacción y trabajo en grupo. Es por esto que se destaca la importancia de modelar aspecto de dinámica grupal relacionados a las interacciones sociales y a las responsabilidades inter-individuales.

4 Espacio de Interacción entre Docente y Alumno Destinado a la Evaluación de Modelos UML

El objetivo de este trabajo es describir la interacción en un proceso de evaluación colaborativa en un ejercicio de modelado UML en donde participa un grupo de alumnos en el desarrollo del mismo. El objetivo es obtener de forma clara y oportuna, información relacionada a los resultados funcionales del enunciado en base a correcciones realizadas por un docente. Estas acciones pueden enmarcarse dentro de una actividad distribuida en modo sincrónico o asincrónico. Esto significa que quienes participen podrán hacerlo en el mismo o diferentes momentos, pero siempre en diferentes lugares.

En [11] se proponen un conjunto de formalismos para la especificación adecuada de las tareas colaborativas que se deseen implementar. En base a eso, tomamos los

siguientes modelos que serán de utilidad para tal efecto: 1) Tabla CCD (Concepto, Categoría, Definición) con el objetivo de representar todos los conocimientos relativos al hecho. 2) Casos de interacción y diagramas de interacción grupal. 4) Diagramas de secuencia de dinámica grupal y 5) Diagrama de desarrollo de objetos conceptuales.

La evaluación se propone por medio de un espacio de colaboración asincrónico, utilizando como herramienta un foro, el cual denominaremos PC (Punto de Crítica). Durante el proceso de evaluación el docente puede considerar necesario abrir un PC utilizando un elemento propio de la herramienta, el cual pueda arrastrarse al diagrama en cuestión. Este elemento simboliza que en ese lugar o componente del diagrama, el docente tiene algo para corregir, informar o criticar el cual podrá ser accedido por los alumnos que colaboren con ese modelo.

En ese PC el docente va a abrir un hilo de un foro con el estado “en proceso” el cual va a poder ser comentado por cualquier participante durante el proceso de evaluación. Cuando algún alumno participante solicite revisión, el estado cambiará a “solicitud de revisión” y quedará a la espera de la misma para que luego el docente (en ese mismo momento, o en algún otro) pueda reevaluar ese punto y dar por terminado o reabrir el mismo según crea correspondiente para iniciar nuevamente el proceso.

Todos los mensajes realizados por los alumnos dentro de un PC podrán ser aceptados o rechazados por el docente. Siempre que exista un comentario rechazado, deberá estar acompañado con otro comentario justificando el rechazo y las acciones a tomar.

Para esto se propone utilizar los formalismos descritos anteriormente, con el objetivo de describir y documentar por medio de modelos todas aquellas actividades necesarias para que el alumno (o conjunto de alumnos) puedan disponer aquellas correcciones realizadas a lo largo del tiempo y poder hacer un seguimiento y análisis. Y por último será necesario conocer toda la información que contextualice lo que está sucediendo durante el proceso de evaluación.

4.1 Tabla CCD (Concepto-Categoría-Definición)

Esta tabla se utiliza para representar los conocimientos relativos al modelo de la dinámica grupal. Un concepto puede ser alguna de las siguientes categorías: actor, objeto o interacción. Los actores son sujetos de la dinámica, un objeto es el asunto que recibe la facultad de interactuar con actores y las interacciones definen los procesos que los actores acuerdan sobre los objetos.

Tabla 1 Tabla CCD del caso objeto de estudio "Evaluación de un modelo UML"

Concepto	Categoría	Definición
DOCENTE	ACTOR	Profesional que cumpla el rol de docente y sea el responsable de guiar el aprendizaje y posea los conocimientos técnicos necesarios de análisis de sistemas y modelado UML.

ALUMNO	ACTOR	Estudiante cursando una materia de análisis de sistemas
PUNTO DE CRÍTICA	OBJETO	Componente de un modelo que refiere a una situación a analizar dentro de un modelo UML. Este componente se simbolizará con un ícono predefinido y contendrá un foro como herramienta de colaboración.
MODELO UML	OBJETO	Representa el modelo UML utilizado como herramienta de análisis funcional.
MODELA	INTERACCIÓN	El actor que representa un alumno dentro de un grupo de alumnos realiza una actividad de modelado utilizando la herramienta CASE provista.
SOLICITA REVISION	INTERACCION	El actor que representa un alumno dentro de un grupo de alumnos realiza una solicitud de revisión a quien sea el docente responsable de ese grupo, participante de la actividad de modelado.
SOLICITA CORRECCION	INTERACCIÓN	El docente utiliza el componente "Punto de crítica" para abrir un hilo de un foro que represente la solicitud de corrección.
REVISAR	INTERACCION	El docente realiza una revisión del modelo sin realizar ninguna.
CORRIGE	INTERACCION	Un alumno realiza las correcciones correspondientes en base al punto de crítica iniciado por el docente. Luego podrá solicitar nuevamente revisión.
SOLICITA APROBACION	INTERACCION	El alumno envía correcciones realizadas al docente para que sean aprobadas.
APRUEBA CORRECCIONES	INTERACCION	El docente da por finalizada la evaluación realizada y la da por aprobada.

ENVIA APROBACIÓN	INTERACCION	El docente envía la aprobación del modelo.
REALIZA COMENTARIO	INTERACCIÓN	Cualquier involucrado en el proceso de evaluación podrá realizar comentarios sobre el foro abierto por el docente. Estos comentarios deberán estar relacionados con el tema a tratar.
ACEPTA COMENTARIO	INTERACCION	El docente acepta un comentario realizado por un alumno. La aceptación implica una respuesta.
RECHAZA DE COMENTARIO	INTERACCIÓN	El docente rechaza los comentarios realizados por los alumnos. El rechazo implica una respuesta a través de un comentario.

4.2 Casos de Interacción y Diagramas de Interacción Grupal

Estos dos diagramas son dos formalismos para modelar interacción. Los casos de interacción modelan la interacción entre dos actores y los diagramas de interacción grupal muestra de forma integrada las interacciones de todos los actores considerados en el proceso de modelado. Estos diagramas se apoyan en los diagramas de casos de uso. El escenario planteado posee solo dos actores, por lo tanto solo se van a utilizar los diagramas de casos de interacción que se muestran a continuación.

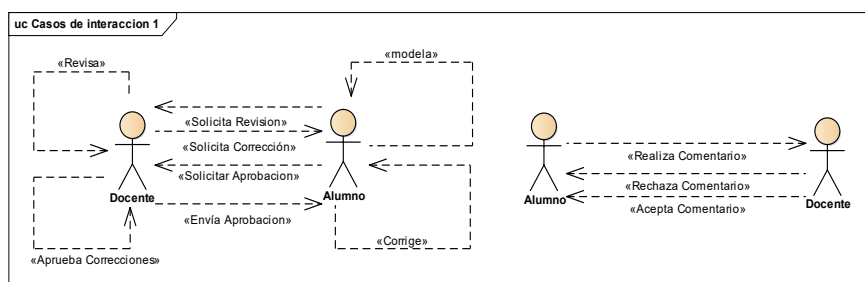


Ilustración 1 Casos de Interacción para "Evaluación colaborativa de modelos UML"

4.3 Diagramas de Secuencia de Dinámica Grupal

Este diagrama expresa la dinámica grupal dentro de un contexto temporal.

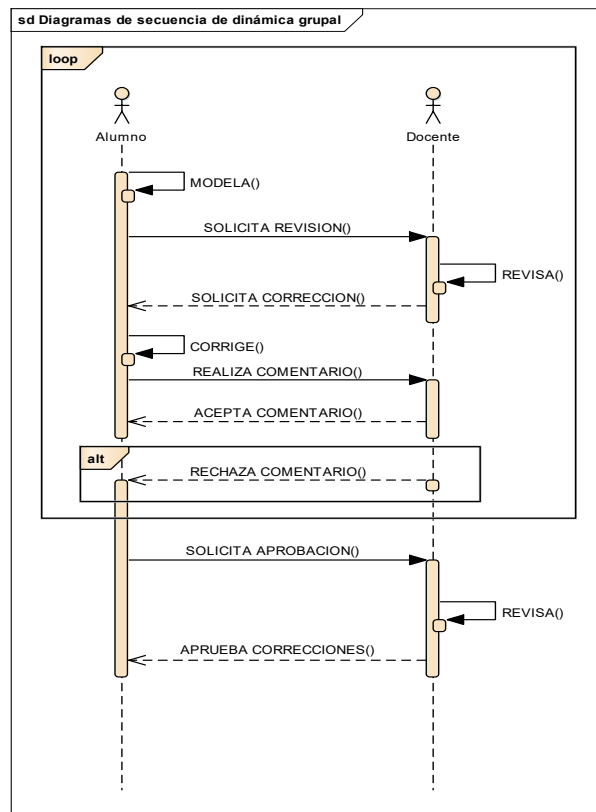


Ilustración 2 Diagrama de secuencia de dinámica grupal para "Evaluación Colaborativa de modelos UML"

4.4 Diagrama de Desarrollo de Objetos Conceptuales

Este diagrama representa el objeto conceptual que grupo de trabajo debe desarrollar. El espacio colaborativo debe satisfacer las actividades que soporten las distintas transformaciones que marquen la evolución de dicho objeto.

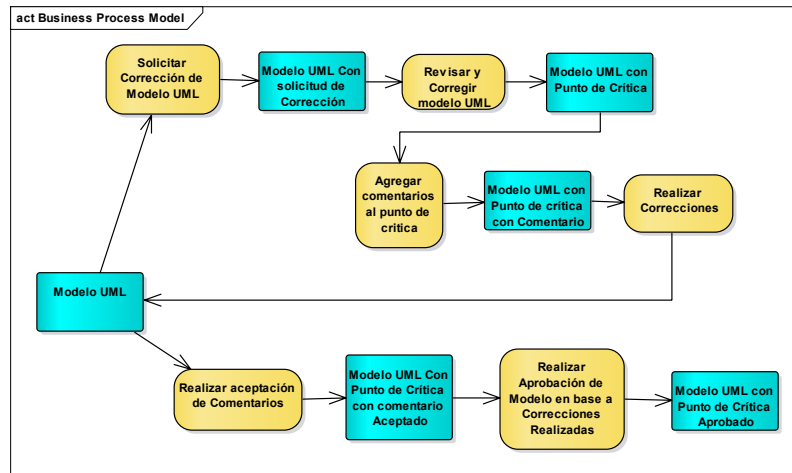


Ilustración 3 Representación de Objetos Conceptuales y sus Transformaciones

5 Conclusiones y Trabajos Futuros

En trabajos anteriores se identificó como debilidad de las herramientas CASE actuales, la carencia de integración con entornos de enseñanza y aprendizaje [1] [10]. Esto se opone al crecimiento acelerado de las TIC y su aplicación constante en ese tipo de entornos, lo cuáles están cada vez más integrados con el aula generando entornos mixtos que permitan aprovechar al máximo todo el potencial de la tecnología aplicada a la educación. Todo el crecimiento de las nuevas Tecnologías de Comunicación aporta las herramientas necesarias para utilizar entornos virtuales y aprovechar al máximo las bondades de la enseñanza y aprendizaje colaborativo.

Si trasladamos todo lo anteriormente mencionado al área de la Ingeniería de Software, se destaca la necesidad de contar con entornos de enseñanza y aprendizaje integrados con herramientas CASE. Esta integración debe considerar aspectos colaborativos entre docentes y alumnos, en particular, los que derivan de la evaluación y seguimiento del aprendizaje y aplicación de modelos para el análisis y diseño de software. Identificar y describir esta actividad no es un proceso sencillo. Contar con un método de especificación y el uso de modelos para formalizar los roles, las interacciones y las transformaciones de forma colaborativa, representa acercarse a un proceso de calidad.

Como futuros trabajos se prevé por un lado continuar con el desarrollo de mecanismos colaborativos evaluación. Para esto ampliaremos el uso de Rubricas y su integración con entornos colaborativos.

Por otro lado, avanzar sobre el diseño y especificación de todos los procesos colaborativos involucrados en el diseño de diagramas UML, identificar y caracterizar toda la información o awareness necesaria y formalizar todo por medio de modelos.

Referencias

1. Arteaga, J. M., Rodríguez, F. Á., González, L. A. G., & Delgado, F. J. P. (2005). Modelo para el aprendizaje colaborativo del análisis y diseño orientados a objetos. *Apertura impresa*.
2. Battaglia N., Neil C., De Vincenzi M., Martínez M. (2016). UAICase: Integración de un Entorno Académico con una Herramienta CASE en una Plataforma Virtual Colaborativa. In *XI Congreso de Tecnología en Educación y Educación en Tecnología*.
3. Bibbó, L. (2009). Modelado de sistemas Colaborativo (Tesis de Maestría). Universidad Nacional de La Plata, Facultad de Informática. Buenos Aires. Argentina.
4. Bouillon, P., Krinke, J., & Lukosch, S. (2005, April). Software engineering projects in distant teaching. In *Software Engineering Education & Training, 18th Conference on* (pp. 147-154). IEEE.
5. Collazos, C., Guerrero, L., & Vergara, A. (2001, November). Aprendizaje Colaborativo: Un cambio en el rol del profesor. In *Proceedings of the 3rd Workshop on Education on Computing, Punta Arenas, Chile*.
6. Daniele, M., Uva, M., Martelloto, P., & Picco, G. (2010). Aplicación de herramientas CASE a la enseñanza de Ingeniería de Software: Gestión de la Configuración de Software y Testing Funcional. In *V Congreso de Tecnología en Educación y Educación en Tecnología*.
7. Filippi, J. L., Lafuente, G. J., & Bertone, R. A. (2010). Diseño de un ambiente de aprendizaje colaborativo. In *V Congreso de Tecnología en Educación y Educación en Tecnología*.
8. Hernández, L. R. B. (2016). Extensión de la herramienta Visual Paradigm for UML para la evaluación y corrección de Diagramas de Casos de Uso. *Serie Científica-Universidad de las Ciencias Informáticas*, 9(4).
9. Herrera, A., Rodríguez, D., & García Martínez, R. (2013). Taxonomía de mecanismos de awareness. In XVIII Congreso Argentino de Ciencias de la Computación.
10. Neil C., De Vincenzi M., Battaglia N., Martínez M. (2016). Herramientas Colaborativas Multiplataforma en la enseñanza de la Ingeniería de Software. In *XVIII Workshop de Investigadores en Ciencias de la Computación*.
11. Rodríguez, D., & García Martínez, R. (2012). Modelado de interacciones en espacios virtuales dedicados a trabajo colaborativo. In *XVIII Congreso Argentino de Ciencias de la Computación*.
12. Solano, A., & Collazos, C. (2013). Modelo para el diseño de actividades colaborativas desde un enfoque práctico. *Revista Universitaria RUTIC*, 1(2)

Herramientas lúdicas como apoyo a la enseñanza de la programación

Gladys Dapozo; Cristina Greiner, Raquel Petris

Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura
Universidad Nacional del Nordeste, Av.Libertad 5450, 3400, Corrientes, Corrientes, Argentina
{gndapozo,cgreiner, rpetris}@exa.unne.edu.ar

Resumen. En este trabajo se presenta el resultado de una experiencia sobre la implementación de una modalidad de aprestamiento para los alumnos ingresantes a la carrera Licenciatura en Sistemas de Información de la Universidad Nacional del Nordeste. Esta modalidad incorporó al inicio del dictado de la asignatura Algoritmos y Estructuras de Datos I, una serie de actividades basadas en las herramientas Lighthbot y Scratch, con el objetivo de incorporar conceptos básicos de programación mediante herramientas lúdicas que resultan más atractivas para los alumnos. Los resultados indican que estas actividades contribuyeron positivamente a la comprensión de los conceptos básicos de la programación estructurada.

Palabras clave: Herramientas lúdicas de programación, enseñanza de la programación, vocaciones TIC.

1. Introducción

Los resultados de la reciente encuesta del Observatorio Permanente de la Industria del Software y Servicios Informáticos (OPSSI) perteneciente a la Cámara de la Industria Argentina del Software (CESSI), arrojaron para el 2015 un aumento promedio del empleo del 5,6%, un crecimiento en las ventas del 29,8% en pesos, y un aumento del 12,7% de las exportaciones en dólares estadounidenses (USD) respecto del 2014 [1]. Es evidente el crecimiento del sector pero una dificultad manifiesta es la escasez de recursos humanos formados para esta actividad, situación que se ve agravada por la disminución del interés de los jóvenes por realizar carreras de Informática. *“La gran transformación digital de estos años tanto de la economía como de la sociedad mantienen una alta y sostenida demanda mundial y local de software y de servicios informáticos. Debemos seguir encontrando soluciones a la escasez de talentos, que continúa siendo uno de los mayores desafíos a resolver para sostener el crecimiento de esta industria”*, según Aníbal Carmona, presidente de la CESSI.

En Argentina, la Fundación Sadosky¹ trabaja en la articulación entre el sistema científico-tecnológico y la estructura productiva en el ámbito de la informática y las telecomunicaciones, generando con ello un impacto positivo en la sociedad y en las posibilidades de desarrollo del país. Uno de sus objetivos es incorporar el estudio de programación en las escuelas argentinas y fomentar el incremento de la matrícula en carreras relacionadas con las Tecnologías de la Información y las Comunicaciones (TIC). Para ello lleva adelante diversos programas, entre estos, Vocaciones TIC y Program.ar.

¹<http://www.fundacionsadosky.org.ar/>

El programa Vocaciones en TIC tiene como propósito despertar interés en los jóvenes para estudiar carreras vinculadas con las TIC, en forma más amena y desestructurada, mediante la programación de juegos y animaciones. Para ello, se realizan visitas a las escuelas del nivel medio para realizar talleres de programación, basados en la herramienta Alice (Rebeca en español), orientados a la elaboración de juegos y animaciones, de manera sencilla y amigable.

En tanto, la iniciativa Program.AR² tiene como objetivo llevar la enseñanza y el aprendizaje de las Ciencias de la Computación a la escuela argentina. Esta incluye múltiples aspectos relacionados con la difusión y popularización de la disciplina, la generación de contenidos escolares y la formación docente, entre otros. La propuesta es desarrollada de manera conjunta por la Fundación Sadosky, el portal educ.ar y el Programa Conectar Igualdad.

La Facultad de Ciencias Exactas y Naturales y Agrimensura de la Universidad Nacional del Nordeste (FaCENA-UNNE) ha adherido a estos programas, mediante convenios específicos con la Fundación Sadosky. En este marco, un equipo de docentes y estudiantes de la Licenciatura en Sistemas de Información, ha llevado adelante el proyecto de Visitas a las escuelas desde el 2013 al 2016 y la capacitación a los docentes, como resultado de las convocatorias 2015 y 2016. En estas actividades se recolectaron datos para generar información que aporte a la problemática de las vocaciones TIC y la promoción e incorporación del pensamiento computacional en los niveles educativos no universitarios. Los resultados de estas experiencias se publicaron en [2], [3], [4].

Por otra parte, las universidades argentinas enfrentan hoy problemáticas preocupantes con respecto a la retención y la permanencia de los alumnos ingresantes a las carreras científico-tecnológicas en general, y a las vinculadas con la disciplina Informática en particular. Se sabe también que la deserción no sólo responde a cuestiones socioeconómicas. Entran en juego la diversidad cultural, problemáticas de orden personal, la carencia de conocimientos, la falta de estrategias cognitivas y metacognitivas de abordaje de los mismos, la falta de adecuación a los tiempos de estudio universitarios. También es necesario tener en cuenta los problemas generados por las diferencias respecto de aprendizajes y conocimientos previos [5].

La Licenciatura en Sistemas de Información de la UNNE no es ajena a las situaciones mencionadas. Un desgranamiento y abandono del 50% de los alumnos en el primer año de la carrera dan cuenta de la problemática [2]. Para revertir esta situación, se plantea como objetivo favorecer una mejor competencia para la programación, aplicando como estrategia el potencial de conocimiento y creatividad a través del juego, utilizando las herramientas Lightbot y Scratch.

2. Metodología

Desde el año 2013, la FaCENA-UNNE implementó un Programa de Seguimiento a los Ingresantes (Resolución N°700/13 CD) en articulación con los espacios de Tutorías y el Gabinete Psicopedagógico con el propósito de acompañar al ingresante, mejorando sus posibilidades de adecuarse a la universidad y permanecer en la carrera elegida.

En este marco se propuso, a partir del año 2014, el inicio anticipado de las actividades académicas para los alumnos ingresantes, con el objetivo de desarrollar

²<http://program.ar>

actividades de aprestamiento por un periodo de 3 semanas previo al dictado de los contenidos propios de cada asignatura del primer cuatrimestre del primer año de cada una de las carreras. El equipo docente de cada cátedra trabajó conceptos y herramientas para articular con los contenidos universitarios, entendiendo como contenido en un sentido amplio, es decir no sólo lo referido a lo conceptual sino también a lo procedimental y actitudinal, generalmente transversales y útiles para el desenvolvimiento en las diferentes actividades características de un alumno universitario, para cumplir con los siguientes objetivos:

- Acompañar a los estudiantes a la adaptación progresiva a la dinámica universitaria
- Generar espacios de enseñanza con un aumento gradual de las exigencias de la apropiación conceptual
- Favorecer la reactivación de conocimientos previos, en un trayecto de complejidad creciente y de progresivo acercamiento a los desafíos cognitivos exigidos en la educación superior
- Promover y orientar la revisión de elecciones vocacionales.

Para cumplir con este propósito en la carrera Licenciatura en Sistemas de Información se definió el período de aprestamiento en la primera asignatura disciplinar orientada a la iniciación de la programación, Algoritmos y Estructuras de Datos I. En este lapso de tiempo se desarrollaron actividades orientadas a estimular el pensamiento computacional mediante la incorporación de herramientas lúdicas de programación. El objetivo es motivar a los alumnos y ofrecer una base de ideas y técnicas que permitan apoyar una mejor comprensión de los conceptos básicos de programación que forman parte de los contenidos de la asignatura.

Las actividades previstas se tomaron del curso Introducción a la Programación y su Didáctica, que la Fundación Sadosky propone para la formación de docentes a través de la iniciativa Program.Ar [6].

El periodo de aprestamiento se desarrolló desde el 22 de febrero al 11 de marzo de 2.016, mediante 2 clases semanales de 2 hs. En la propuesta formativa se insistió en los conceptos de “abstracción”, referido a pensar una estrategia antes de intentar la resolución de los ejercicios, la “descomposición del problema en partes”, traducida a la creación de bloques o procedimientos en la solución diseñada y en la “legibilidad” de la solución, mediante la definición de nombres representativos de los procedimientos creados.

Finalizado el periodo de aprestamiento se realizó una actividad de evaluación que consistía en resolver un problema que requería utilizar los conceptos de secuencia, alternativas y repetición condicional, utilizando la herramienta Scracht,

A continuación, la asignatura desarrolló sus contenidos específicos, introducción a la programación estructurada utilizando el lenguaje de programación C para el desarrollo de los laboratorios. Durante el dictado, al introducir los distintos conceptos (estructuras de control, funciones, parámetros, recorrido en arreglos), se buscó relacionar con las actividades de Scracht en las que se utilizaban técnicas similares.

Al finalizar el dictado, se realizó una encuesta a los alumnos, con el propósito de determinar si el conocimiento adquirido en el periodo de aprestamiento, les resultó útil para una mejor comprensión de los conceptos y para adquirir las habilidades necesarias para el logro de los objetivos de aprendizaje de la asignatura.

3. Resultados

A continuación, se describen los resultados de la encuesta realizada utilizando el formulario en Google Docs: <https://docs.google.com/forms/d/1uMPDsyPHkTFhrz-NhiROJiZAE5YWU2hZtbkYVWjlBUE/edit?ts=579a2957>

Consultados los alumnos respecto a su apreciación sobre las herramientas utilizadas, Lighthbot y Scratch, se observa una leve preferencia hacia la última, de la cual el 66% piensa que es *lúdica, didáctica y entretenida*. En ambos casos, sólo el 5% considera que es *Compleja* (Ver Figura 1).

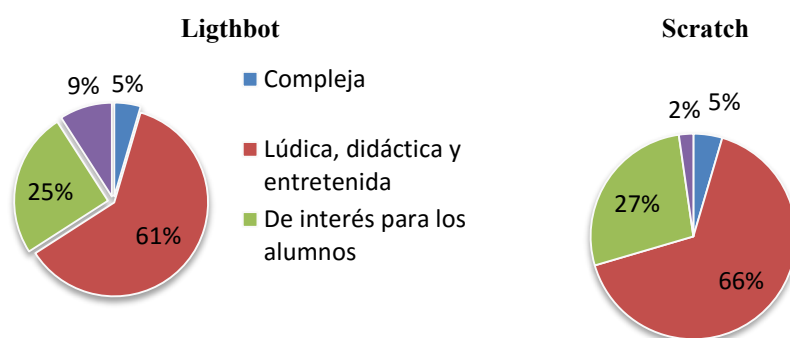


Figura 1: Apreciación sobre las herramientas considerando diversos aspectos

En cuanto a la dificultad para el aprendizaje y uso de las herramientas, si bien Lighthbot resulta levemente más fácil de aprender (61%) se observa que Scratch no representa mayor dificultad (el 39% aprende y usa con escasa dificultad), tal como se muestra en la Figura 2.

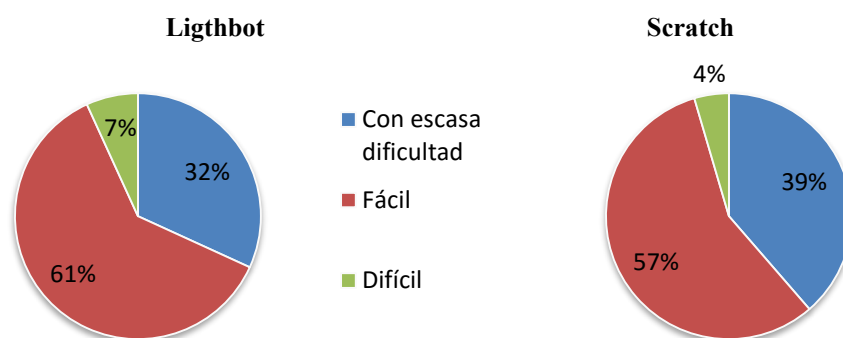


Figura 2: Dificultad en el aprendizaje y uso de las herramientas

De las respuestas en relación a las herramientas, se observa en general una alta aceptación hacia las mismas, y en ambos casos resulta difícil sólo para un 7% y un 4%.

Se preguntó también a los alumnos en qué grado las actividades realizadas durante el aprestamiento contribuyeron a incorporar los aspectos claves que conforman el método para diseñar soluciones, tales como:

- Pensar una estrategia antes de resolver los ejercicios (Abstracción)
- Descomponer el problema en partes (Modularización)
- Asignar nombres representativos a los módulos (Legibilidad)

En todos los casos los alumnos señalaron su satisfacción con la realización de estas actividades. Un 68% opina que contribuyeron significativamente a comprender el concepto de *abstracción*, un 75% a la *modularización* y 71% a la *legibilidad*, como puede observarse en las figuras 3, 4 y 5.

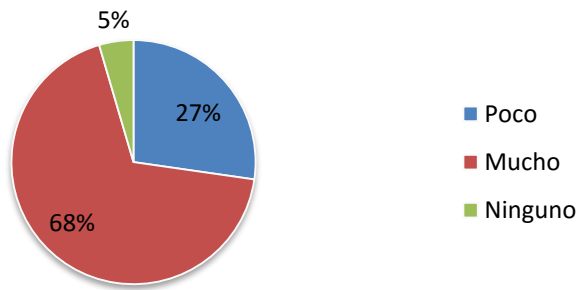


Figura 3: Contribución en la abstracción

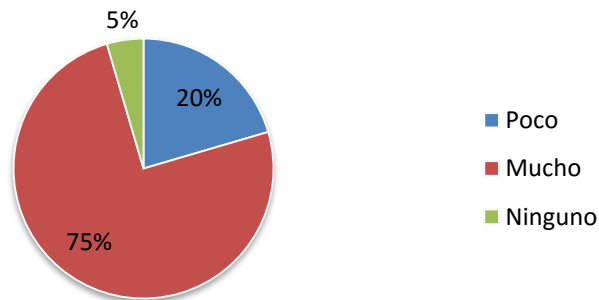


Figura 4: Contribución en la modularización

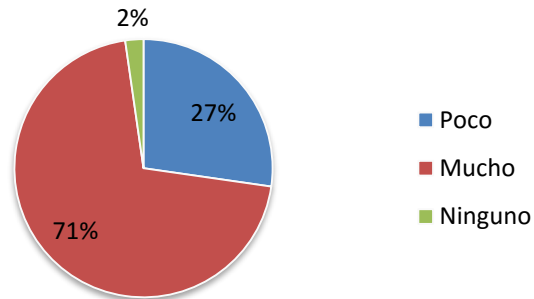


Figura 5: Contribución en la legibilidad

También se preguntó a los alumnos en qué grado las actividades del aprestamiento lo ayudaron a entender los conceptos básicos relacionados con la programación estructurada, tales como:

- Secuencia
- Alternativa (Selección)
- Iteración (Repetición)
- Modularización (Uso de funciones)
- Parámetros (Generalización)

En la mayoría de los casos, los alumnos valoraron las actividades de aprestamiento realizadas para mejorar la comprensión de los mencionados conceptos. En un alto grado en el caso de la *repetición* (91%), y en menor grado en el tema *parametrización*, en el que el 50% de los alumnos manifestó que le resultó poco útil. Esto puede deberse al escaso tiempo dedicado a este tema, para la complejidad del mismo.

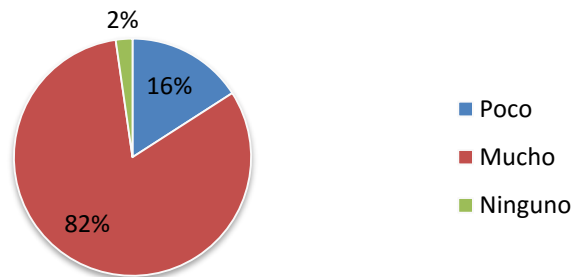


Figura 6: Comprensión de la estructura secuencial

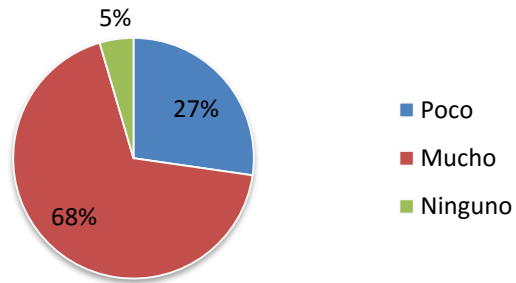


Figura 7: Comprensión de la estructura selectiva

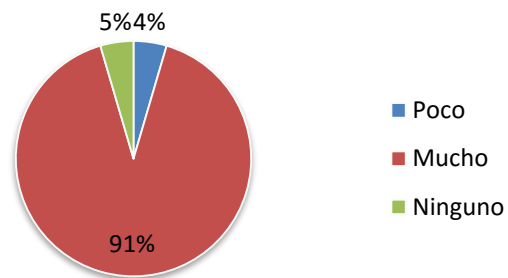


Figura 8: Comprensión de la estructura repetitiva

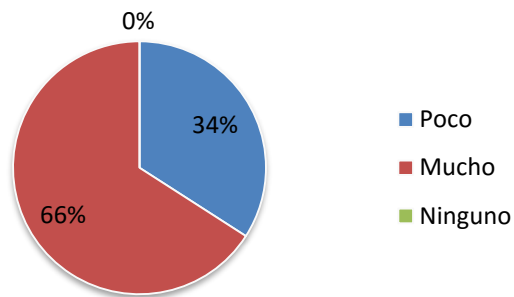


Figura 9: Comprensión del uso de funciones

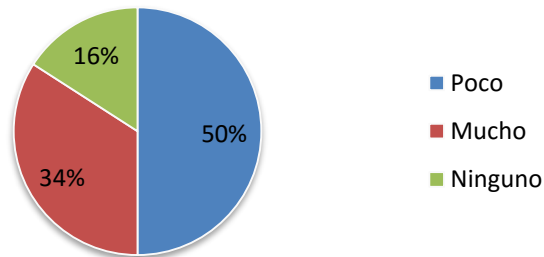


Figura 10: Comprensión del uso de parámetros

Finalmente, se preguntó a los alumnos si consideraban que sin las actividades de aprestamiento hubieran logrado resolver los problemas planteados en los prácticos de la asignatura. El 70% respondió que le hubiera costado más, y el resto señaló que hubiera llegado al mismo resultado.

4. Conclusiones

De los resultados obtenidos, se infiere que la experiencia resultó positiva para la comprensión de los conceptos propios de la programación en el nivel introductorio universitario, facilitando la incorporación de los conceptos y técnicas formales del área Algoritmos y Estructuras de Datos.

De los aspectos señalados por los alumnos surge la necesidad de asegurar que los mismos cuenten con el equipamiento necesario para realizar la práctica.

A futuro, se pretende continuar con esta propuesta, realizando los ajustes que surgen de los resultados de la encuesta y del feedback de los docentes y alumnos.

5. Referencias

- [1] OPPSI. Reporte anual del sector de software y servicios informáticos de la República Argentina. Año 2015. Disponible en: <http://www.cessi.org.ar/descarga-institucionales-2007/documento2-130347cd83ae771a9f3db3da5407269a>
- [2] Gladys N. Dapozo, Cristina Greiner, Gabriel Osmar Pedrozo Petrazzin, Jorge Chiapello. "Investigación para fortalecer actividades de promoción y retención de alumnos en carreras de Informática". Anales del XX Congreso Argentino de Ciencias de la Computación. 1a ed. San Justo. Buenos Aires. Universidad Nacional de La Matanza, 2014. ISBN 978-987-3806-05-6. 23 de octubre de 2014.
- [3] Gladys N. Dapozo, Cristina Greiner, Gabriel Osmar Pedrozo Petrazzin, Jorge Chiapello. Vocaciones TIC. ¿Qué tienen en común los alumnos del nivel medio que tienen interés por carreras de Informática?". IX Congreso de Tecnología en Educación y Educación en Tecnología (TE&ET 2014). ISBN 978-987-24611-1-9. Pag. 128 a 137. Universidad Nacional de Chilecito. Chilecito. La Rioja. 12 y 13 de junio de 2014.
- [4] Gladys Dapozo, Raquel Petris, Cristina Greiner, María Cecilia Espíndola, Ana María Company, Mariano López. "Capacitación en programación para incorporar el pensamiento computacional en las escuelas". X Congreso de Tecnología en Educación y Educación en Tecnología (TE&ET 2015). ISBN 978-987-3977-30-5. Universidad de Morón. Buenos Aires. Junio de 2015.

- [5] Marcela F. López, Paola del Olmo, Carina Reyes, Eduardo F. Fernández. La programación lúdica como estrategia de articulación entre niveles”. VIII Congreso de Tecnología en Educación y Educación en Tecnología (TEyET 2013). Universidad Nacional de Santiago del Estero (Santiago del Estero, Argentina)
- [6] Iniciativa Program.Ar.”La programación y su didáctica”. Disponible en: <http://program.ar/la-programacion-y-su-didactica/>

Scrum como Herramienta Metodológica en el Entrenamiento Cooperativo de la Programación: Proceso Evaluativo

Lorena Del Moral Sachetti y Ángel Rubén Barberis

Universidad Nacional de Salta. Facultad de Ciencias Exactas
Departamento de Informática

{barberis, lorena_dms}@cidia.unsa.edu.ar

Resumen. La Enseñanza de la Programación de Lenguajes es y ha sido una tarea difícil. La falta de habilidades y motivación en la programación son variables que causan una alta tasa de deserción y una baja calidad cognitiva en su aprendizaje. Este hecho se refleja en asignaturas, que necesitan como sustento básico, los conceptos y habilidades sobre la Programación, para el anclaje pedagógico de nuevos contenidos y capacidades. Como posible solución a los problemas de aprendizajes observados, en el año 2014 se implementó una propuesta pedagógica basada en el Entrenamiento Cooperativo de la Programación, en las asignaturas Programación Numérica y Cálculo Numérico de la carrera Licenciatura en Análisis de Sistemas de la UNSa.

El presente trabajo expone los resultados del diseño e implementación del proceso evaluativo de los aprendizajes de la metodología propuesta, en las que se explican las actividades realizadas y las bases teóricas sobre las que se sustentan.

Palabras Claves: Programación, Proceso Evaluativo, Scrum, Metodología.

1 Introducción

En las carreras informáticas, la programación de computadoras es una asignatura básica y fundamental; ya que la calidad de las competencias alcanzadas determina la continuidad del estudiante en la carrera y en el desempeño profesional del futuro egresado. Es evidente que el proceso educativo de la programación resulta altamente preocupante para los docentes.

Varios investigadores como [1, 2] demuestran que los estudiantes no logran aprehender los conceptos de programación enseñados, siendo este uno de los principales factores de la alta tasa de deserción en los primeros años de la carrera y el bajo rendimiento en las asignaturas de los años superiores. Por lo que, resulta necesario, investigar herramientas y/o metodologías que contribuyan a disminuir las dificultades del proceso en pos de mejorar la calidad educativa del mismo. En este contexto, los

autores pusieron en ejecución una metodología que concibe las prácticas de la programación como un entrenamiento denominado “*Scrum como Herramienta Metodológica en el Entrenamiento Cooperativo de la Programación*” [3]. Esta propuesta que lleva implementándose desde el año 2014 ha logrado reducir la tasa de deserción al 5% y aumentar del rendimiento académico al 83.6%.

A fin de comprobar en qué medida se han logrado los objetivos educativos expuestos en la propuesta metodológica, resulta de vital importancia la realización de un proceso evaluativo, reflexionando sobre las prácticas docentes, detectando obstáculos y dificultades, y de ese modo, tomar las decisiones necesarias para superarlos, priorizando siempre las necesidades pedagógicas de los alumnos, de modo de ayudarlos, guiarlos y encauzarlos hacia la apropiación correcta de los conceptos de la programación y su entrenamiento.

En este sentido, se entiende *la evaluación educativa* como un proceso continuo, programado, ordenado y sistemático de investigación [4], que valiéndose de métodos y técnicas cuantitativas y cualitativas, recaba datos e información pertinente, para ser comparados con criterios previamente establecidos [5], para finalmente emitir juicios de valor muy bien fundamentados, facilitando la toma de decisiones sobre la mejor forma de intervenir en la realidad educativa que se está investigando [4-7]. Según Santos Guerra y otros [8], la evaluación es un momento de aprendizaje, que posibilita el análisis y la reflexión de los procesos de enseñanza-aprendizajes, transformando las prácticas pedagógicas, en pos de mejorar la calidad educativa.

2 Paradigmas de Investigación

El proceso evaluativo llevado a cabo, se enmarca dentro de los paradigmas cuantitativo y cualitativo de la investigación educativa. El enfoque cuantitativo, analiza la implementación de la propuesta pedagógica, y el tratamiento estadístico de la información. Se trata de identificar variables y causas, en los problemas de aprendizajes de los alumnos, que serán rigurosamente analizadas, para garantizar objetividad en los resultados. Se pone énfasis en todo aquello que sea observable y medible, para producir información que sea de utilidad para el control. Por otra parte, desde la perspectiva del paradigma cualitativo, se pretende realizar una valoración de los aspectos contextuales y situacionales, tales como las acciones humanas, valores y creencias de los alumnos, es decir que se trata de interpretar resultados a partir de la observación y la voz de los actores involucrados. Los docentes de la cátedra, actúan como investigadores, de acuerdo con sus propios conocimientos y creencias por lo cual no se puede exigir objetividad y neutralidad absoluta.

Algunos referentes importantes en educación, como Gimeno Sacristán y Pérez Gómez [7] exponen la importancia de combinar y complementar ambos paradigmas en una investigación integral de evaluación educativa. Tal como señalan los autores, para conocer el grado en el que se han alcanzado los objetivos de la propuesta metodológica, es necesario aplicar métodos cuantitativos, mientras que el paradigma cualitativo permite una retroalimentación de la acción didáctica. Luego de innumerables investigaciones en el campo educativo [7,8,9], actualmente se acepta que ningún

paradigma es mejor que otro, ya que generan distintos tipos de datos y poseen sus propias limitaciones, por lo que es necesario complementarlos con el fin de obtener mayor validez, credibilidad y rigor en los resultados alcanzados. A este proceso de combinación se le denomina triangulación [10], que busca lograr en el investigador evaluador subjetividad crítica racional, diferenciándola del subjetivismo como interpretación parcial y/o tendenciosa de los datos.

Denzin [11] define el término triangulación como “*la aplicación y combinación de varias metodologías de la investigación en el estudio de un mismo fenómeno*”. Para otros autores como Rodríguez, Pozo y Gutiérrez [12] la triangulación es una “*técnica de confrontación y herramienta de comparación de diferentes tipos de análisis de datos*” que contribuyen a validar un estudio de investigación y potenciar las conclusiones que de él se desprenden [7,12,13]. Existen diversas formas de triangular, en el presente artículo describen brevemente las que se utilizaron en el proceso evaluativo llevado a cabo: la *triangulación de datos*, que permite el análisis de información contrastando datos obtenidos por medio de diferentes estrategias y fuentes de información. La *triangulación entre métodos*, que permite contrastar los resultados analizando coincidencias y diferencias a partir de la aplicación de métodos cualitativos y/o cuantitativos en la investigación educativa [16].

3 Instrumentos de Recolección de Datos

Los instrumentos de recolección de datos usados en el proceso de investigación evaluativa se describen a continuación:

La *observación simple*, que trata de conocer situaciones que tienen un cierto carácter público; se utiliza para relevar formas de comportamiento, y conocer diversos aspectos de las conductas más manifiesta de los estudiantes. Los resultados de estas observaciones, apuntan a los aspectos visibles de la realidad social, aunque no por ello, pueda negarse su importancia.

Por medio de la *observación participante*, el investigador debe primeramente integrarse a los grupos de estudiantes, para una vez allí, ir realizando una doble tarea: desempeñar roles dentro del conjunto, e ir recogiendo los datos que procura conseguir. Esta tarea de observación, está a cargo de los alumnos tutores, ya que ellos son alumnos avanzados de la carrera, por lo que pueden, por su rol de alumnos, integrarse fácilmente a los grupos de estudios.

El *debate*, como un instrumento de intercambio de opiniones críticas, se lleva a cabo en un ambiente clase y con la dirección de un coordinador, rol que cumple el docente de práctica, para mantener el respeto y la objetividad entre las posturas educando y educador.

Por último, el principal instrumento de recolección de datos utilizados son todas las *producciones de los alumnos* como trabajos prácticos individuales y grupales, evaluaciones parciales, coloquios, defensa oral de trabajos, etc.

Los detalles de la instrumentación de las formas de recolección de datos expuestas se describirán en la sección 4.4.

4 Proceso Evaluativo

4.1 Localización

El proceso evaluativo estuvo destinado a evaluar la metodología propuesta en [3] y la asimilación de los contenidos en la asignatura *Programación Numérica y Cálculo Numérico* correspondiente al 2° cuatrimestre del 2° año de la carrera Licenciatura en Análisis de Sistemas, de la Universidad Nacional de Salta. Dichas asignaturas comparten el cuatrimestre con materias como Teoría de la Computación I, y Paradigmas y Lenguajes, por lo que, los alumnos deben cursar simultáneamente 3 asignaturas.

En el cursado de las asignaturas, hay estudiantes de diversas edades, aunque la mayoría están entre los 19 y 22 años. Se tienen alumnos que cursan por primera vez y recurrentes o repitentes. A pesar de estas diferencias, el denominador común es la inexperiencia y falta de madurez para afrontar el desafío que significa iniciar una carrera universitaria, instancia en la cual, no han adquirido todavía una disciplina emocional, ni han afianzado técnicas de estudios adecuadas para llevar adelante la carrera.

A grandes rasgos es posible describir la asignatura mencionada como el estudio y análisis de diferentes métodos y técnicas numéricas para la solución de ecuaciones, sistemas de ecuaciones, interpolación, regresión lineal, diferenciación e integración; adquiriendo la capacidad para implementar estos en un lenguaje de propósito general como Java, C++ o Delphi, o de uso específico como Maple o Mathematicas. En términos generales, la programación numérica, proporciona al alumno de Informática un medio para adquirir más experiencia en la programación de algoritmos.

La asignatura cuenta con un profesor responsable de cátedra, quien guía las prácticas pedagógicas de los demás docentes, como así también, es el encargado del buen desarrollo de las clases teóricas. Se cuenta además, con tres Jefes de Trabajos Prácticos, cada uno a cargo de una comisión de entre 20 o 25 alumnos, cuya función principal es guiar a los alumnos en la resolución de los trabajos prácticos y en el entrenamiento de la programación. Cada uno de estos docentes, cuenta con un tutor alumno, que ayuda en las labores pedagógicas diarias del aula.

4.2 Objetivos

Los objetivos y las metas que se buscó alcanzar con el proceso evaluativo se resumen en a) **Generales:** 1) Realizar una evaluación integral sobre la implementación de la propuesta pedagógica “*Scrum como Herramienta Metodológica en el Entrenamiento Cooperativo de la Programación*”; 2) Disponer de las acciones evaluativas, como verdaderas herramientas de adecuación de la propuesta metodológica a las características pedagógicas de los alumnos, permitiendo así reducir la tasa de deserción y subir el rendimiento académico, bajo el principio de mejorar la calidad educativa. b) **Específicos:** 1) Recoger la mayor cantidad de datos, información, documentos y observaciones que se generen en el contexto educativo de la enseñanza de la Programación Numérica y el Cálculo Numérico; 2) Realizar un pertinente procesamiento y análisis de la información recogida, utilizando técnicas y herramientas adecuadas;

3) Presentar permanentemente a los docentes de la cátedra, los resultados de las acciones evaluativas realizadas, de modo que puedan adecuar sus prácticas pedagógicas, en función de los objetivos curriculares que se persiguen.

Las **metas** que se alcanzaron fueron: 1) Realizar reuniones docentes para dialogar en relación a los aprendizajes de los alumnos; 2) Desarrollar habilidades sociales, de intercambio y negociación, madurando el pensamiento reflexivo y analítico entre los docentes; 3) Comprometer los esfuerzos de todos los docentes en el progreso continuo de la formación profesional y en la práctica pedagógica, cumpliendo con los objetivos curriculares, dinamizando las clases, potenciando el aprendizaje de los alumnos y contribuyendo a la mejora permanente de la calidad educativa.

4.3 Actividades y Temporalización

Las asignaturas se dictan dos días por semana. En cada día se imparten 2 horas de teoría y 2 de prácticas, cumpliendo así con un total de 4 horas de teoría y 4 horas de práctica, lo que contabiliza un total de 8 horas semanales. Además, se dispone de 2 horas semanales, para la realización de consultas sobre el desarrollo de las guías.

Las actividades y ubicación temporal que conforman el diseño de evaluación se detallan a continuación:

Actividad 1: “*Presentación Pedagógica*”

Presentación de los lineamientos pedagógicos de la metodología que se pretende implementar, exponiendo problemas de aprendizajes observados. Explicación de la metodología.

Responsable: Responsable de cátedra.

Destinatarios: Docentes.

Tiempo estimado: 4 hs.

Mes: agosto.

Actividad 2: “*Evaluación Diagnóstica Docente*”

Generación de un espacio para que los docentes expresen sus motivaciones y expectativas acerca de la metodología. Debido al ambiente de confianza entre los docentes, es posible realizar la actividad de forma oral (*conversación*), analizando los cuestionamientos conjuntamente. Los docentes (Jefes de Trabajos Prácticos) están a cargo de tomar nota del “torbellino de ideas” que genere la reunión y de cualquier otro aspecto que se considere importante.

Responsable: Responsable de cátedra.

Destinatarios: Docentes.

Tiempo estimado: 2 hs.

Mes: agosto.

Actividad 3: “*Evaluación Diagnóstica Estudiantil*”

Generación de un espacio para que los alumnos expresen sus experiencias personales, intereses y motivaciones que tienen acerca de la materia y de sus contenidos. Se busca recoger información sobre las competencias, técnicas, habilidades, procedimientos y

capacidades en el planteamiento y resolución de problemas computacionales que poseen los alumnos.

Responsable: Responsable de cátedra y Docente Auxiliares.

Destinatarios: Docentes.

Tiempo Estimado: 2 hs.

Mes: agosto.

Actividad 4: “*Presentación de Resultados*”

Presentación de los resultados obtenidos a los estudiantes, con el fin de generar un espacio de reflexión y concientización hacia los aspectos positivos de la metodología que se va a implementar y del cual forman parte. Esto es una forma de “hacerlos sentir importantes” del proceso educativo, por lo que se espera, a partir de esta actividad, un mayor compromiso, por parte de ellos.

Responsable: Responsable de cátedra.

Destinatarios: Estudiantes.

Tiempo Estimado: 2 hs.

Mes: agosto

Actividad 5: “*Recogida y Análisis Preliminar de Información*”

Obtención de toda información observable de las clases teóricas y prácticas. Los docentes (responsables de Prácticas) se encargan de realizar un análisis preliminar y plasmar resultados de forma gráfica o textual.

Responsable y Destinatarios: Docentes de la cátedra.

Tiempo Estimado: 2 hs

Mes: durante el transcurso del cuatrimestre.

Actividad 6: “*Reunión Docente Semanal*”

Las reuniones docentes semanales tienen por objeto realizar una puesta en común de las experiencias pedagógicas vividas y analizar toda información recogida por los auxiliares. En base a ello, es posible proponer nuevos cursos de acción, teniendo en cuenta los resultados positivos o negativos, que se van obteniendo.

Actores y Destinatarios: Docentes

Tiempo Estimado: 2 hs

Mes: durante el transcurso del cuatrimestre.

Actividad 7: “*Charla informativa final*”

Antes de que finalice formalmente la cursada de la materia, se organiza una charla con los alumnos para hacerlos partícipes de los resultados obtenidos acerca de la implementación de la propuesta pedagógica. En dicha reunión, ellos podrán plantear cuestionamientos, contar experiencias, problemas vividos, etc. Además cada docente puede dirigirse a los estudiantes para contarles su propia experiencia. De esta manera, todos los actores involucrados en el proceso educativo, y a partir de las experiencias de cada uno, se podrá analizar y proponer nuevas líneas de acción en pos de que la calidad educativa sea verdaderamente una realidad al alcance de todos.

Responsables: Docentes

Destinatarios: Estudiantes y Docentes

Tiempo Estimado: 1,5 hs

Mes: Noviembre.

4.4 Procesamiento de la Información

Para evaluar el aprendizaje de los alumnos se han observado y analizado con rigor los elementos que se han podido obtener de las prácticas pedagógicas, como ser trabajos grupales, individuales y observaciones del comportamiento estudiantil en las aulas, encuestas, criterios con indicadores de logros.

Los prácticos grupales permiten analizar en qué medida los alumnos fueron capaces de completar las guías de ejercicios, y por supuesto, ello en base a si comprendieron o no los conceptos enseñados en clases teóricas. Para ello fue necesario tener en cuenta los siguientes cuestionamientos:

¿Lograron terminar el práctico? Una respuesta afirmativa, significa que los alumnos han logrado superar el reto del trabajo grupal cooperativo, cumpliendo cada uno sus funciones y trabajando verdaderamente como un equipo.

¿Los ejercicios están resueltos correctamente? Si realmente fue así, los alumnos entienden la complejidad de los ejercicios, y son capaces de resolverlos correctamente.

Las respuestas negativas, resaltan que los estudiantes entienden los problemas propuestos y consientes de la complejidad inherente en su resolución, no son capaces de plantear una solución, o bien, no entendieron el caso de estudio. En estas situaciones, se decidió impartir clases de revisión de conceptos y prácticas adicionales, basada en la formulación y resolución de problemas.

Por medio de la defensa oral de los trabajos prácticos grupales, se analiza la dinámica y organización grupal, las ventajas y desventajas de trabajar en equipo, el compromiso individual y las razones por la falta de ello, la expresión del vocabulario técnico y el grado de desarrollo de sus capacidades comunicativas.

Las observaciones del trabajo en equipo desarrollado en el aula, permiten a los docentes guiar a los estudiantes en el buen desempeño del trabajo grupal, las formas de organización y comunicación, la división y asignación de tareas individuales, etc. También es posible identificar estudiantes tímidos, y poner en práctica acciones de socialización con sus pares, logrando paulatinamente una participación más activa.

Las actitudes de los alumnos en las clases teóricas reflejan si se sienten incentivados, motivados e interesados con los contenidos impartidos. Cuando no fue así, fue necesario averiguar a qué se debía el desinterés, ya sea porque las clases les resultan aburridas y desmotivadoras, si no logran comprender, o si no pueden visualizar alguna aplicación o importancia en el mundo real (fuera del ámbito puramente académico).

Los trabajos evaluativos parciales e individuales, permiten valorar el entendimiento de la complejidad algorítmica y la resolución de problemas. La evaluación individual muestra el grado en que los alumnos han aprendido adecuadamente, y su capacidad de poner en práctica los conocimientos sin ayuda de algún tutor.

Los trabajos evaluativos finales, individuales y orales, son los que terminan por acreditar los conocimientos sobre los temas impartidos en la materia. A partir del análisis del desarrollo de las evaluaciones, se puede inferir alguna métrica de desempeño de la metodología puesta en práctica.

4.5 Resultados de la Triangulación y Autoevaluación

Del análisis de resultados del procesamiento de información recopilada, se puede advertir una actitud de reserva y poco participativa del alumnado en las etapas iniciales del cursado de las materias. La asignación de roles, división de tareas y organización son los factores de mayor impacto en el trabajo grupal. La consideración y atención del aspecto psicológico del alumnado, sumado a las clases teórica centrada en la dinámica colaborativa, y prácticas enfocadas en la dinámica cooperativa, permitió el desarrollo nuevas habilidades, no solo como programador, sino también como ser humano. Este ambiente propició un mayor diálogo entre los protagonistas del proceso educativo, confianza, actitud activa y participativa, acciones que permiten a la cátedra detectar rápidamente problemas cognitivos y sociales que permiten instrumentar acciones correctivas.

El cambio en la estrategia educativa en las asignaturas implicó un mayor compromiso de los docentes, sobre todo en el modo de relacionarse con los alumnos. Tanto las clases teóricas como prácticas debían ser más interactivas entre profesor-alumno y alumno-alumno para favorecer una reestructuración cognitiva. Conscientes del esfuerzo, los docentes se mostraron alentados por los resultados obtenidos en la aplicación de la propuesta metodológica.

En líneas generales, puede decirse que aunque el proceso evaluativo es una tarea ardua (debido a la cantidad de alumnos y gran volumen de información para procesar), resultó ser una experiencia enriquecedora y reflexiva para el trabajo docente. Se desarrollaron habilidades sociales, de intercambio y negociación, madurando el pensamiento reflexivo y analítico entre los docentes.

El proceso de autoevaluación realizado, tiene sus resultados más relevantes en la elaboración de un plan de acción para la mejora de la calidad educativa. Con el compromiso del plantel docente, se puede llevar a cabo acciones viables que mejoren la calidad y equidad de la educación, en sus aspectos institucionales y pedagógicos, y que por lo tanto, impacten en los logros de aprendizaje de los alumnos.

Los resultados obtenidos, si bien son alentadores, no finalizan con la elaboración de un plan de acción, sino más bien, surge necesidad de realizar un seguimiento o monitoreo de la ejecución de los planes elaborados, así como su evaluación, que no se base en un mero control de actividades realizadas, sino que permita identificar logros y falencias, y que a la vez proporcione información útil y confiable para la comunidad educativa.

5 Referencias

[1] Costelloe E. (2004). "Teaching Programming The State of the Art. CRITE. Technical Report". Department of Computing, Institute of Technology Tallaght, Dublin, Ireland.

[2] De Giusti, Armando; Madoz Cristina, Gorga Gladys y otros. (2003). "Enfoques y herramientas en la enseñanza de un primer curso de computación (CS1)". IX Congreso Argentino de Ciencias de la Computación (CACIC). Pp. 1240-1251.

- [3] Barberis, A. R. y Del Moral Sachetti, L. E. (2016). "Scrum como Herramienta Metodológica en el Entrenamiento Cooperativo de la Programación: De la Teoría a la Práctica". XI Congreso de Tecnología en Educación y Educación en Tecnología 2016 (TE&ET 2016). pp. 365-374. Universidad de Morón, Argentina.
- [4] Lafourcade, Pedro. (1969). "Evaluación de los Aprendizajes". Ed. Kapeluz. Buenos Aires, Argentina.
- [5] Nieto, J.M. (1994): "La Autoevaluación del Profesor. Cómo puede el Profesor Evaluar su Propia Práctica Docente". Editorial Escuela Española. Madrid, España.
- [6] Duro, Elena y Nirenberg, Olga. (2014). "Autoevaluación de Escuelas Primarias. Instrumento de Autoevaluación de la Calidad Educativa- IACE". Argentina.
- [7] Sacristán; Gimeno y Pérez Gómez; Ángel. (1996). "Comprender y Transformar la Enseñanza". Quinta Edición. Ediciones Morata. Madrid, España.
- [8] Santos Guerra, Miguel Ángel. (1991). "La evaluación: un proceso de diálogo, comprensión y mejora". VIII Jornadas de Estudio sobre la Investigación en la Escuela. Universidad de Málaga.
- [9] Cook y Reichardt. (1995). "Métodos Cualitativos y Cuantitativos en Investigación Evaluativa". Ediciones Morata. Madrid, España.
- [10] Anderson, Gary; Augustowsky, Gabriela y otros. (2007). "La Investigación Educativa. Una Herramienta de Conocimiento y de Acción" (Compilación). Ediciones Novedades Educativas de México.
- [11] Denzin, Norman. (1970). "The Research Act: A Theoretical Introduction to Sociological Methods. New Jersey: Transaction Publishers.
- [12] Rodríguez, C., Pozo, T. & Gutiérrez, J. (2006). "La triangulación analítica como recurso para la validación de estudios de encuesta recurrentes e investigaciones de réplica en Educación Superior". RELIEVE. Revista electrónica de Investigación y Evaluación Educativa, 12 (2), p. 289-305.
- [13] Aguilar Gavira, Sonia y Barroso Osuna, Julio. (2015). "La Triangulación de Datos como Estrategia en Investigación Educativa". Píxel-Bit. Revista de Medios y Educación. N° 47.
- [14] Alzás, García; Teresa, Casa Garcia; Luis; y otros. (2016) "Revisión Metodológica de la Triangulación como Estrategia de Investigación". Congreso CIAQ. Investigación Cualitativa en Ciencias Sociales.

Experiencia de enseñanza de la programación en un contexto de vulnerabilidad social

Francisco Bavera¹, Sonia Permigiani¹, Gastón Scilingo¹, María Marta Novaira¹, Cecilia Kilmurray¹, Marcelo Arroyo¹, Cesar Cornejo¹, Violeta Castresana², and Carmiña Verde²

¹ Dpto. de Computación, FCEFQyN, Universidad Nac. de Río Cuarto, Argentina

² Fac. Cs. Humanas, Universidad Nac. de Río Cuarto, Argentina.

{pancho, spermigiani, gaston, mnovaira,
ckilmurray, marroyo, ccornejo}@dc.exa.unrc.edu.ar
{violetacastresana, carmiverde}@gmail.com

Resumen En la actualidad un alfabetismo muy popular es el Alfabetismo digital, que implica el conocimiento y la comprensión de tecnologías digitales. En este trabajo abordamos la enseñanza de la programación, a través de la familiarización con algunas herramientas tecnológicas visuales, y el acceso a las TIC. En particular, reportamos acerca de vivencias y aprendizajes con niños en situación de vulnerabilidad social quienes concurren al Hogar “Ciudad de los Niños”, ubicado en Río Cuarto, Córdoba, Argentina.

Keywords: Programación, Herramientas Tecnológicas, Contexto Social Vulnerable

1. Introducción

Aprender con el lenguaje de las ciencias informáticas implica aprender un nuevo alfabetismo, pero tradicionalmente el término alfabetismo se lo consideró como la habilidad de usar texto para comunicarse a través del espacio y el tiempo.

Los alfabetismos están vinculados a las relaciones sociales, institucionales, culturales e históricas. Además están conectados con las identidades sociales y con el hecho de ser determinados tipos de personas.

En la actualidad uno de los alfabetismos más populares es el Alfabetismo digital también conocido como alfabetización digital y alfabetización multimedia donde la habilidad es la de localizar, organizar, entender, evaluar y analizar información utilizando tecnología digital. Implica tanto el conocimiento de cómo trabaja la alta tecnología de hoy día como la comprensión de cómo puede ser utilizada. Las personas digitalmente alfabetizadas pueden comunicarse y trabajar más eficientemente, especialmente con aquellos que poseen los mismos conocimientos y habilidades.

En este artículo se presenta la experiencia de enseñar computación a niños que residen en la Institución *Ciudad de los Niños*. Esta Institución se focaliza en la atención integral de niños y niñas en situación de vulnerabilidad. Los niños

que residen (transitoriamente) en esta institución provienen de contextos familiares altamente vulnerables, atravesados por situaciones continuas de violencia, inestabilidad laboral y afectiva, con escasos recursos para hacerse cargo de la crianza.

En la actualidad existen numerosas herramientas tecnológicas que están al alcance de niños que viven en un contexto social de clase media y alta, pero que los niños a los que está destinado el proyecto, que origina la experiencia que presentamos aquí, no tienen posibilidades de acceder.

La experiencia de enseñar Ciencias de la Computación a los niños que viven en la Ciudad de los Niños es novedosa en varios aspectos:

- Contexto de niños en situación particularmente vulnerable.
- Actividad extracurricular, no obligatoria.
- Limitaciones en recursos e infraestructura.
- Características de los niños (grupo etario variado, algunos niños concurren a escuelas con modalidad especial).

Para la realización del Proyecto PICUCo³, se analizaron numerosas herramientas y materiales para enseñar conceptos de computación. El Criterio que se utilizó para la selección de las herramientas fue variado; algunos materiales fueron elegidos por limitaciones de recursos al comienzo del proyecto (no había computadoras que se pudieran utilizar en el laboratorio de la Ciudad de los Niños); otras herramientas fueron seleccionadas por su simplicidad y adaptabilidad a las diferentes edades de los niños. Además, algunas herramientas fueron descartadas por limitaciones de hardware; otras porque no contaban con documentación en español, y algunas no llegaron a utilizarse por limitaciones de tiempo.

Durante el desarrollo del proyecto se presentaron distintos tipos de dificultades. Hubo complicaciones técnicas, limitaciones de recursos, entre otras. Además surgieron complicaciones relacionadas con la participación y motivación de los niños. También se presentaron obstáculos relacionadas con el nivel cognitivo de los niños.

A través de la experiencia desarrollada a lo largo de todo un año se pudo observar que el uso de herramientas visuales favorece a la motivación y aprendizaje de conceptos de computación en niños que por su contexto particular de vulnerabilidad social no tenían acceso a tecnologías de la información y comunicación.

2. Contexto: "La Ciudad de los Niños"

La "Ciudad de los Niños" es una obra destinada a la atención integral de niños y niñas en situación de vulnerabilidad. Cuenta con tres grandes proyectos: un taller de oficios, una escuela con nivel primario y secundario destinada a la inclusión educativa de adolescentes y un proyecto de acogimiento a la infancia

³ Proyecto Integrador Ciudad de los Niños, Universidad y Computación

que consta de 4 casas en la que viven transitoriamente entre 30 y 40 niños y niñas bajo medida excepcional, según Ley nacional Nro. 26.061 y provincial Nro. 9944 de Protección integral de la Infancia y Adolescencia. El proyecto Casas es el que dio origen a la Obra hace 26 años y los otros fueron pensados en respuesta a necesidades específicas que fueron surgiendo con el devenir de la tarea.

La población de niños y niñas que reside en las Casas es la destinataria de las actividades presentadas en este trabajo. Son niños y niñas que viven temporalmente en esta institución hasta tanto se generen condiciones apropiadas para retornar a sus hogares o se busquen soluciones alternativas que garanticen el derecho a vivir en familia. Todos ellos provienen de contextos familiares altamente vulnerables, atravesados por situaciones continuas de violencia, inestabilidad laboral y afectiva, desorden en las relaciones interpersonales, con escasos recursos para hacerse cargo de la tarea de crianza, produciendo un impacto profundo en los niños/as que se ven arrojados a vivir una niñez sin cuidados, sujetos a los más diversos riesgos, una niñez sin infancia.

En el contexto social del que se desprenden estos niños, la desigualdad que sufren no favorece a las relaciones sociales, producen tensiones y generan desconfianza y baja estima. Es un contexto desfavorable que dificulta el alcance de los niños a las tecnologías que pueden manipular otros niños, de sus mismas edades, en contextos sociales más favorables. Se debe comenzar a pensar desde otra concepción relacionada a la igualdad de posiciones, para poder generar acciones políticas sociales para la educación que nos aseguren los trayectos, más que las “adquisiciones sociales” [5].

3. Herramientas y Materiales de Aprendizaje

En la actualidad existen numerosas herramientas visuales y materiales para enseñar computación en todos los niveles educativos. En esta sección listamos todas las herramientas que se tuvieron en cuenta para este proyecto y sus características.

- Mis Amigos Robóticos [9]: Libro con una colección de actividades y juegos off-line para enseñar Ciencias de la Computación a personas de todas las edades.
- Paint: Editor simple y visor de imágenes desarrollado por Microsoft.
- Code.org[13]: La Hora del Código[12] es una campaña mundial para que niños y niñas completen una hora jugando, aprendiendo y creando con tecnología on-line.
- Scratch [15]: Entorno visual de programación libre principalmente orientado a la enseñanza mediante la creación de juegos.
- Khan Academy [16]: Cursos libres, on-line, de código abierto, que se utilizan para enseñar distintas temáticas. Existen numerosos cursos para enseñar conceptos de matemática, lógica, computación, además de otras Ciencias.

- Alice[17]: Es un entorno innovador 3D, que permite de forma sencilla crear animaciones para contar historias, jugar juegos interactivos o videos que se pueden compartir a través de la web.
- Kturtle[11]: Es un programa educativo diseñado para la programación, forma parte del entorno de escritorio KDE.
- Minibloq[10]: Entorno gráfico de programación de robots y tecnología Arduino.

3.1. Herramientas Seleccionadas

De las herramientas mencionadas anteriormente, si bien utilizamos la mayoría, fuimos seleccionando algunas según diferentes criterios en diferentes momentos del proyecto.

Al Inicio del proyecto, en la Ciudad de los Niños no se disponía de computadoras que pudiéramos utilizar, por lo que el primer criterio utilizado para seleccionar actividades planteadas en Mis Amigos Robóticos [9] fue debido a una limitación física. El material elegido cuenta con numerosas actividades para enseñar conceptos básicos de computación sin la necesidad de contar con una computadora.

Ante la necesidad de una forma de “medir” y analizar los conceptos de programación que los niños lograban aprender se eligió tomar uno de los cursos que se plantean en La Hora del Código (<https://studio.code.org/hoc>), ya que no sólo es útil para enseñar conceptos de computación, también es una forma de evaluar los avances en el aprendizaje, a través de la distinción de los pasos saltados o completados en el transcurso de la actividad.

Durante el transcurso del proyecto se planteó la creación de videoclips, por el interés que manifestaban los niños en ver y escuchar videos de youtube. Para esto se seleccionó la herramienta Scratch ya que permite la creación de animaciones en forma sencilla sin que necesiten conocer la herramienta.

Para brindar nociones de robótica se utilizó un robot y para explicar cómo programarlo se eligió MiniBloq por su simplicidad a la hora de explicar y programar un robot con tecnología Arduino.

Decidimos no utilizar aún la herramienta Alice, debido a que, para que el programa funcione y se visualice en forma correcta, se requiere de un hardware con características superiores a las que se disponían en el laboratorio. Dado que se necesitaba una herramienta sencilla que permitiera personalizar y simplificar el armado y visualización de Historietas, se decidió desarrollar un Prototipo de herramienta de software en el contexto de este proyecto. Esta herramienta permite la generación de historias secuenciales y también con bifurcaciones temporales. Se pretende extenderla para incluir la generación de animaciones. Esta herramienta es libre y de código abierto. Esta disponible en:

<https://github.com/marcelodarroyo/html5-book.git>. Actualmente la herramienta se encuentra en etapa de prueba.

4. Descripción de la Experiencia

Los destinatarios del Proyecto son niños y niñas menores de edad que viven temporalmente en la “Ciudad de los niños” por carecer de un ámbito familiar apropiado (bajo medida excepcional que los protege de situaciones de riesgo).

Son niños y niñas que, de Lunes a Viernes, asisten a la escuela en sus tres niveles inicial, primario y secundario; 3 de ellos concurren a una Escuela de Modalidad Especial.

Durante el año 2015 se llevaron a cabo 24 encuentros los días sábados en la Ciudad de los Niños. Estos encuentros se concretaron en dos períodos: el primero abarcó desde el mes de abril hasta junio y el segundo, desde septiembre hasta noviembre. Los niños asistentes se dividieron en dos grupos conformados con 6 y 11 integrantes, respectivamente, quienes trabajaban durante una hora en cada encuentro. Los grupos se dividieron (a sugerencia de los responsables de la institución) en niños (varones) y niñas cuyas edades oscilan entre los 8 y los 15 años.

A continuación describimos las actividades desarrolladas:

Inicialmente, a modo de pre-test, se pidió a los niños que plasmaran la idea que tenían de lo que significaba Computación a través de un dibujo o con palabras.

Luego, mediante el uso de un vocabulario robot predefinido, se realizaron actividades para enseñar a transformar acciones del mundo real en instrucciones, a la vez que adquirieron práctica en la codificación de instrucciones y comprendieron la necesidad de precisión en la codificación.

Siguiendo la propuesta de code.org, se desarrolló uno de los cursos online, La Hora del Código.

Además se realizaron actividades propuestas en Program.ar [8] que incluyen la utilización de Scratch. Luego se avanzó con actividades también en Scratch, para la creación de videoclips y tarjetas animadas.

Finalmente Se planteó la realización de una historieta.

5. Impacto de la Experiencia

Durante el desarrollo del proyecto se presentaron distintos tipos de dificultades que son útiles para analizar y ayudan a ganar experiencia en el desarrollo del proyecto.

Algunos de los problemas que se presentaron tuvieron que ver con la escasez de recursos, ya que al comienzo del proyecto los niños debían compartir máquina porque no alcanzaba una computadora por niño. Este problema fue resuelto a través de la compra de máquinas destinadas al laboratorio de la Ciudad de los Niños con recursos provenientes del proyecto de extensión PICUCo, financiado

por la SPU ⁴ de la Nación.

Además se presentaron complicaciones relacionadas con la participación y motivación de los niños, ya que al ser una actividad extracurricular, muchos de ellos se desmotivaban con las actividades que requerían de más de dos encuentros consecutivos o que necesitaban de explicaciones específicas en la pizarra. A raíz de esto se re-planificaron algunas actividades.

Una de las complicaciones técnicas que se presentó, fue el problema con la velocidad de Internet que se disponía en el laboratorio para el desarrollo de actividades on-line como por ejemplo la Hora del Código. A raíz de esta complicación no se pudo avanzar con otros cursos de code.org porque el acceso a un proveedor diferente de Internet no es una variable que pueda manejar este proyecto.

También hubo complicaciones en algunas actividades con los niños que tienen dificultades para leer y escribir.

Una característica importante a analizar, a la hora de seleccionar las herramientas tecnológicas y actividades que se realizan en este tipo de proyectos, es el nivel cognitivo que necesitan tener los niños que la utilizan para poder resolver las actividades que se planteen en forma autónoma. Por ejemplo, a la hora de guardar las diferentes imágenes creadas con Paint, los niños con dificultades para leer y escribir necesitaron de la asistencia de los docentes para escribir el nombre del archivo. Estos mismos niños cuando se planteó la actividad de crear una historieta se limitaron a seleccionar imágenes que representaban los personajes y fondos de la historia sin poder realizar ningún diálogo o explicación por escrito. También requirieron asistencia para la lectura de las instrucciones y ayudas para resolver los problemas que se plantean en code.org.

Durante el desarrollo de una de las actividades planteadas en la Hora de Código, “Angry Birds”, pudimos observar diferentes comportamientos que son objeto de discusión y describimos a continuación.

Por un lado, frente a la actividad mencionada, “Angry Birds”, notamos que las niñas que asisten a una escuela especial, se resistían a guiar por un laberinto al pájaro enojado para alcanzar al cerdo que robó sus huevos. Intuimos que el rechazo se debía al grado de dificultad que tenía la tarea, que incluía habilidades de arrastrar y soltar, para acomodar los bloques de instrucciones en la posición correcta (encastrar), como también nociones de direccionamiento, girar a la izquierda o a la derecha. Es por esto que para el caso particular de estas niñas decidimos cambiar la actividad de la hora de código, por una mucho más básica que las ayudara a superar esas habilidades. Descubrimos que no sólo se motivaron para realizar las actividades planteadas en las etapas 1, 2 y 3 (mapas felices, arrastrar y soltar, y muévelo-muévelo), sino que además pudieron avanzar en la actividad a la que se resistieron inicialmente, que corresponde a la cuarta etapa del mismo bloque de actividades.

⁴ Secretaría de Políticas Universitarias

Por otro lado, notamos que solo dos varones y dos niñas, lograron completar los 20 pasos correspondientes a la actividad del “Angry Birds”, sin saltarse ningún paso. Uno de los varones incluso eligió continuar con otro curso más avanzado, planteado en la hora del código, ya que le sobró tiempo.

Dos de los varones más pequeños (9 y 10 años), salteaban los pasos que tenían la dificultad particular de resolver el laberinto de una manera eficiente, es decir, los bloques anidados que se debían usar de la manera prefijada (no se podían desarmar).

Algo a remarcar del Angry Birds es que los bloques de instrucciones, relacionados con los movimientos en el laberinto, incluyen los de girar el objeto (girar a la izquierda y hacia la derecha). Aunque parezca algo intuitivo, para la mayoría de los niños reconocer qué instrucción dar para realizar los movimientos en la dirección correcta resultó complicado, requirió la repetición de los pasos muchas veces y en algunos casos incluso la asistencia de los docentes.

6. Conclusiones

Podemos concluir que es muy difícil poder evaluar de forma cuantitativa al grupo de participantes. El grupo (en su totalidad) carecía de experiencias en este tipo de actividades dirigidas a esta franja etario en este contexto social.

Inicialmente se elaboró una propuesta de actividades y un plan de trabajo. Tanto las actividades como el plan de trabajo si bien fueron implementadas, las mismas fueron continuamente ajustadas y reformuladas por distintos motivos, algunos relacionado con problemas técnicos, otros tuvieron que ver con el nivel cognitivo de los niños y otros por la pérdida de interés en el tipo de actividades planteadas.

Se cree que actualmente se cuenta con un plan de trabajo maduro, realista y realizable que se ajusta a los objetivos planteados. Lo cual facilitará la ejecución de las actividades durante el segundo año de ejecución.

Se generó material para el desarrollo de las actividades. El cual estará disponible próximamente en formato digital y con acceso libre bajo licencia Creative Commons.

Se desarrolló un Prototipo de herramienta de software para el uso en las actividades. Esta herramienta permite la generación de historias secuenciales y también con bifurcaciones temporales. Se pretende extenderla para incluir la generación de animaciones. Esta herramienta es libre y de código abierto. Esta disponible en: <https://github.com/marcelodarroyo/html5-book.git>.

6.1. Trabajos a Futuro

Las investigaciones que surjan de esta experiencia y la propuesta puede ser adoptada por las asignaturas “Practica Docente” y “Taller de Diseño de Software Educativo” del Profesorado de Ciencias de la Computación. Carrera que forma docentes de computación para todos los niveles educativos. La asignatura “Practica Docente” tiene como objetivo la realización de una práctica docente

(desde la planificación, fundamentación e inserción en una institución educativa) de los futuros docentes. Por lo cual, los estudiantes de esta asignatura podrían realizar su práctica en instituciones similares a la de este proyecto y realizando las actividades propuestas en el mismo. En cuanto a la asignatura “Taller de Diseño de Software Educativo” tiene como objetivo el estudio y realización de distintos software educativos, con lo cual, se podría poner el foco en desarrollar y/o estudiar alternativas de software educativo para aplicar en este proyecto.

En el caso de la Carrera Profesorado en Ciencias de la Computación, en especial en las materias arriba mencionadas y las acciones propuestas, creemos que se generaría una innovación pedagógica, ya que, se orientaría las acciones y estudios de los alumnos (del profesorado) a destinatarios que se encuentran en grupos de riesgo, y además, se cambia el paradigma y las herramientas a utilizar por los futuros docentes para entusiasmar y motivar a los alumnos (si se hace foco principalmente en mini robots).

Creemos que el proyecto debe ser replicado y esperamos un importante impacto en los receptores, pero para poder justificar adecuadamente esta afirmación necesitamos poder evaluar los resultados y realizar las modificaciones que se desprendan de la experiencia adquirida con la ejecución del proyecto.

Referencias

1. “Computational thinking and thinking about computing” Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 1881, pages 3717-3725, The Royal Society, Volume 366,2008
2. “Formación docente y TICs: sobre un proyecto de extensión para el aprovechamiento de los recursos informáticos en las escuelas” .Merayo, Miriam; Espósito, Sandra; Gavino, Sergio; Guzner, Graciela Beatriz. I Congreso de Tecnología en Educación y Educación en Tecnología. 2013
3. “Formación docente y TICs: sobre un proyecto de extensión para el aprovechamiento de los recursos informáticos en las escuelas.” . Merayo Miriam, Espósito Sandra, Gavino Sergio y Guzner Graciela., TE&ET 06. 2006.
4. “Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas”. Fundación Sadosky. Enero 2013.
5. “Repensar la justicia social”. Francois Dubet, Ed. Siglo Veintiuno. ISBN 978-987-629-163-7, Agosto 2014.
6. “Mindstorms: Children, Computers, and Powerful Ideas” . Papert, Seymour. New York, NY, USA, ISBN 0-465-04627-4. Basic Books, Inc.,1980
7. “Computadoras en las Escuelas (y niños para compartirlas)”. Adrián Paenza. Pagina 12, 14 de Agosto 2014.
8. Actividades para aprender a programar <http://programar.gob.ar/descargas/manual-docente-descarga-web.pdf>
9. Mis Amigos Robóticos <http://www.horadelcodigo.cl/TMAR>
10. Minibloq blog.minibloq.org/
11. Sitio oficial de Kturtle . <https://edu.kde.org/kturtle/>
12. La Hora del Código. <https://studio.code.org/hoc/1>
13. Sitio oficial de Code.org. <https://code.org/>
14. UNICEF (2004). Experiencias de inclusión educativa desde la perspectiva de aprender juntos. Estudio de casos en regiones de Argentina. Argentina: Buenos Aires.

15. Scratch <https://scratch.mit.edu>
16. Khan Academy <https://es.khanacademy.org>
17. Alice <http://www.alice.org>

Integración de materiales didácticos en 3D para el Aprendizaje Basado en Problemas en la enseñanza de Artes

Fernandez, Mirta^{1,2} - Godoy Guglielmone Ma. Viviana²

¹ Facultad de Artes, Diseño y Ciencias de la Cultura (FADyC)

² Facultad de Ciencias Exactas y Naturales y Agrimensura (FaCENA)

Universidad Nacional del Nordeste (UNNE)

{mirtagf@hotmail.com, mvvg2001@yahoo.com}

Resumen: Consistente el replanteo de las prácticas de enseñanza habitual en una asignatura, desde una perspectiva crítica pero basada en evidencias; en este trabajo se analiza el grado de consecución de unos objetivos de aprendizajes y se plantea la necesidad de ampliar la mirada para avanzar hacia el reconocimiento de formas alternativas de adquirir conocimientos. Mediante la inclusión de materiales didácticos en 3D al comienzo de un eje curricular y su integración en la estrategia de ABP (Aprendizaje Basado en Problema). La propuesta surge de dicha revisión, a fin de mejorar la experiencia en el estudio y los vínculos en torno a la incidencia del ordenador y el software en las artes. Se entiende que dicho replanteo redundará en mejores resultados de los obtenidos, dado que se ponen en juego diversos aspectos cognitivos.

Palabras clave: Aprendizaje Basado en Problemas, Enseñanza en Tecnologías Aplicadas, Revisión de metodologías, Materiales educativos en 3D.

1. Introducción

Las diversas expresiones artísticas introducen la TICs (Tecnologías de la información y la Comunicación) en arte convencional o generan nuevas intervenciones [1], [2] y [3]. Resultado del diálogo entre diferentes disciplinas tradicionales como la literatura, la pintura, el dibujo, la escultura, la música, el teatro, la danza, la fotografía y el cine.

La Asignatura Introducción a las Tecnologías Aplicadas al Arte, se dicta en la FADyC de la UNNE y tiene como objetivos brindar una aproximación acerca de “Como las tecnologías han ido fluctuando a través del tiempo y han conseguido instalarse para la obtención de diversos tipos de producciones”. En un eje temático, puntualmente se aborda mediante clases teóricas y prácticas, la incidencia del ordenador y el software en las artes [4].

En consonancia con lo expuesto por [5], el presente trabajo surge del resultado de integrar investigación y práctica docente; coincidiendo que esta integración es un proceso sistemático de análisis de la propia docencia (procedimientos, ejercicios, evaluación, etc.) y sus efectos; para sencillamente “hacer posible el aprendizaje de los alumnos”.

Se introduce brevemente en concepciones del ABP, como una alternativa estratégica para éste propósito, en el sentido que es una metodología que se centra en

el aprendizaje, la reflexión y la investigación, permitiendo que los alumnos se reúnan para analizar en grupos y resolver un problema seleccionado o diseñado especialmente para el logro de ciertos objetivos de aprendizaje, acompañados del docente en rol de facilitador.

A fin de complementar la propuesta, se incorporan recursos de producción docente: un conjunto de materiales didácticos tridimensionales, específico y disciplinar; para este propósito.

Seguidamente, desde una perspectiva crítica pero basada en evidencias, se analiza el grado de consecución de unos objetivos de aprendizajes y se plantea la necesidad de ampliar la mirada para avanzar hacia el reconocimiento de formas alternativas de adquirir conocimientos.

Se describe la propuesta; se plasman consideraciones acerca de su implementación y la relevancia de ampliar la discusión y estudio sobre competencias de los propios docentes como mediadores de enfatizar el desarrollo metacognitivo del aprendizaje.

1.1. Antecedentes de la experiencia

Tal como se mencionó, en un eje temático particularmente, se aborda mediante clases teóricas y prácticas la incidencia del ordenador y el software en las artes.

Consistente en la presentación de varios casos de estudios, como disparadores, en experiencias anteriores se proyectaban producciones artísticas que hacían uso de diversas técnicas. En el marco de la literatura que da los fundamentos a la temática y desarrollada como lo ilustra la **Fig. 1.**



Fig.1. Representación de la metodología de trabajo original.

En la misma, se alcanzaba resultados satisfactorios en que los estudiantes se encontraban con “contradicciones” acerca de lo conocido e interpretado y se lograba aportes de interés para los objetivos. No obstante, se evidenció que la baja interacción y simple descripción de las técnicas, eran insuficientes para lograr aportes de relevancia y un conocimiento significativo.

Además, en dichos ciclos los estudiantes mostraron gran interés por conocer la diversidad de producciones de diferentes índoles y materiales, a su vez que el escollo mayor se encontraba en establecer vínculos, entre las concepciones teóricas y las prácticas artísticas actuales.

1.2. Propuesta de intervención.

Tal como lo expone Mon, Del Giorgio y Donadello [6], “humanizar” la enseñanza modificando los roles e incluyendo dinámicas más abiertas, redundando en una mejora del ambiente de la clase y en la relación estudiante-docente. Por tanto, se replantea la dinámica en un doble sentido: metodológico y de incorporación de materiales; tal como se ilustra en la Fig. 2, y se describe en los siguientes apartados.

En sentido metodológico, el ABP como una alternativa adecuada que se centra en el aprendizaje, la reflexión y la investigación, permite que los alumnos se reúnan para analizar en grupos y resolver un problema seleccionado o diseñado especialmente para el logro de ciertos objetivos de aprendizaje, acompañados del docente en rol de facilitador.



Fig. 2. Incorporación de materiales en la dinámica de ABP.

En cuanto a los materiales, como se ilustra, se integraron materiales didácticos: generados con componentes de RA (Realidad Aumentada) y mediante impresión en 3D con materiales reciclables.

1.3. Caracterización del Aprendizaje Basado en Problema

El ABP se considera un método pedagógico de aprendizaje basado en la investigación, que tiene raíces en la filosofía constructivista, en particular el trabajo de Piaget, Dewey, y Vygotsky [7], [8] y [9]. Implican la creación de preguntas, haciendo la investigación para hacer frente a los interrogantes, analizar, interpretar, y dar con posibles soluciones [10] y [11].

En el ABP, los docentes asumen el papel de facilitador o entrenador más que el transmisor de conocimiento y los estudiantes toman problemáticas o retos, realizan investigaciones, y mediante trabajo en colaboración abordan las soluciones [12], [13], [14].

Centrado en el estudiante y se inicia con un problema real o realístico, algunas de sus características se resumen en:

- Debe plantear un desafío cognitivo que los motive a indagar.
- Centrado en el trabajo colaborativo.
- Debe provocar cambios significativos en el aprendizaje.
- Debe relacionarse con los objetivos del curso y con situaciones de la vida real.

Por otra parte, la literatura extiende que *“los buenos problemas tienen material interesante, provocador, como por ejemplo una breve historia o descripción; ...”* [15]. Bajo este escenario, se trabajó en la búsqueda de material específico disciplinar.

1.4. Materiales educativos tridimensionales: RA e Impresión 3D

En el ámbito educativo las tecnologías en 3D constituyen elementos especialmente eficaces en lo relacionado con la forma en que los estudiantes perciben la realidad física, apreciar sus dimensiones, con objeto de facilitar la captación de sus diversas particularidades, en ocasiones imperceptibles para los sentidos.

Tales son las tecnologías con RA y la impresión en 3D, factibles generar modelos que simplifican la complejidad multidimensional del mundo circundante, lo que, desde una perspectiva académica, aporta completud a cualquier experiencia de aprendizaje [16].

La RA, comprende aquella tecnología capaz de complementar la percepción e interacción con el mundo real, brindando al usuario un escenario real, aumentado con información adicional generada por ordenador [16]. Sus potencialidades son infinitas y los últimos estudios demuestran el interés en diversos contextos disciplinares [17], [18], y [19].

Incluida también en el grupo de las denominadas tecnologías emergentes, la impresión 3D abarca diferentes alternativas de manufactura que permiten crear objetos a partir de un modelo digital, atravesando una amplia gama de actividades productivas y sociales [20].

El informe NMC Horizon Report, sobre la Educación Científica Superior, otorgan un apartado especial a ambas tecnologías, mencionando su relevancia en la enseñanza, el aprendizaje, o la investigación creativa, avizorando un futuro prometedor a tres años aproximadamente [21].

La conjunción de la integración de materiales con la metodología surge del trabajo interdisciplinario de los docentes, para la generación de insumos aprovechables en el aula, en este caso materiales tridimensionales.

2. Materiales y Métodos

Considerando las acepciones vinculadas con las fases que en que ocurre el ABP, [12], [13], [14] [22], [23] y [24] se realiza una adaptación al contexto mediante: delimitación los objetivos de aprendizaje, selección de materiales, diseño de las estrategias y definición de criterios de evaluación. A modo de retroinformación se prevé la evaluación de la propuesta.

3. Resultados

En general se consideró de vital importancia que la experiencia guarde relación con los conocimientos previos. Al mismo tiempo, poner de manifiesto una serie de

elementos desconocidos que demanden más información, tal como la producción de artística con tecnología digital propiciando la naturalidad del contexto.

3.1. Objetivos propuestos

Mediante la proyección de casos de estudios y reflexionar en torno a:

- Generación de arte por computadora.
- Organizar la teoría marco de las corrientes artísticas actuales y reflexionar en torno a la problemática de la incidencia del ordenador y el software en las artes.
- Pronosticar los usos diversos de las diferentes técnicas.

3.2. Selección de materiales

Su incorporación pretende que los estudiantes reconozcan materiales, técnicas y potencialidades de las tecnologías actuales, sin desestimar los fundamentos artísticos que sustentan la asignatura en particular y la carrera en general. Por lo que se seleccionaron y reprodujeron piezas artísticas, monumentos y esculturas, tales como se muestran en la **Fig. 3** y **Fig. 4**.



Fig. 3. Ejemplos de animación de esculturas y monumentos con RA.



Fig. 4. Ejemplo de réplica de escultura mediante impresión en 3D.

Dichos materiales surgen del trabajo interdisciplinario de los docentes que cuentan con medios para reproducirlos y valiéndose de recursos disponibles con software Open Source y de licencia Creative Commons.

3.3. Diseño la estrategia de ABP

A nivel de actividad se propone las fases que se muestran en la **Fig. 5**:



Fig. 5. Método de ABP adaptado al contexto.

- i) Para el C1 (generación de esculturas mediante tecnología de RA), se menciona algunos detalles de la obra, se proyecta animaciones de las piezas obtenidas de dicha técnica, adelantado pistas y se sugiere aplicar conocimientos previos
 - ii) Se abre un debate inicial que pone a consideración el interrogante disparador surge en torno a “¿Con que técnica creen ustedes se ha sido concebida?” Los alumnos intentarán explicar y desarrollarán ideas relacionadas con los procesos y conceptos subyacente.
 - iii) Se organizan las ideas bajo una guía tendientes a reconocer características de las producciones artísticas con materiales analógicas y las digitales, vincular los “principios” que propone la bibliografía de referencia y categorizar el Caso.
 - iv) Se comparte las elaboraciones y se describe la técnica con la que fue concebida dichas esculturas y se la caracteriza.
- Se itera para el C2 (Impresión de escultura en 3D).
- v) Se invita a experimentar con marcadores y obtener las escenas definidas para el C1 y las piezas para el C2 (apreciar el material, textura, forma, etc.) y replantear el debate en torno a la Tecnología digital para la producción artísticas.
 - vi) Se proyectan otras tendencias emergentes en la generación de arte por computador.
 - vii) Se solicita realizar un informe grupal, complementado con la bibliografía, discriminando el C1 y C2 y una conclusión final.
- La evaluación se hará en base de valorar el grado de integración y la verificación de los aportes obtenidos cumpla con los objetivos del problema.

3.4. Evaluación de la propuesta

Para los alumnos se tendrá en cuenta criterios cualitativos en comparación con la experiencia que antecede, en bases a los aportes significativos que se alcancen, tanto desde el punto de vista tecnológico como artísticos.

En torno a la propuesta en general, se examinará el grado de participación y el interés de los estudiantes; de la misma forma, el rol de facilitadores de los docentes y el control de los tiempos asignados para la experiencia.

4. Discusiones

Tal como lo refleja Burbules [25] en su estudio, la experiencia indica que, mediante las TICs, el estudiantado posee una variedad de recursos al alcance que ayuda en gran medida; en otras puede ser motivo de repensar nuevas formas de presentar los interrogantes: dado que los mismos tienen la posibilidad de consultar mediante dispositivos portables (móviles, tabletas y otros) en el aula la experiencia anterior no representaba desafíos cognitivos suficientes. Por tanto, se plantea la necesidad de contar con nuevos instrumentos, lógicas y enfoques.

Evaluando críticamente tales planteamientos es posible apreciar modificación de la práctica con respecto a la metodología no se restringe a la mera formalidad, ya que se replantea la actividad cognoscitiva, modificando la manera regulada en que se desarrollaba la experiencia.

Además, el ABP enfatiza su aporte a los diferentes estilos de aprendizaje y la promoción de la investigación autónoma, dado que la asignatura pretende conseguir una concepción introductoria, no obstante, permite otra instancia de indagación.

Se considera de importancia a su vez, profundizar en la temática del ABP en su praxis, pues esta es una experiencia que, aunque nos da luces sobre lo estudiado, generalizar el éxito en los resultados dependerá de su adecuada implementación y del perfil del alumnado.

5. Conclusiones y trabajos futuros

La experiencia trata de un cambio metodológico en el sentido de modificar las clásicas prácticas transmisivas, permitiendo que sean los alumnos, partícipes de su propio proceso de aprendizaje.

La inclusión de los materiales promueve la participación activa e intentan satisfacer las expectativas de los alumnos mediante la manipulación, motivarlos en indagación acerca de las técnicas emergentes que aparentan complejidad pero que están al alcance del estudiantado, para su avance académico-profesional. En relación

a ello se prevé seguir trabajando en la línea producción de materiales de esta índole y su integración, para alcanzar mayor impacto.

Desde el punto de vista de la evaluación de la propuesta, tal como se menciona, los docentes pasan a tener un rol de facilitador, asimismo se considera de relevancia, evaluar sus diferentes dimensiones (grado de consecución de los objetivos, organización, tiempos, etc..) para su retroalimentación.

6. Referencias Bibliográficas

1. Manovich, L. (2001). *The language of new media*. MIT press.
2. Levis, D. (1999). *La pantalla ubicua. Comunicación en la sociedad digital*. Ciccus. Buenos Aires. La Crujía.
3. Levis, D. (2001). *Arte y computadoras: del pigmento al bit (Vol. 14)*. Editorial Norma.
4. Fernández, M. G., Barrios, W. G., Godoy Guglielmone, M. V., & Gendin, G. (2013). *Arte y TIC: Experiencias iniciales con herramientas de software en la formación de Licenciados en Artes Combinadas*. In XVIII Congreso Argentino de Ciencias de la Computación.
5. Morales Vallejo, P. (2010). *Investigación e innovación educativa*. REICE. Revista Electrónica Iberoamericana sobre Calidad, Eficacia y Cambio en Educación.
6. Mon, A., Del Giorgio, H., & Donadello, B. (2015). *Estrategias didácticas Innovadoras en la enseñanza de TICs para ingeniería en informática*. In XXI Congreso Argentino de Ciencias de la Computación (Junín, 2015).
7. Dewey, J. (1997). *How we think*. Courier Corporation.
8. Ginsburg, H. P., & Opper, S. (1988). *Piaget's theory of intellectual development*. Prentice-Hall, Inc.
9. Vygotsky, L. S. (1962). *Language and thought*. Massachusetts Institute of Technology Press, Ontario, Canada.
10. Bell, T., Urhahne, D., Schanze, S., & Ploetzner, R. (2010). *Collaborative inquiry learning: Models, tools, and challenges*. *International journal of science education*, 32(3), 349-377.
11. Wilhelm, J. D., & Wilhelm, P. J. (2010). *Inquiring minds learn to read, write, and think: Reaching all learners through inquiry*. *Middle school journal*, 41(5), 39-46.
12. Barron, B., & Darling-Hammond, L. (2008). *Teaching for Meaningful Learning: A Review of Research on Inquiry-Based and Cooperative Learning*. Book Excerpt. George Lucas Educational Foundation.
13. Savery, J. R. (2015). *Overview of problem-based learning: Definitions and distinctions*. *Essential readings in problem-based learning: Exploring and extending the legacy of Howard S. Barrows*, 5-15.
14. Esteban Guitart, M. (2011). *Del "aprendizaje basado en problemas"(ABP) al "aprendizaje basado en la acción"(ABA). Claves para su complementariedad e implementación*. REDU. *Revista de Docencia Universitaria*, 9(1), 91.
15. Garza, E. (2000). *Las técnicas didácticas en el modelo educativo del Tec de Monterrey*. Col. Tecnológico, Monterrey. En http://sitios.itesm.mx/va/dide/docs_internos/inf-doc/tecnicas-modelo.PDF, [Consulta: 25-03-2016].
16. De Pedro Carracedo, J., & Méndez, C. L. M. (2012). *Realidad Aumentada: Una Alternativa Metodológica en la Educación Primaria Nicaragüense*. *IEEE-RITA*, 7(2), 102-108.
17. Cabero, J. y Barroso, J. (2016). *Posibilidades educativas de la Realidad Aumentada*. *NAER. New Approaches in Educational Research*, 5, 1, 46-52.
18. Fundación Telefónica 2011, "Realidad Aumentada: una nueva lente para ver el mundo", Editorial Ariel, enero de 2011.

19. Caspa, E. Rivera, LB Quispe De la Cruz, and CA Montalvo Yarnold. "Realidad aumentada e inteligencias múltiples en el aprendizaje de matemáticas." Concurso de Proyectos Feria Tecnológica IEEE INTERCON (2011).
20. Mincyt, (2016). La impresión 3D, herramienta clave para impulsar la innovación tecnológica. Secretaría de Planeamiento y Políticas en Ciencia, Tecnología e Innovación Productiva.
21. New Media Consortium. (2014). NMC Horizon Report 2014 Higher Education Edition.
22. Barrón, A. (1993). Aprendizaje por descubrimiento: principios y aplicaciones inadecuadas. Enseñanza de las Ciencias, 11(1), 3-11.
23. Schmidt, H., & van der Molen, H. (2001). Self-reported competency ratings of graduates of a problem-based medical curriculum. Academic Medicine, 76(5), 466-468.
24. Morales, P., & Landa, L. (2004). Aprendizaje Basado en problemas. Theoria (13), 145-157.
25. Burbules, Nicholas C.: El aprendizaje ubicuo y el futuro de la enseñanza, Encounters/Encuentros/Rencontres on Education, Vol. 13, 2012, pág 3 a 14. (2012).

Construyendo Bases para la Formación Profesional en las Carreras de Ingeniería y Sistemas de la UNRN

Edith Lovos¹, Martín Goin¹,

¹ Universidad Nacional de Río Negro, Centro de Investigación CIEDIS
8500 Viedma Río Negro, Argentina
{elovos,mgoin}@unrn.edu.ar

Abstract. En este artículo se presenta un plan de trabajo para el desarrollo de propuestas de enseñanza y aprendizaje en modalidad virtual. Las mismas tienen como destinatarios a los alumnos de nivel medio de la zona de influencia de dos de las Sedes de la UNRN, que manifiesten intenciones de continuar estudios superiores en las carreras de ingeniería y licenciatura en sistemas. Se exponen algunas referencias teóricas que sostienen la propuesta, y se presenta el plan de trabajo propuesto. Así mismo, se presentan algunos resultados alcanzados con una primera implementación y acciones futuras.

Keywords: Ingresantes, e-learning, ingeniería, informática.

1 Introducción

El programa de ingreso a la Universidad Nacional de Río Negro (UNRN), se aplica todas las carreras de formación de grado. Y se basa en la exigencia de aprobación de dos asignaturas extracurriculares, a saber: Razonamiento y Resolución de Problemas (RRP) e Introducción a la Lectura y Escritura Académica (ILEA) durante el primer cuatrimestre del primer año. A través de estas asignaturas, se busca que los alumnos ingresantes completen una formación básica que les permita avanzar hacia niveles crecientes de complejidad y creatividad en el pensamiento de las diferentes áreas de conocimiento. Los alumnos ingresantes a la UNRN, cuentan con la posibilidad de rendir los exámenes de ambas asignaturas de manera voluntaria y libre durante fechas especiales en los meses de Diciembre y Febrero y /o cursarlas en forma presencial durante el primer cuatrimestre. En caso de no aprobar o no rendir estos exámenes voluntarios, el estudiante debe regularizarlas con el examen final correspondiente o promocionarlas.

En el caso de las carreras de ingeniería y sistemas de la UNRN, el manejo de los contenidos que se abordan en estas asignaturas, por parte de los alumnos ingresantes, resultan de gran soporte para las materias vinculadas a la programación de los primeros años.

Con la intención de lograr procesos de enseñanza-aprendizaje que permitan un mejor acompañamiento a los alumnos ingresantes a las carreras antes mencionadas, se

vienen llevando adelante diversas innovaciones docentes [1,2,3,4,5] en las materias específicas de programación de primer año. Sin embargo como docentes de estos cursos, nos interrogamos respecto si una instancia previa inmersión en los procesos de resolución de problemas matemáticos y en el *arte de programar*¹, que brinde a los futuros estudiantes una base de preparación para las materias vinculadas a la programación que se dictan en los primeros años, puede ayudar a sostener a los estudiantes durante el cursado y a mejorar el rendimiento académico. Todo esto, en el marco de una propuesta que contemple un formato accesible a todos los estudiantes, atendiendo particularmente a la geografía de la provincia de Río Negro, en la que está inserta la UNRN.

El presente trabajo indaga acerca de los beneficios que aportarían estos cursos, en un contexto de articulación entre nivel medio y universidad y llevado a la práctica en modalidad e-learning. Se presentan además algunos resultados obtenidos en una primera experiencia desarrollada en el contexto de la Sede Andina de la UNRN .

1.1 Motivación

La enseñanza de la programación es una base fundamental y uno de los primeros cursos que deben tomar los alumnos ingresantes a la carreras vinculadas a la informática. Sin embargo, la enseñanza y el aprendizaje de la disciplina es una actividad intelectual compleja y dificultosa, tanto para estudiantes como para docentes; sobretodo cuando su impacto resulta de suma importancia en la mayoría de las asignaturas sucesivas y en el campo profesional del futuro egresado. [6,7,8]. Sin embargo, esta situación no se acota a las carreras específicas de informáticas, sino que también se hace presente en cursos de programación para otras disciplinas, como es el caso de las ingenierías. Entre los problemas específicos, se destacan el no poder aplicar conceptos básicos y el diseño de algoritmos, sin importar cual sea el paradigma de programación utilizado. [9]

En el caso particular de las carreras de la UNRN, que incluyen en sus trayectos de formación inicial, cursos específicos de programación, se ha observado un desgranamiento que coincide con otros estudios [10,11,12] que indican que los estudiantes manifiestan arribar a la universidad con una preparación insuficiente, no contar con hábitos de estudios, entre ellos el análisis y comprensión de textos. En [11] se señalan algunas de las características presentes en los estudiantes de cursos introductorios de programación, que traen dificultades en el aprendizaje y la concreción de los objetivos de estos cursos, a saber: falta de entrenamiento en pensar y expresar ideas, dificultad para aprender temas básicos que requieran algún grado de abstracción, falta de orientación vocacional adecuada, gran disparidad de

¹ Donald Knuth. The Art of Computer Programming

conocimientos y formación previa, escasa valoración por el trabajo sistemático, y cierta metodología de “prueba y error” asociada al uso de las computadoras.

Por otra parte, y teniendo en cuenta la disminución del interés que existe por parte de los futuros estudiantes universitarios en las carreras vinculadas a la ciencia y tecnología, resulta importante diseñar estrategias de ingreso a los estudios superiores, que posibiliten ayudar a los alumnos a reafirmar su orientación vocacional y a estimularlos en forma positiva para avanzar en la educación superior.[8,13]. Las mismas deben tener en cuenta no solo los contenidos académicos específicos, sino también las características de los alumnos actuales: acceden y manejan diversos dispositivos tecnológicos, gustan de estar conectados en forma permanente, de la comunicación a través de las redes sociales y tienen una participación activa a través de los medios digitales traspasando las barreras de espacio y tiempo [14]

1.2 Espacios Virtuales de Enseñanza y Aprendizaje

El avance de las Tecnologías de la Información y la Comunicación (TIC), ha producido cambios sociales que afectan a la educación, modificando el perfil de los estudiantes, en particular sus formas de comunicación y estudio. Es así que surgen nuevas formas de enseñanza y aprendizaje, entre ellas, el e-learning que posibilita el desarrollo de la formación a distancia apoyada en recursos tecnológicos, dónde la interacción y la comunicación son parte fundamental del modelo. Entre sus ventajas más destacadas se encuentran: flexibilización en el acceso a la información, facilita la autonomía del estudiante, el trabajo en equipos y la colaboración entre otros. Sin embargo, también requiere más tiempo del docente, y de habilidades de aprendizaje autónomo por parte del estudiante. Cabero [15] sostiene que el éxito de propuestas formativas en esta modalidad, depende del sentido de comunidad que se logre crear, ya que este ayudará a lidiar con una de las variables a considerar que es el sentimiento de soledad de los alumnos. Y en este sentido, remarca la necesidad de diseñar e-actividades que promuevan la participación activa de los alumnos.

La estrategia formativa e-learning ha sido acompañado por la evolución tecnológica, pasando por diferentes modelos: centrado en los materiales, centrado en el aula virtual, centrado en la flexibilidad y participación. [16] Las plataformas educativas o entornos de enseñanza y aprendizaje (EVEA) son aplicaciones web que combinan herramientas para la enseñanza y aprendizaje en línea sea éste, exclusivamente no presencial (e-learning) y/o en forma combinada (b-learning). Luego de acuerdo a su funcionalidad se pueden clasificar como: de carácter específico o general. Éstas últimas, no están pensadas para una materia específica, o para desarrollar y/o adquirir determinadas habilidades. Ejemplos de plataformas educativas de propósito general

son los denominados Sistemas de Gestión de Aprendizaje (LMS) como es el caso de la plataforma de código abierto Moodle. Un LMS, debe proveer herramientas para las siguientes funcionalidades: (i) la administración del Entorno de Aprendizaje, (ii) la comunicación de los participantes; (iii) la gestión de contenidos; (iv) la gestión del trabajo en grupos, y (v) la evaluación.[17]

Existen variadas experiencias y estudios en universidades nacionales, en relación al uso de estrategias de e-learning en cursos introductorios al nivel superior , como son los casos de: el programa UBA XXI (Universidad de Buenos Aires), la UNNE (Universidad Nacional del Nordeste), la UNLP (Universidad Nacional de La Plata) y la UNSA (Universidad Nacional de Salta) [13,18,23,24], lo cual nos anima a pensar una propuesta de este tipo en el contexto de la UNRN, que posibilite la construcción de conocimiento de base para el desarrollo de una carrera universitaria vinculada a la ingeniería y/o informática, y la construcción de una comunidad académica activa haciendo frente a las distancias físicas que separan a las tres sedes de la Universidad.

2 Plan de Trabajo Propuesto

La UNRN, es una universidad que cuenta con menos de 10 años de creación y se divide en 3 sedes académicas: Atlántica, Andina y Valle Medio y Alto, y un rectorado. De esta forma, sus propuestas de formación se extienden al territorio de la provincia de Río Negro y a la región norte de la Patagonia Argentina. Así, las intervenciones pedagógicas que se comentan más adelante, están destinadas a los alumnos de nivel medio de la zona de influencia de la UNRN, que cursan el último año del secundario, en particular aquellos que manifiesten una orientación hacia las carreras vinculadas a la ingeniería y/o informática.

El plan de trabajo propuesto para estos procesos de formación se conforma de 6 etapas de trabajo, a saber:

1. Selección de Muestra: se establece contacto con los directivos de todas las escuelas de nivel medio (CEM) de la zona de influencia de UNRN, ya sea vía mail o telefónicamente, para informar acerca del proyecto. Dejando a su cargo, en los siguientes 15 días, el sondeo de los posibles estudiantes interesados en participar del curso virtual. Según los resultados obtenidos, se realizará un cronograma de encuentros (presencial o via videoconferencia) con aquellos CEM con estudiantes interesados en la propuesta, para dar información sobre los contenidos del curso y la metodología de trabajo .

2. Diagnóstico Inicial. Se propone el desarrollo de un cuestionario inicial (CI). A través de este, se busca recolectar información de los estudiantes respecto a las condiciones de accesibilidad, los usos y competencias más relevantes, aplicaciones de software y conocimientos previos en relación a la programación. De esta forma se analizará el grado de familiaridad con diversos aspectos relacionados al uso de las TIC dentro del ámbito educativo. El cuestionario será de tipo estructurado, conformado por preguntas en su mayoría cerradas con variables discretas y en otros casos de escala.
3. Diseño y Desarrollo de Plataforma Virtual. Selección de la plataforma virtual que permita crear un aula y dar soporte para que los alumnos participantes accedan a los materiales del curso (textos, videoclases, software, links recomendables, etc), realicen consultas al equipo docente (en forma sincrónica y asincrónica) e interactúen con sus pares. Los contenidos académicos dentro del aula virtual, se organizan por unidades temáticas y en forma semanal se actualizará el cronograma de actividades.
4. Formación y Seguimiento. Durante el desarrollo del curso, se realizará un seguimiento de los alumnos con el objeto de registrar las actitudes y aptitudes alcanzadas. Para ello, se administrarán guías prácticas individuales y otras para el trabajo en equipos colaborativos que permitan aplicar los conceptos trabajados en cada unidad temática. Estas últimas serán evaluaciones sumativas y de entrega obligatoria, con valoración numérica, tomando en cuenta los siguientes criterios: claridad en la resolución de problemas, análisis significativo de la información, capacidad de integrar conceptos, interpretación de los resultados obtenidos, estrategias para la resolución de problemas y responsabilidad frente al trabajo para la entrega en tiempo y forma. Durante el desarrollo de los mismos, los alumnos contarán con la asistencia del tutor.
5. Evaluación y Cierre. Como cierre se propone que los alumnos desarrollen un examen optativo en forma individual. El propósito es ofrecerles una instancia de evaluación respecto a la aplicación de los conceptos trabajados en el curso. Así mismo, se aplicará una encuesta que busca conocer las valoraciones que hacen los alumnos respecto al desarrollo, organización del curso, acompañamiento y motivación del tutor, funcionamiento de la plataforma, desempeño en el aprendizaje y sugerencias para la mejora de la propuesta.

6. Análisis y Documentación. Con los datos de la etapa anterior, se registrarán e instrumentarán formas y técnicas de análisis, permitiendo así un mejor entendimiento del objeto en estudio.

3 Resultados

3.1 Caso RRP en modalidad e-learning (E-RRP)

En el año 2013, en la Sede Andina de la UNRN, se llevó adelante una implementación del plan de trabajo descrito en el apartado anterior, que permitió trabajar los contenidos de la asignatura RRP en modalidad e-learning (E-RRP). Del mismo participaron alumnos de 9 establecimientos de nivel medio de San Carlos de Bariloche y Pilcaniyeu, que manifestaron interés vocacional por las carreras de ingeniería de la Sede Andina. De allí surgieron 49 candidatos, de los cuales solo 24 participaron del curso y 15 de ellos finalizaron el mismo. El equipo docente del curso estuvo conformado por un docente. La duración total del curso fue de 4 meses.

Los contenidos que se abordaron en el curso son los mismos tanto para la modalidad presencial como virtual, a saber: Conjuntos Numéricos, Operaciones aritméticas combinadas, Ecuaciones e Inecuaciones, Polinomios, Factorización, Funciones lineales y cuadráticas.

Para llevar adelante la propuesta se utilizó el sistema de gestión online de cátedras/materias ECATHS[19], que aunque no cumple con todas las funcionalidades requeridas para un LMS, ofrece un espacio virtual de interacción y construcción de conocimiento individual y colectivo; sin necesidad de instalar ningún software, pues funciona en la nube. Soporta textos de todos los formatos, videos e imágenes, además es muy fácil su configuración y gestión y muy rápida la carga de datos. Es una herramienta Beta, con lo cual está en desarrollo continuo, es un producto argentino y de uso libre y gratuito.

A través de la fase de análisis se realizó un estudio comparativo de los resultados obtenidos en el examen voluntario de RRP entre ingresantes que participaron del curso RRP en modalidad e-learning denominado “Grupo experimental” y aquellos que no lo hicieron llamado “Grupo de control”, ambos recientemente egresados del nivel medio. Además se realizó y aplicó un mecanismo de seguimiento de cada estudiante y de monitoreo de la experiencia empírica desde el inicio de la experiencia hasta el examen de ingreso a la Universidad. De un total de 33 estudiantes, de los cuales 13 pertenecen al grupo experimental mientras que 20 al grupo de control, se pudo observar que en el primer grupo aprobaron las evaluaciones de RRP un 77% mientras que en el segundo solo un 15%.

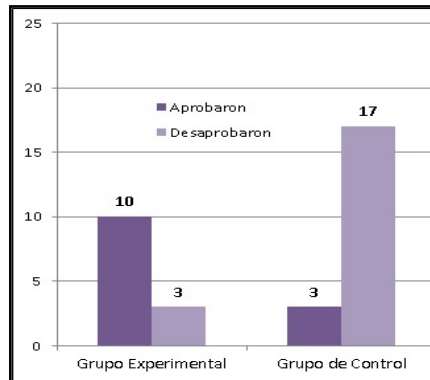


Fig.1. Comparación entre grupos según resultados del examen final de RRP.

Así mismo, respecto a las notas obtenidas en los exámenes, se pudo observar que quienes participaron del curso E-RRP y desaprobaron el examen lo hicieron con mejores notas en promedio, como se puede apreciar en el gráfico de la figura 2. Para tener una idea, la calificación 4 corresponde al 60% del examen con respuestas correctas.

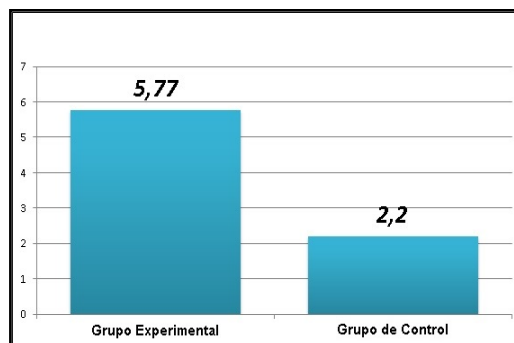


Fig.2. Promedio de nota de aprobados y desaprobados de cada grupo en RRP

En relación a la deserción que se produjo durante la experiencia, los mayores niveles se registraron en el primer mes y la tasa mayor de inactividad se dió en el último mes, cuando la mayoría de los alumnos están realizando su viaje de estudios/egresados.

A través del análisis del cuestionario inicial, se observó que los alumnos presentan un alto nivel de acceso a la tecnología lo cual les permitió adaptarse rápidamente al modelo de enseñanza y aprendizaje en forma no-presencial. En relación a la fase de seguimiento, se pudo observar que el inconveniente más notorio en gran parte de los estudiantes, estuvo asociado al desarrollo de las actividades grupales. Esto no solo se reflejó en los resultados de las mismas, sino también en las respuestas obtenidas en el cuestionario que completaron al finalizar el curso.

Entre los aspectos positivos de ésta experiencia cabe destacar las opiniones de los alumnos resaltando la inclusión de las TIC en el proceso de aprendizaje, la interacción con el tutor y el uso de la plataforma virtual, en decires de los alumnos: *“Esta fue una buena experiencia de aprendizaje”*, *“Me sirvió para prepararme como alumno/a universitario/a”* y *“Conocí y aprendí temas nuevos en Matemática”*.

3.2 Caso E-RPA

El segundo caso denominado E-RPA es una propuesta para trabajar contenidos específicos de resolución de problemas algorítmicos: ciclo de resolución de problemas, algoritmo, datos y tipos de datos, representación de las soluciones algorítmicas y estructuras de control. Está previsto que se desarrolle durante tres meses, entre Agosto y Noviembre de 2016 con alumnos de nivel medio de los CEM ubicados en la zona de influencia de las Sede Andina y Atlántica de la UNRN. Como plataforma educativa se ha seleccionado Moodle, ya que las carreras de la Sede Atlántica cuentan desde el año 2009, con un espacio de apoyo a las asignaturas en la misma. Así el uso de este recurso, permitirá realizar un seguimiento posterior de los alumnos que finalmente inicien su formación profesional en alguna de las carreras antes mencionadas. Por otra parte, Moodle permite ampliar su funcionalidad, a través del uso de pluggins y la inclusión de materiales digitales desarrollados con otros aplicativos.

En relación a la cantidad de alumnos, se propone un máximo de 25 participantes y un equipo docente conformado dos docentes: uno de la asignatura informática de las carreras de ingeniería (Sede Andina) y otro de las materias de programación de la Lic. en Sistemas (Sede Atlántica).

Se propone el uso del software FreeDFD[19] para el desarrollo de los algoritmos, posibilitando la representación gráfica, la ejecución y depuración de las soluciones algorítmicas. De esta forma, se busca focalizar en el desarrollo de habilidades para la resolución de problemas y la forma de pensar algorítmicamente evitando tratar con la sintaxis de un lenguaje específico [22]. FreeDFD es de uso libre, cuenta con una interfaz intuitiva y estar disponible en español. Así mismo se puede utilizar tanto en Windows como en Linux y en este sentido es compatible con los SO de las netbooks que reciben los alumnos de los CEM a través del programa Nacional Conectar Igualdad.

Actualmente se está llevando adelante la primer etapa del proyecto, y hasta aquí un total de 27 estudiantes de nivel medio han manifestado interés en la propuesta. De estos, el 11% pertenecen a la localidad de Viedma, el 4% a Carmen de Patagones, el 4% de El Bolsón y el 81% restante corresponden a Bariloche. De los cuales el 45% llevan adelante sus estudios en escuelas técnicas. Hasta aquí, del total de estudiantes interesados, el 41% estudia en una escuela de gestión privada.

Se espera que al finalizar el curso los alumnos puedan:

- explorar, expresar simbólicamente, implementar en pseudocódigo y evaluar diferentes algoritmos con tipos de datos simples.
- desarrollar habilidades para el trabajo en equipo en un entorno no presencial

4 Conclusiones

La primer experiencia, RRP en modalidad e-learning (E-RRP), dió lugar a un proceso de enseñanza aprendizaje, que fomentó en los alumnos la autonomía (característica del “oficio del alumno universitario” necesario para el éxito académico), el trabajo en equipo, el dinamismo y la flexibilidad en el aprendizaje. Así mismo, la experiencia generó un acercamiento entre las escuelas medias y la Sede Andina de la UNRN promoviendo el interés por carreras científico tecnológicas de la Sede.

Los resultados de esta primera experiencia, permiten a considerar a experiencias de este tipo, como estrategias para ayudar a los alumnos que participen de las mismas, a desarrollar habilidades que les permitan una mejor inserción y rendimiento académico en las cursadas de las primeras materias vinculadas a la resolución de problemas.

En relación a la implementación de E-RPA, y teniendo en cuenta la velocidad con la que se suceden los avances tecnológicos, que impactan en los procesos sociales, el pensamiento computacional, como señala Simari [21]: “*se convierte en un componente irremplazable de las habilidades necesarias para comprender y desempeñarse en el mundo*”. En este sentido, una propuesta de inmersión en el proceso de resolución de problemas con algoritmos, resultará beneficiosa, más allá del interés vocacional por una determinada carrera.

Referencias

1. Goin Martín. (2015). Estrategias didácticas en Entornos Virtuales para capacitación en Matemática como Articulación entre la Escuela Media y la Universidad Nacional de Río Negro. Tesis Maestría en Educación en Entornos Virtuales. Universidad Nacional de la Patagonia Austral (UNPA).
2. Goin Martín. (2016). Caminando Junto al Lenguaje C. Lecturas de Cátedra. UNRN. Viedma Río Negro
3. Lovos, E., Gonzalez, A., Mouján, I., Bertone, R., & Madoz, C. (2012). Estrategias de enseñanza colaborativa para un curso de Programación de primer año de la Lic. Sistemas. XVIII Congreso Argentino de Ciencias de la Computación. Universidad Nacional del Sur. Bahía Blanca
4. Lovos, Edith. (2014). El uso de estrategias colaborativas mediadas por tecnología. La enseñanza de la programación en el primer año de la Licenciatura en Sistemas de la UNRN. Tesis de Maestría. Facultad de Informática, UNLP.

5. Lovos, E, Gibelli, T, Bertone R. (2015). Programación Estructurada en un Curso Introductorio. Una Experiencia Explorando Python. XXI Congreso Argentino de Ciencias de la Computación. Universidad Nacional del Nordeste.
6. Costelloe, E. (2001). Teaching Programming. The State of the Art. Department of Computing, Institute of Technology Tallaght, Dublin 24. CRITE Technical Report, 2004a.
7. Lahtinen E, Ala-Mutka K, et al.(2005) A Study of the Difficulties of Novice Programmers. 10Th annual SIGCSE conference on Innovation an technology in computer science education ItiCSE '05
8. Xinogalos; Stelios 2014)
9. Muñoz, R., Barriá, M., Noël, R., Providel, E., & Quiroz, P. (2012). Determinando las Dificultades en el Aprendizaje de la Primera Asignatura de Programación en Estudiantes de Ingeniería Civil Informática. *Agosto del*.
10. García, J.C.,González, M.L.,Zanfrillo, A.I. (2011). Desgranamiento universitario: perspectiva estudiantil en Ingeniería. En XI Coloquio Internacional sobre Gestión Universitaria en América del Sur. Florianópolis. Septiembre 2011. <http://repositorio.ufsc.br/xmlui/handle/123456789/259>
11. Cernadas, María A., Cuenca Pletsch, Liliana R. (2009). Las TIC como medio para evitar el desgranamiento temprano. En IV Congreso de Tecnología en Educación y Educación en Tecnología. <http://sedici.unlp.edu.ar/handle/10915/1900>
12. De Giusti, A. E., Madoz, M. C., Gorga, G., Feierherd, G. E., & Depetris, B. O. (2003). Enfoques y herramientas en la enseñanza de un primer curso de computación (CS1). En IX Congreso Argentino de Ciencias de la Computación
13. Dapozo, G. N., & Pelozo, S. S. (2009). Implementación del módulo Informática para los ingresantes a la carrera Licenciatura en Sistemas de Información de la UNNE. *TE & ET*.
14. Mariño Drews Olga (2013). Fortalecimiento de la enseñanza de la ingeniería con las tecnologías de información y comunicaciones. En Revista de Ingeniería. Universidad de los Andes. Colombia. Nro 39 ISSN.0121-4993. Julio-Diciembre 2013. pp.46-49
15. Cabero-Almenara, J. (2006). Bases pedagógicas del e-learning. *RUSC. Universities and knowledge society journal*, 3(1), 1.
16. Gros Salvat, Begoña. (2011). *Evolución y retos de la educación virtual: construyendo el e-learning del siglo XXI* (Vol. 3). Editorial UOC.
17. Fernández-Pampillón Cesteros, A. (2009). Las plataformas e-learning para la enseñanza y el aprendizaje universitario en Internet.
18. De Giusti, A. E., Sanz, C. V., Madoz, M. C., Gorga, G., & Gonzalez, A. H. (2013). Experiencias y estrategias de incorporación de tecnologías informáticas en el ámbito educativo. In *I Jornadas de Intercambio de Experiencias*.
19. Moodle. <https://moodle.org/?lang=es>
20. ECATHS. www.ecaths.com
21. Simari, G. R. (2013). Los fundamentos computacionales como parte de las ciencias básicas en las terminales de la disciplina Informática. In *VIII Congreso de Tecnología en Educación y Educación en Tecnología..*
22. Xinogalos, S. (2013). Using Flowchart-based Programming Environments for Simplifying Programming and Software Engineering Processes. In Proceedings of 4 th IEEE EDUCON Conference, Berlin, Germany, 13-15 March 2013, IEEE Press, pp. 1313-1322.
23. Mac Gaul de Jorge, M., López, M. F., & Fernández, E. F. (2008). Evaluación en entorno virtual para el ingreso masivo universitario. In *XIV Congreso Argentino de Ciencias de la Computación*.
24. <http://www.uba.ar/academicos/uba21/contenidos.php?id=12&n1=1>

Introduciendo conceptos de Cloud Computing utilizando el entorno CMRE

Laura De Giusti¹, Franco Chichizola¹, Sebastián Rodríguez Eguren¹, Mariano Sánchez¹, Juan Manuel Paniego¹, Armando De Giusti^{1,2}

¹Instituto de Investigación en Informática LIDI (III-LIDI) – Facultad de Informática –UNLP

² Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

Argentina

{ldgiusti, francoch, seguren, msanchez, jmpaniego, degiusti}@lidi.info.unlp.edu.ar

Abstract. Se analiza el modo de incorporar conceptos básicos de computación en la nube (“cloud computing”) en etapas tempranas del aprendizaje de los alumnos, empleando el entorno multi-robot CMRE que se ha utilizado para introducir conceptos de concurrencia y paralelismo en un curso CS1.

El entorno CMRE (Concurrent Multi Robot Environment) se extiende para poder considerar la nube como un robot externo, con capacidades diferenciales y posibilidad de comunicarse en forma independiente con cada robot del entorno virtual y o real, de modo de explicitar al alumno los conceptos básicos de los servicios en la nube, sin cambiar el entorno simple en el que trabaja.

En el trabajo se analizan algunos casos ejemplo para mostrar el modo de presentar los conceptos para el alumno y relacionar las aplicaciones con actividades colaborativas de los robots para las cuales se requiere interacción con la nube.

Keywords: Concurrencia, Paralelismo, Algoritmos Paralelos, Cloud Computing.

1 Introducción

Las currículas informáticas clásicas [ACM68][ACM78][ACM99] contenían conceptos de concurrencia en diferentes áreas (Lenguajes, Paradigmas, Sistemas Operativos) pero omitían casi totalmente el tratamiento del paralelismo, salvo al plantear los conceptos de sistemas distribuidos.

Las actuales arquitecturas multiprocesador han producido un notorio impacto, obligando a replantear el “modelo base” para la enseñanza conceptual de las arquitecturas y de la programación. Reemplazar la “máquina de Von Neuman” con un solo hilo de control, por un esquema que integra múltiples procesadores cada uno con uno o más hilos de control y varios niveles de memoria accesible en forma diferenciada [GEP06][MCC08] conlleva también el necesario aprendizaje de la *programación paralela* sobre las arquitecturas actuales [ACM13]. Sin embargo, la programación paralela (y los conceptos fundamentales de concurrencia) resulta más compleja para un alumno en las etapas iniciales de su formación, y es necesario contar con nuevas estrategias que permitan abordarla exitosamente [DEG15].

El alumno actual de un curso CS1 está “entrenado” en elementos informáticos de alta interacción tales como consolas y juegos electrónicos, celulares, tablets y

computadoras. Esta familiarización favorece la tendencia al empleo de herramientas interactivas para la enseñanza de conceptos fundamentales de programación. Tendencia que se fortalece con elementos tales como la programación de robots [MIN15] a alumnos desde la escuela media y en los cursos universitarios iniciales [ACM04][ACM08][ACM13][DEG13].

En esta línea, la posibilidad de dar los primeros pasos en el mundo de la programación mediante un entorno gráfico e interactivo, coordinado con demostraciones en el mundo real con robots permite reducir la brecha que tradicionalmente existió entre la abstracción y la posibilidad de “ver” y “verificar” la ejecución de algoritmos. [AMD09][HOO13].

El entorno gráfico CMRE, donde se cuenta con un conjunto de robots que se mueven en una ciudad, ha permitido incorporar la enseñanza de los conceptos básicos de concurrencia y paralelismo en un curso inicial de Informática. En sucesivos trabajos ha sido presentado a la comunidad universitaria y está siendo empleado desde hace 3 años en la UNLP.

Asimismo los avances en las tecnologías de virtualización y cómputo distribuido han dado origen al paradigma de Cloud Computing, que se presenta como una alternativa a los tradicionales sistemas de Clusters y Grids para ambientes de cómputo paralelo [ROD07][DEG07] [BER08][KON09].

Las arquitecturas tipo “*Cloud*” se presentan como una evolución natural del concepto de *Clusters* y *Grids*, integrando grandes conjuntos de recursos virtuales (hardware, plataformas de desarrollo y/o servicios), fácilmente accesibles y utilizables por usuarios distribuidos, vía WEB [MIL08][VAQ09][FOS10]. Estos recursos pueden ser dinámicamente reconfigurados para adaptarse a una carga variable, permitiendo optimizar su uso. Al enfocarnos en Cloud Computing aparecen problemas clásicos de la Ciencia Informática, extendidos para este nuevo modelo de arquitectura: planificación, virtualización, asignación dinámica de recursos, migración de datos y procesos [ARD09][VAZ09].

El desafío de iniciar a los alumnos en la conceptualización del modelo de “*Cloud Computing*” a partir del primer año de sus estudios ha sido la motivación de este trabajo, en el que se trata de integrar al entorno CMRE algunas funcionalidades para comprender conceptualmente lo que el alumno puede esperar de la computación en la nube.

El trabajo se ha estructurado lo siguiente: la Sección 2 presenta el entorno CMRE actual. En la Sección 3 se abordan los aspectos centrales de Cloud Computing, mientras la Sección 4 se centra en las extensiones del entorno CMRE para considerar el rol del Cloud y la programación de aplicaciones que se vinculan con la nube. En la Sección 5 se detallan resultados alcanzados, conclusiones y líneas de trabajo futuro.

2 El entorno CMRE actual

Las características principales del entorno CMRE pueden resumirse de la siguiente manera [GEP06][DEG14]:

- Existen múltiples procesadores (robots) que realizan tareas y que pueden cooperar y/o competir. Los mismos representan los “cores” de una arquitectura

multiprocesador real. Estos robots virtuales pueden tener un reloj propio y diferentes tiempos para la ejecución de sus tareas específicas.

- El modelo de ambiente (“ciudad”) en la que desarrollan sus tareas admite áreas privadas (sólo puede ser recorrida por un robot de los declarados en el programa), parcialmente compartidas (un conjunto de robots pueden recorrerlas) y totalmente compartidas (todos los robots declarados pueden recorrerla).
- Si se instancia a un sólo robot en un área que abarque toda la ciudad, se repite el esquema del Visual Da Vinci [DEG13].
- Cuando dos o más robots están en un área compartida (parcial o totalmente), compiten por el acceso a las esquinas del recorrido y a los recursos que allí existan. Para esto deben sincronizar.
- Cuando dos o más robots (en un área común o no) desean intercambiar información (datos o control) deben hacerlo por mensajes explícitos.
- La sincronización se da por un mecanismo equivalente a un semáforo binario.
- La exclusión mutua puede generarse con la declaración de las áreas alcanzadas por cada robot. Acceder a otras áreas de la ciudad no está permitido.
- Todo el modelo de ejecución es sincrónico y permite la existencia de un reloj virtual de ciclos, que a su vez permite asignar tiempos específicos a las operaciones en cada robot, simulando la existencia de una arquitectura heterogénea.
- El entorno permite ejecutar el programa de manera tradicional, o paso a paso por instrucciones, dando al usuario un control detallado sobre la ejecución del programa, de manera de poder controlar situaciones típicas de concurrencia tales como conflictos (colisiones) o deadlocks, por ejemplo, se puede saber si cada robot está ejecutando o ha finalizado.
- En la ejecución paso a paso, el efecto de las operaciones se puede reflejar en los robots físicos, comunicados vía Wi-fi. Los robots físicos poseen un sistema operativo Linux que permite ejecutar un servidor http implementado en NodeJS [NJS15]. De esta manera el entorno se comunica con los robots (cada robot físico se corresponderá con uno virtual en el ambiente). La comunicación entre ellos es punto a punto, y bidireccional, es decir, el entorno envía las instrucciones al robot físico y luego este último envía su respuesta al entorno indicando la finalización de la instrucción indicada.
- Una dificultad que presenta actualmente el entorno es la de no tener una “visión global” del estado de las esquinas de la ciudad y de los robots (posición, cantidad elementos en su bolsa, etc), y por lo tanto no se puede resolver problemas de índole colaborativo o de optimización de trabajo general. Por esto se plantean algunas ideas en este trabajo y también se está experimentando con un “servidor” que contenga esta información (o la adquiera por una cámara general para los robots físicos) y se pueda comunicar con todos los robots para optimizar algoritmos.

3 Cloud Computing

3.1 Conceptos generales de Cloud

Antes de adentrarnos en la utilización de Cloud Computing con el encorno CMRE se deben mencionar algunos conceptos básicos del paradigma de Cloud Computing o Computación en la nube.

La computación en la nube es un modelo o paradigma de cómputo distribuido para la prestación y consumo de servicios de TI [VAZ09]. En los últimos años, ha ganado aceptación considerable como manera eficaz de adquirir y utilizar servicios informáticos con un costo accesible y predecible. Esto permite que el usuario final de estos recursos y servicios pueda acceder a los mismos sin necesidad de invertir grandes sumas de dinero en la adquisición de hardware para el armado y mantenimiento de un centro de cómputos el cual involucraría espacio físico, alimentación, una red para interconectar los equipos, refrigeración, almacenamiento y soporte técnico. Además el consumo sobre el cloud puede ajustarse a la demanda, no exigiendo compra de infraestructura local en función del crecimiento de nuevos servicios que se requieran [VAQ09]. En pocas palabras, Cloud Computing nos provee de recursos TI bajo demanda, a un costo proporcional al consumo de infraestructura y software requeridos.

Cuando hablamos de Cloud Computing debemos tener en cuenta que podemos elegir entre tres modelos de servicio: SaaS, PaaS e IaaS. A continuación, se mencionan las características principales de cada uno de estos modelos:

- SaaS (Software as a Service): el usuario cloud puede hacer uso de las aplicaciones que contrate al correspondiente proveedor. Un ejemplo puede ser una PyME que contrate una aplicación de correo electrónico para sus 30 empleados. La aplicación no podrá ser modificada por la PyME ni sus usuarios a excepción de posibles configuraciones de usuario o personalizaciones que le permita el proveedor. La aplicación se encontrará alojada en las infraestructuras cloud del proveedor y el usuario no tendrá ningún control sobre las mismas. Ejemplos de este tipo de servicios son: Dropbox, Gmail, Google Doc, Office 365, entre otros.
- PaaS (Platform as a Service): Cuando los recursos proporcionados por el Cloud son entornos configurados con todas las herramientas de software necesarias para el desarrollo y/o despliegue de aplicaciones personalizadas, sin que el usuario se ocupe de la instalación y administración de la infraestructura subyacente, entonces se considera un modelo PaaS. Estas plataformas ofrecen todo lo necesario para soportar los ciclos de vida completos de una aplicación y/o servicio WEB disponible en Internet. Ejemplos de este tipo de servicios son: Windows Azure, Salesforce, Dot Cloud, entre otros.
- IaaS (Infrastructures as a Service): Este modelo ofrece, por medio de la tecnología de virtualización, recursos de infraestructura virtual (servidores, equipos, dispositivos de almacenamiento, dispositivos de red y software de base), flexibles y escalables; en donde múltiples usuarios coexisten compartiendo el mismo hardware físico de forma transparente, segura e independiente. Sobre dicha infraestructura virtual alojará sus aplicaciones y plataformas; sobre estas últimas el

usuario tiene el control, pero no sobre la arquitectura física real asignada. Por Ej. Amazon Web Services, Windows Azure, Google Compute Engine, Rackspace, entre otros.

Otra característica a tener en cuenta cuando hablamos de Cloud Computing es el modelo de despliegue, del cual podemos distinguir 3 tipos:

- Cloud público: Es aquel desplegado por un proveedor de IT, que ofrece servicios (IaaS, PaaS y/o SaaS) de acceso público desde Internet. Generalmente, este modelo de despliegue se relaciona con un contexto comercial, donde los usuarios son considerados clientes y pagan por el tiempo de uso de los servicios. La principal ventaja, es la posibilidad de acceder inmediatamente a un conjunto de recursos con una mínima inversión. Las desventajas que presenta son el alto costo acumulado en el tiempo de uso del mismo y las garantías sobre la privacidad y seguridad de la información del usuario.
- Cloud privado: Se considera un Cloud privado, al que se encuentra desplegado en la intranet (en la red de datos detrás del firewall) de una organización, institución o empresa. El despliegue se efectúa sobre la infraestructura de hardware de la organización y los servicios proporcionados son de características similares al Cloud público, con la diferencia que el acceso está limitado a la red privada y no presenta carácter comercial. Las principales ventajas de un Cloud privado es la capacidad de brindar mayor eficiencia en el uso de la infraestructura física, como también un alto nivel de seguridad sobre los datos sensibles de la organización. Con el objetivo de lograr mayor eficiencia en la utilización de los recursos del datacenter, éstos se consolidan a través de la virtualización. Por otro lado, dado que el Cloud privado se encuentra desplegado en la Intranet de la organización, se garantiza la seguridad de los datos por medio de las mismas políticas y medidas de seguridad del datacenter.
- Cloud híbrido: Se denomina Cloud híbrido al despliegue de un Cloud privado que utiliza de forma segura los recursos de un Cloud público. Este modelo es de gran utilidad cuando una organización requiere aumentar rápidamente sus recursos privados para satisfacer los picos de demanda de determinados servicios. Generalmente, la organización contrata servicios de un Cloud público para expandir los recursos de su Cloud privado. A esta modalidad, también se la conoce como “Cloudbursting”. Un Cloud híbrido permite a las organizaciones obtener los beneficios combinados de un Cloud privado y uno público, logrando así el aumento de la eficiencia en el uso de la infraestructura física del datacenter, un mayor nivel de seguridad de los datos y una mayor disponibilidad de recursos para atender grandes demandas de servicio.

Un usuario puede adoptar uno o más de estos modelos según sus necesidades. La decisión vendrá condicionada por dónde desea centrar sus esfuerzos y conocimientos: en las aplicaciones, en las plataformas y/o en las infraestructuras tecnológicas. Qué elementos le aportan valor a su negocio y por lo tanto quiere seguir implicado más de cerca en su evolución y cuales no le suponen un valor diferencial y prefiere contratar a un proveedor especializado.

Si hacemos hincapié en los beneficios que nos brinda el paradigma de Cloud Computing a la hora de utilizar sus servicios, nos encontramos con la posibilidad de tener recursos disponibles bajo demanda, interfaces de administración basadas en servicios web, escalabilidad y elasticidad de recursos, aprovisionamiento automático de recursos, y autoservicio. Sin embargo, también existen riesgos, que es necesario conocer y tener en cuenta a la hora de utilizar estos servicios: seguridad física de los datos, de las conexiones a los servicios, en la autenticación y el control de acceso; dependencia absoluta de la red y del ancho de banda que se tenga, necesidad de estar siempre conectado; dependencia del proveedor, posibles fallas del cloud o las comunicaciones, aspectos legales que pueden variar de acuerdo a la ubicación geográfica de los datacenters del proveedor; integridad de los datos, posibles pérdidas, privacidad y confidencialidad y potencial acceso a datos privados sin aviso por aplicación de leyes o mandatos judiciales.

3.2 Objetivos conceptuales para el aprendizaje en un alumno de un curso CS1

La idea principal es que el alumno adquiera de manera sencilla los conceptos relacionados a Cloud Computing. En una primera instancia el énfasis estará en ver el Cloud como SaaS.

Desde este punto de vista, en el Cloud se tiene un componente externo al sistema, que en CMRE será un “super robot” que conoce el estado global de la ciudad, y brinda un conjunto de servicios para optimizar algoritmos orientados a resolver de manera colaborativa problemas que involucren más de 1 robot en la ciudad.

Teniendo en cuenta que el “super robot” se encuentra en un hardware distinto al resto de los robots, no poseen un área compartida, por lo que la comunicación es exclusivamente por medio de mensajes (lo cual responde al modelo del Cloud, con el cual nos comunicamos vía red local o internet).

4 Extensiones conceptuales del entorno CMRE para Cloud Computing

4.1 La nube como un “super robot” externo al entorno.

Conceptualmente para el alumno (que en nuestro caso cursa Taller de Programación en el segundo semestre de primer año, con 3 paradigmas: imperativo, orientado a objetos y concurrente) es relativamente simple admitir que el entorno CMRE tiene posibilidad de vincularse con un “robot externo” con otras capacidades de resolución de problemas y de comunicación de primitivas a ejecutar por múltiples robots en el entorno.

Análogamente este modelo se aplica a los robots físicos empleados en demostración, los cuales efectivamente se pueden conectar vía Internet a la nube (“super robot” del modelo) y recibir comandos y control.

Es de hacer notar que el “super robot” ve todo el escenario de la ciudad, la ubicación de objetos en las esquinas, eventuales obstáculos en los recorridos y también conoce todos los datos de los robots (por ej. velocidad de movimiento de cada uno).

Claramente sobre este “super robot” pueden correr algoritmos más complejos que los admitidos por los comandos de los robots del entorno CMRE actual.

4.2 Posibilidades de optimización de algoritmos multirrobot

Al tener una visión global del entorno (“ciudad”) se pueden plantear problemas de optimización simple, en los que vía comunicación con el “super robot” externo, cada uno de los robots del entorno realicen una determinada tarea en función de una función objetivo a maximizar.

Es interesante comprender que esto responde a uno de los paradigmas más empleados en programación concurrente/paralela que es el de “master-slave”, con la particularidad que el “master” tiene un repertorio de instrucciones extendido y la posibilidad de resolver algoritmos más complejos que cada uno de los robots “esclavos”. Además respetando la posibilidad de heterogeneidad en los robots “esclavos” para enriquecer la resolución de problemas.

4.3 Alcance de la programación por un alumno de un curso CS1

El alcance planteado para el curso 2017 es tener problemas donde la ciudad es compartida por múltiples robots y se plantea un objetivo que requiere colaboración, en el cual el modo más natural de “optimizarlo” es conocer dinámicamente la situación de toda la ciudad y todos los robots.

Esto se podría resolver con la arquitectura de CMRE actual, pero requiere una fase de intercambio de mensajes entre los robots para decidir la próxima acción de cada uno (lo cual hace más lenta la respuesta al crecer el número de robots y/o la dimensión de la ciudad). Por otro lado las decisiones seguirán siendo “sub óptimas” ya que el conocimiento disponible se limitará a la suma del alcance del conocimiento de cada robot del entorno.

El alumno de primer año *no* programará la aplicación de optimización global sobre el “super robot”, simplemente podrá pedir un servicio de software desde cada uno de los robots del entorno. Naturalmente sí se le explica la potencialidad de ese “super robot” en la nube para brindar servicios de software más complejos que los propios del CMRE básico.

4.4 Alcance de la programación por alumnos avanzados

Con el mismo planteo los alumnos avanzados (3er. año Programación Concurrente, 4to. año Sistemas Paralelos y otras asignaturas afines) pueden trabajar con la programación de los algoritmos de optimización en el “super robot” y además comprender conceptos como el control dinámico del programa que ejecuta cada robot del entorno, en función de optimizar una función objetivo.

4.5 Extensiones sobre el entorno de robots físicos

Naturalmente la vinculación de los robots físicos con la nube permite introducir los mismos conceptos, con una demostración real de la implementación de algoritmos colaborativos que involucran el conjunto de robots físicos (incluso siendo heterogéneos y respondiendo a diferentes comandos). Es interesante analizar que nuevos componentes físicos del entorno (como una o más cámaras que analizan la

escena) pueden agregarse y sus datos ser capturados y procesados en la nube, comunicando luego los comandos orientados a la optimización de la tarea global a cada uno de los robots físicos.

Esta etapa crece en potencialidad para los alumnos avanzados, que se introducen a los temas de Cloud Robotics [KEH15] que marcan una de las líneas actuales de gran interés en las aplicaciones informáticas.

4.6 Ejemplo de ejercitación experimental

A modo de ejemplo se ha trabajado con un caso muy simple en el que N robots tienen que alcanzar N objetivos en la ciudad (ej. 8 robots y 8 objetos a recoger) de modo que el número de pasos totales a realizar por el conjunto de robots sea mínimo.

Notar que este problema permite extensiones (N diferente para robots y objetivos, diferente velocidad de movimiento de los N robots, cambio de la función a optimizar, etc.).

Resulta claro que en la nube (nuestro “super robot”) con conocimiento de las N ubicaciones de los robots y los objetos, planificar un movimiento óptimo, en número de pasos por ejemplo, es relativamente sencillo y permite definir una “trayectoria” a realizar por cada robot que optimice el resultado final.

Además se pueden definir otros objetivos de interés como minimizar las instancias de sincronización entre robots (por llegar al mismo tiempo a la misma esquina por ejemplo) y permite fácilmente comparar la eficiencia del resultado con algoritmos de “fuerza bruta” tales como asignar aleatoriamente un objetivo a cada robot.

5 Resultados actuales, Conclusiones y Líneas de Trabajo Futuro

Para lograr el objetivo planteado en el trabajo inicialmente se han estudiado los conceptos de Cloud Computing y la forma de incorporarlos gradualmente utilizando el entorno CMRE.

A partir de los conocimientos adquiridos, se ha adaptado el entorno CRME para que pueda ser utilizado desde un Cloud. Esta adaptación incluye el tener que comunicar el entorno (que corre en el Cloud) con los robots físicos. Para esto se ha estudiado el tema y se ha implementado un algoritmo de comunicación con sockets en lenguaje Python.

Asimismo se ha mantenido CMRE sobre una máquina local y se ha habilitado la comunicación virtual de cada robot con el cloud, pasando por la máquina local como un intermediario transparente al alumno.

Además en 2016 se avanzó en el trabajo presentado el año anterior [DEG15] para enseñar la importancia del concepto de consumo energético y exceso de temperatura en el área de concurrencia y paralelismo. En este sentido, actualmente se está desarrollando una tesina de grado y se ha modificado en forma experimental el entorno CMRE, incluyendo las instrucciones para medir y graficar el consumo energético y la temperatura de cada robot en una aplicación. A partir de los valores obtenidos se pueden tomar decisiones como reducir la velocidad de un robot, o hasta detener momentáneamente uno de ellos.

Asimismo el trabajo de los robots físicos contra el Cloud forman parte de un trabajo de Beca Doctoral y de 2 cursos de posgrado enfocados en Cloud Robotics.

Como Conclusiones podemos sintetizar:

- El entorno CMRE permite trabajar con algunos aspectos de Cloud Computing, en particular el enfoque de SAAS, sin modificaciones sustanciales.
- La potencialidad de vincular robots trabajando en tiempo real con la nube excede la simple enseñanza inicial de conceptos de programación paralela, pero es muy importante para los alumnos aprender algunos conceptos básicos desde un curso CS1.
- La programación de aplicaciones colaborativas sobre el entorno CMRE requiere mecanismos que permitan conocer el contexto global de la ciudad y características de cada robot. Este “conocimiento” y “volumen de datos” es natural que residan en la nube, lo cual se refuerza si pensamos en tratamiento de video o imágenes en tiempo real para la toma de decisiones por los robots.
- Es necesario diferenciar el alcance de la programación que puede realizar un alumno de un curso CS1 y aquel alumno de cursos avanzados.

Las principales líneas de trabajo futuro incluyen:

- Transmitir las nociones de IAAS permitiendo que el “super robot” pueda instanciar robots o datos en el entorno a demanda.
- Incorporar efectivamente al menos una cámara global para procesar el entorno para los robots físicos.
- Analizar casos con robots heterogéneos donde la función a optimizar sea el consumo de energía total.

6 Bibliografía

- [ACM04] ACM/IEEE-CS Joint Task Force on Computing Curricula. “Computer Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering”. Report in the Computing Curricula Series. 2004.
- [ACM08] ACM/IEEE-CS Joint Interim Review Task Force. “Computer Science Curriculum 2008: An Interim Revision of CS 2001”. Report from the Interim Review Task Force. 2008.
- [ACM13] ACM/IEEE-CS Joint Task Force on Computing Curricula. “Computer Science Curricula 2013”. Report from the Task Force. 2013.
- [ACM68] ACM Curriculum Committee on Computer Science. “Curriculum ‘68: Recommendations for the undergraduate program in computer science”. Communications of the ACM, 11(3):151-197. 1968.
- [ACM78] ACM Curriculum Committee on Computer Science. “Curriculum ‘78: Recommendations for the undergraduate program in computer science”. Communications of the ACM, 22(3):147-166. 1979.
- [ACM99] ACM Two-Year College Education Committee. “Guidelines for associate-degree and certificate programs to support computing in a networked environment”. New York: The Association for Computing Machinery. 1999.
- [AMD09] AMD. “Evolución de la tecnología de múltiple núcleo”. <http://multicore.amd.com/es-ES/AMD-Multi-Core/resources/Technology-Evolution>. 2009.
- [ARD09] Ardissono L., Goy A., Petrone G., Segnan M. "From Service Clouds to User-centric Personal Clouds". 2009 IEEE Second International Conference on Cloud Computing.

- [BER08] Bertogna, M., Grosclaude, E., Naiouf, M., De Giusti, A., Luque, E.: "Dynamic on Demand Virtual Clusters in Grids". In: 3rd Workshop on Virtualization in High-Performance Cluster and Grid Computing (VHPC 08). España. (2008).
- [DEG07] De Giusti, A., Naiouf, M., Pettoruti, J. E., Pousa, A., Rodríguez, I. P., De Giusti, L., Chichizola, F., Ardenghi, J., Bertogna, L., Printista, M.: Parallel Algorithms on Multi-Cluster Architectures using GRID Middleware. Experiences in Argentine Universities". In: Garcia Tobio, J., Doallo Biempica, R., López Cabido, I., Martín Santamaría, M. J., Gómez Tato, A., Touriño Dominguez, J. (eds.), 1st Iberian Grid Infrastructure Conference (IBERGRID) Proceedings, pp. 322—332. Spain. (2007)
- [DEG13] De Giusti, L., Leibovich, F., Sanchez, M., Chichizola, F., Naiouf, M., De Giusti, A. "Desafíos y herramientas para la enseñanza temprana de Concurrencia y Paralelismo". Congreso Argentino de Ciencias de la Computación (CACIC), 2013.
- [DEG14] De Giusti, A., De Giusti L., Leibovich, F., Sanchez, M., Rodriguez Eguren, S. "Entorno interactivo multirrobot para el aprendizaje de conceptos de Concurrencia y Paralelismo". Congreso Tecnología en Educación, Educación en Tecnología. 2014.
- [DEG15] Laura De Giusti, Fabiana Leibovich, Franco Chichizola, Marcelo Naiouf, Armando De Giusti. "Incorporando conceptos en la enseñanza de Concurrencia y Paralelismo utilizando el entorno CMRE" Congreso Argentino de Ciencias de la Computación 2015 (CACIC 2015).
- [FOS10] Foster I. "There's Grid in them thar Clouds". 2 de Enero, 2008.<http://ianfoster.typepad.com/blog/2008/01/theres-grid-in.html>. Noviembre, 2010.
- [GEP06] Gepner P., Kowalik M.F. "Multi-Core Processors: New Way to Achieve High System Performance". In: Proceeding of International Symposium on Parallel Computing in Electrical Engineering 2006 (PAR ELEC 2006). Pags. 9-13. 2006.
- [HOO13] Hoonlor A., Szymanski B. K., Zaki M. J., Thompson J. "An Evolution of Computer Science Research". Communications of the ACM. 2013.
- [KEH15] Kehoe B., Patil S., Abbeel P., Goldberg K. "A Survey of Research on Cloud Robotics and Automation". IEEE Transactions on Automation Science and Engineering (T-ASE): Special Issue on Cloud Robotics and Automation. Vol. 12, no. 2. Apr. 2015.
- [KON09] Kondo, D., Javadi, B., Malecot, P., Cappello, F., Anderson, D. P.: "Cost-benefit analysis of Cloud Computing versus desktop grids". In: IPDPS '09 Proceedings. IEEE International Symposium on Parallel and Distributed Processing. Washington, USA (2009).
- [MCC08] McCool M. "Scalable Programming Models for Massively Parallel Multicores". Proceedings of the IEEE, 96(5): 816–831, 2008.
- [MIL08] Miller M. "Cloud computing: web-based applications that change the way you work and collaborate online". Que Publishing. USA 2008.
- [MIN15] Proyecto PROGRAM.AR del Ministerio de Ciencia y Tecnología e Innovación Productiva. Web: <http://www.mincyt.gob.ar/divulgacion/programar-9920>.
- [NJS15] <https://nodejs.org/api/http.html>
- [ROD07] Rodríguez, I. P., Pousa, A., Pettoruti, J.E., Chichizola, F., Naiouf, M., De Giusti, L., De Giusti, A.: "Estudio del overhead en la migración de algoritmos paralelos de cluster y multicluster a GRID". In: Proceedings del XIII Congreso Argentino de Ciencias de la Computación. Argentina (2007).
- [VAQ09] Vaquero, L. M., Rodero-Merino, L., Caceres, J., Lindner, M.: "A Break in the Clouds: Towards a Cloud Definition". In: ACM SIGCOMM Computer Communication Review, Volume 39, Issue 1, pp. 50--55. USA. (2009).
- [VAZ09] Vázquez Blanco, B., Huedo, E., Montero, R. S., Llorente, I. M.: "Elastic Management of Cluster-based Services in the Cloud". In: Proceedings pp.19-24, ACM Digital Library. ISBN 978-1-60558-564-2. 2009.



V ETHICOMP

Latinoamérica

ID	Trabajo	Autores	
9220	Experiencia de estudio grupal. Primera introducción a las armas autónomas.	Lautaro De León (UNLP), Joaquín Bogado (UNLP), María Emilia Corrons (UNLP), María Beatriz García (UNLP), F. Javier Díaz (UNLP)	1368

Experiencia de estudio grupal. Primera introducción a las armas autónomas.

Federico Otarán¹, Lautaro De León¹, Joaquín Bogado¹, María Emilia Corrons¹, María Beatriz García², Francisco Javier Díaz¹

¹ Laboratorio de Investigación de Nuevas Tecnologías en Informática
LINTI - Facultad de Informática - UNLP

{ldeleon,ecorrns,federico.otaran}@cespi.unlp.edu.ar

{jbogado,jdiaz}@linti.unlp.edu.ar

<http://linti.unlp.edu.ar/>

² Sin afiliación

mariabeatriz.garcia@gmail.com

Resumen Este trabajo está dividido en dos partes, la primera donde presentamos una metodología que nos parece adecuada para encarar temas de ética profesional entre pares, tomada de otros grupos de investigación. Esta técnica, puesta en práctica, nos permitió elegir y estudiar un tema con cierta profundidad. La segunda parte de los objetivos es mostrar los resultados de dicha metodología al estudio de las cuestiones éticas relacionadas con las armas autónomas. Estos resultados son parciales y parte de un trabajo en progreso.

Keywords: ética informática, armas autónomas, máquinas morales

1. Introducción

“Definitivamente desencadenado, Prometeo, al que la ciencia proporciona fuerzas nunca antes conocidas y la economía un infatigable impulso, está pidiendo una ética que evite mediante frenos voluntarios que su poder lleve a los hombres al desastre” Así comienza el prólogo, fechado en New York en julio de 1979 del libro de Hans Jonas titulado “El principio de Responsabilidad: ensayo de una ética para la civilización tecnológica”. [11] Para Jonas es el encuentro entre la ciencia y la tecnología lo que caracteriza a la modernidad. El “progreso” tiende, en efecto, a cambiar la vida humana a una velocidad cada vez más intensa, esto hace necesario un replanteamiento de la ética. “¿Qué podrá servirnos de guía?...;El propio peligro que prevemos!” [11].

Con la idea de comenzar a estudiar temas de ética profesional, en mayo de 2016 conformamos un grupo de estudio, el cual a través de reuniones regulares, nos iba a permitir investigar la forma en la que la aparición de nuevas tecnologías influye en la sociedad y viceversa, cómo la sociedad influye en la aparición de nuevas tecnologías y en los nuevos usos que se hace de la ya existente.

La mecánica propuesta para el desarrollo del proceso de estudio fue basada en el uso de técnicas asociadas a las metodologías ágiles de desarrollo [20], aunque

también tomamos ideas de grupos de estudios de otra áreas como el psicoanálisis. Dicho proceso, sin embargo, fue adaptado a nuestras necesidades particulares sobre la marcha. Decidimos reunirnos una vez por semana durante unas 2 horas, en las cuales cada uno pondría en común sus ideas respecto a lo investigado durante la semana y si bien mantuvimos cierta flexibilidad en cuanto a horarios, que los temas fueran de interés para el grupo ayudó a no perder la constancia.

Como primer paso y durante los primeros encuentros, además de proponer y dar forma a la metodología, hicimos una primera puesta en común al estilo *brainstorm*[21] para seleccionar un tema de estudio. La decisión de encarar uno solo de manera grupal y no uno por cada integrante se basó, en principio, en que el tiempo que podíamos dedicar al proyecto era bastante reducido y era preferible enfocar los esfuerzos lo más posible. De entre los temas propuestos elegimos las Armas Autónomas por diversos motivos, entre los cuales podemos incluir que, sin haber ahondado en el tema, parecía más sencillo, principalmente debido a que la comunidad científica en su mayoría tiene una posición tomada al respecto[1][2][3]. Incluso Argentina se ha manifestado frente a organismos internacionales respecto a la necesidad de control de este tipo de armas[4][5].

A partir de esto, seguimos una serie de pasos que constituyen la columna vertebral de esta metodología. El primero se basó en obtener de las distintas fuentes bibliográficas disponibles, la información necesaria acerca de esta temática que nos permitiera tener una idea clara del estado del arte de las armas autónomas en la actualidad. Posteriormente, en base a la información acopiada e investigada, tratamos de comprender los argumentos a favor y en contra del desarrollo de las armas autónomas. A su vez, durante las reuniones presenciales, debatíamos entre nosotros sobre la contundencia o la debilidad de éstos y registrábamos las dudas que fueron surgiendo al contrastarlos y discutirlos por sobre nuestras ideas y subjetividades para con los sistemas de armas autónomas. Cabe destacar que muchas de ellas quedaron sin contestar.

Sin desalentarnos ni mucho menos ante la falta de respuestas, seguimos adelante convencidos que mostrando que podemos trabajar en responderlas, crearemos conciencia acerca de la necesidad de discutir estos temas en la comunidad.

2. De nuestros antecedentes

El tema de la Ética Profesional fue nutriendo la curiosidad de uno de los jóvenes que conforma este grupo y fue adquiriendo mayor importancia como eje de investigación con el paso del tiempo y las conversaciones informales con personas interesadas en esta problemática. En el contexto de CACIC 2011 con sede en la Facultad de Informática de la UNLP, se realizaba el primer ETHICOMP Latinoamérica, lo que mostraba el auge y la importancia del tema a nivel nacional.

En 2012, también en el ámbito del II ETHICOMP, se presenta un primer trabajo[22] y en 2013 en III ETHICOMP se presenta un segundo trabajo[23], que buscaba de alguna manera continuar algunas líneas de trabajo surgidas a raíz del trabajo presentados el año anterior, tomando como referencia el trabajo de

Matthew J. Sher. La participación y el avance en la investigación en estos temas siempre fue posible gracias al apoyo incondicional de la dirección del Laboratorio de Investigación de Nuevas Tecnologías en Informática (LINTI).

Fue necesario que transcurriese un tiempo para que alguien tomara la iniciativa de formar el grupo de investigación, que hoy estamos presentando en la comunidad científica, conformado por cuatro investigadores del LINTI, estudiantes de la carrera de Licenciatura en Informática y Licenciatura en Sistemas de la UNLP, dos de ellos egresados y dos en la etapa final de la misma.

Como bien señala Adela Cortina en su libro[10] “¿Para qué sirve realmente la ética?”: “El mundo de las profesiones tiene una larga historia, que se suele contar desde la tradición occidental, de la que se dice que nace con el célebre Juramento de Hipócrates, ligado a una profesión tan valorada como la médica. Otras dos profesiones le acompañarían en los orígenes, la de los sacerdotes y los juristas, de modo que entre las tres se ocuparían de cosas tan importantes para la vida de una sociedad como el bien del cuerpo, el bien del alma y el de la comunidad política. Quien ingresa en una profesión se compromete a proporcionar ese bien a su sociedad, tiene que prepararse para ello adquiriendo competencias adecuadas, y a la vez ingresa en una comunidad de profesionales que comparten la misma meta”.

Y un poco más adelante en este mismo libro podemos leer algo del espíritu que anida en este grupo de estudio e investigación y marca su norte: “Frente al *êthos* burocrático de quién se atiene al mínimo legal pide el *êthos* profesional, la excelencia, porque su compromiso fundamental no es el que le liga a la burocracia, sino a las personas concretas, a las personas de carne y hueso, cuyo beneficio da sentido a cualquier actividad e institución social. Es tiempo, pues, no de despreciar la vida corriente, sino de introducir en ella la aspiración a la excelencia.”

3. Las armas autónomas

Entendemos por Arma Autónoma (Autonomous Weapon), Sistema Letal Autónomo (Lethal Autonomous Weapon System, LAWS) o Arma Robótica Autónoma (Autonomous Robotic Weapons, ARW) a cualquier sistema de armamento capaz de tomar la decisión de disparar contra un blanco sin la intervención de un ser humano, de acuerdo con la definición de “Human-out-of-the-loop Weapons”[15]. No se conocen hasta el momento sistemas de este tipo que estén siendo usados en servicio, pero sí existen sistemas de defensa altamente automatizados como el sistema comercializado por Samsung, SGR-A1 utilizado en la zona desmilitarizada que separa Corea del Norte de Corea del Sur. Estos sistemas requieren pocos pasos para volverse completamente autónomos y países como China, Israel, Rusia y el Reino Unido ya han manifestado su interés en el desarrollo de esta tecnología[16]. Que un algoritmo, es decir, un programa de computadora tome la decisión de disparar terminando así con la vida de alguien en el otro extremo del arma, sin que sea otra persona la que aprieta el gatillo,

plantea problemas filosóficos incipientes acerca de diversos factores implicados en dicha decisión.

Chamayou en su libro “A Drone Theory” [6], se pregunta quién sería el responsable en el caso hipotético de que un robot cometiera un crimen de guerra. ¿El general que lo empleó? ¿El Estado, que es su propietario? ¿Los informáticos que lo programaron?

Müller[17] por su parte, sostiene que los LAWS no plantean desafíos sustanciales a las leyes humanitarias ni plantean problemas a la hora de determinar responsabilidades, que las consecuencias de usar este tipo de armas serían en general positivas y que por lo tanto no deberían prohibirse sino regularse. Sin embargo concede que algunos aspectos de las guerras podrían empeorar¹ y que los principios de Guerra Justa (*Jus in Bello*) de distinción y proporcionalidad² podrían verse comprometidos.

Nosotros a nuestra vez nos preguntamos, como desarrolladores de software, si será posible diseñar una máquina incapaz de cometer un crimen de guerra³ pero que de todas formas continúe siendo útil a sus fines prácticos. Las ventajas que constantemente se remarcan en favor de las armas autónomas [7][8][9] incluyen entre otras:

- Reemplazan a los humanos en el campo de batalla, reduciendo así el posible número de víctimas.
- Una máquina es superior a un humano en varios aspectos. La máquina cuenta con un menor tiempo de reacción y con mayor precisión, no necesita descansar y puede funcionar con igual efectividad en ambientes hostiles y situaciones adversas.
- Una máquina no experimenta sensaciones de ira, enojo, miedo o ansias de venganza. Esto evitaría posibles violaciones a las reglas de enfrentamiento y leyes de guerra.

Respecto al tercer punto, para que un robot sea capaz de seguir las leyes de guerra, debería ser posible codificarlas de alguna manera, es decir, plasmarlas en un algoritmo. Dicho algoritmo diferenciaría para la máquina aquello que es moralmente correcto de lo que no lo es. Sin embargo, el trabajo de M. Englert *et al.*[12] propone límites al accionar moral de una máquina. En dicho trabajo, los autores proponen un Gedankenexperiment, una variante del problema del tranvía descrito por Thomson[13], en la cual:

- Solo hay una única acción correcta⁴ entre dos opciones.

¹En particular, Müller habla de que la reducción del costo de la guerra haría la guerra más probable.

²El principio de Distinción refiere a la distinción de Civiles de Militares y es crucial a la hora de seleccionar un blanco, mientras que el principio de Proporcionalidad refiere a que el uso de la fuerza debe estar en proporción al objetivo militar propuesto.

³Un crimen de guerra es una violación a las leyes de guerra y a las reglas de enfrentamiento descritas en diversos tratados internacionales.

⁴Los autores entienden como “correcta” a aquella opción moralmente aceptable y la diferencian de otra que claramente no lo es.

- Toda la información para resolver el problema está disponible.
- Todas las acciones ocurren de manera determinística.
- Y sin embargo, es fundamentalmente imposible reconocer la opción correcta de manera algorítmica.

El escenario propuesto se resume en que el programa debe elegir la alternativa moralmente más aceptable basándose en determinar si el código fuente de un programa contiene instrucciones maliciosas que efectivamente se ejecutarán al correr dicho programa. Usando métodos formales de teoría de la computación y verificación de programas, los autores demuestran que este problema no puede resolverse para todas las entradas. Dicho de otra manera, habrá casos, situaciones particulares para los cuales un algoritmo no podrá decidir cuál es la opción moralmente correcta a pesar de que se cumplan los puntos enumerados anteriormente.

Aquí nos parece razonable distinguir claramente dos conceptos que algunos autores como W. Wallach y C. Allen en su libro *Moral Machines*[14] utilizan de manera intercambiable, los conceptos de la Moral y la Ética. Parece posible, para dichos autores, diseñar una máquina que distinga el bien del mal siguiendo un algoritmo, es decir una máquina moral. Este comportamiento debería ser especificado de antemano por los programadores de dicho algoritmo. Esta tarea que al parecer es en extremo difícil, además estaría limitada, como queda demostrado por los métodos formales descritos anteriormente. No obstante, creemos que el actuar éticamente y aquí la diferencia con la Moral, es una actividad estrictamente humana e individual, que involucra otras actividades intrínsecamente humanas las cuales no son programables, como por ejemplo la búsqueda del buen vivir, el entendimiento y la deliberación[24].

4. Los drones y las armas autónomas

Notamos también un paralelismo entre algunos argumentos a favor del uso de Drones de Combate con fines militares con los argumentos a favor de las armas autónomas. Los Drones de combate o UAV (Unmanned Aerial Vehicles) se diferencian de las Armas Autónomas en que los primeros cuentan con un operador humano (dos en el caso del MQ-9 Reaper, un piloto y un operador de sensores). Por un lado, el dron permite proyectar fuerza sin proyectar vulnerabilidad[6], esto es, el dron es capaz de infringir fuerza letal a la vez que resguarda la vida de los operadores humanos que generalmente se encuentran en la sala de comando. De esta manera se entiende que el uso de drones permite “salvar vidas”. Sin embargo, las vidas salvadas son las de los combatientes del propio bando que las utiliza, mientras que las vidas que peligran son las de los militares y civiles del otro bando. De la misma manera, las Armas Autónomas permiten salvar vidas, a la vez que, como los drones, eliminan el combate y la posibilidad de defensa del enemigo. Así la guerra deja de ser tal para convertirse en una mera matanza, una aniquilación del enemigo.

La guerra, un estado de conflicto armado entre dos Estados o grupos dentro de un Estado[19], es el único en el cual se le está permitido a una persona tomar

la vida de otra sin que esto sea considerado un crimen. Mejorar las armas con el fin de hacer la guerra más barata bajaría el umbral para empezar un conflicto bélico en lugar de resolver las diferencias por otros medios.

El uso de los drones como dispositivos bélicos hecho por la administración de Obama ha sido duramente cuestionado[18] y sus diversos matices se plasman en la película “Eye in the Sky” dirigida por Gavin Hood y protagonizada por Helen Mirren, Aaron Paul y Alan Rickman. En el film puede verse cómo las reglas de enfrentamiento se ven modificadas en pos de acabar con la vida de cinco conocidos terroristas a pesar de los posibles daños colaterales a la población civil de un país aliado. A la vez que muchas de las situaciones del film plantean los mismos interrogantes si se reemplazara el dron por un arma autónoma, encontramos que en el caso del arma autónoma aparecen posibilidades todavía más siniestras. Está claro que, si bien estas armas podrán tomar decisiones de vida o muerte, actuarán siguiendo órdenes que podrían estar en conflicto con las leyes de guerra, pero si el arma no tiene mecanismos para rehusar una orden, podría estar siendo usada para cometer un crimen.

Supongamos, por otra parte, que se cometiera dicho crimen, adrede o de manera accidental, y que el arma tomara la decisión de disparar contra una persona terminando con su vida. Chamayou[6] argumenta que por un lado existe la posibilidad que los responsables del despliegue, la programación o los dueños de dichos sistemas (el Estado o una empresa privada, por ejemplo) puedan “pasarse la pelota”, y por otro lado, que el único responsable claramente identificado es la propia víctima, la cual por sus actos voluntarios o involuntarios, accionó los mecanismos para que el arma dispare.

5. La posición Argentina respecto a las Armas Autónomas

La posición oficial de Argentina respecto a las armas autónomas puede resumirse en el siguiente extracto tomado de un documento presentado en un debate ante el “Human Rights Council” en Ginebra el 30 de Mayo de 2013[4]: “Deseamos destacar nuestra preocupación por las referencias a que estos sistemas podrían llevar a una ‘normalización del conflicto’, y a una posible carrera de armamentos que crearía divisiones entre los Estados y debilitaría el derecho internacional; el posible fomento de las represalias, la retaliación y el terrorismo; y su impacto en materia de derechos humanos y del derecho internacional humanitario. Como forma de evitar estas consecuencias negativas, el informe concluye en que debe establecerse un órgano internacional encargado de examinar la situación y articular las opciones a más largo plazo.”

Entendemos que el desarrollo de armamento es un negocio global y que una censura total a la investigación y desarrollo de este tipo de dispositivos podría incomodar a muchos Estados. Pero los interrogantes al respecto de las armas autónomas siguen siendo demasiados como para alentar su desarrollo sin que haya garantías, entre las cuales, podemos destacar el acceso a la justicia para

aquellas víctimas del abuso de este tipo de armamento o la falta de certezas en cuanto a las ventajas que éstas brindan.

6. Conclusiones y trabajo a futuro

La metodología descrita nos permitió encarar un tema de estudio como grupo. En futuras iteraciones debemos mejorar aspectos como sistematizar la toma de notas, referencias bibliográficas y extractos. También encontramos dificultades a la hora de fundamentar algunos argumentos, probablemente debido al desconocimiento de las principales corrientes filosóficas. Creemos que sería beneficioso un estudio de dichas corrientes para luego poder aplicarlas a temas específicos concernientes a la ética informática.

Además, consideramos que la postura que toma Argentina sobre la temática de armas autónomas es débil; no es suficiente contar con un organismo que regule su uso. Nos parece que Argentina debería desalentar el desarrollo de las mismas y no invertir fondos públicos en la investigación de estas tecnologías.

Como trabajo a futuro, creemos también necesario compartir nuestras inquietudes y reflexiones con nuestros colegas y con la comunidad en general, como parte integral de las actividades del grupo y a intervalos regulares. De esta manera, el grupo de investigación estaría contribuyendo a uno de los principales objetivos de la Universidad, que es concientizar a las futuras generaciones de profesionales. Estas presentaciones previenen trabajos como el presentado, pero también a través de charlas, paneles y foros de discusión.

Referencias

1. Stop Killer robots Campaigning - <http://www.stopkillerrobots.org>
2. International Committee for Robot Arms Control - <http://icrac.net/>
3. Autonomous Weapons: an Open Letter from AI & Robotics Researchers - <http://futureoflife.org/open-letter-autonomous-weapons>
4. Document delivered by Mr. Mariano Alvares Wagner on behalf of GRULAC. 2013 - http://www.stopkillerrobots.org/wp-content/uploads/2013/05/HRC\Argentina_09_30May2013.pdf
5. Country Statements about Killer Robots. 2014 - http://www.stopkillerrobots.org/wp-content/uploads/2013/03/KRC_CountryStatus_14Mar2014.pdf
6. *Grégoire Chamayou*. A drone theory. Kindle Edition, Ed Penguin 2015.
7. *William M. Fleischman*. Why We Should Not Build Autonomous Robotic Weapons. ETHICOMP 2013.
8. *Ronald Arkin*. Lethal Autonomous Systems and the Plight of the Non-combatant. 2013
9. *Ronald Arkin*. Ethical Robots in Warfare. www.cc.gatech.edu/ai/robot-lab/online-publications/arkin-rev.pdf
10. *Adela Cortina*. ¿Para qué sirve realmente la ética?. Ed.Paidós 2013.
11. *Jonas Hans*. Principio de Responsabilidad - Ensayo de una ética para la civilización tecnológica. Ed. Herder 1995.
12. *M. Englert, S. Siebert, M. Ziegler*. Logical Limitations to Machine Ethics with Consequences to Lethal Autonomous Weapons. CoRR 2014.

13. *Judith J Thomson*. The Trolley Problem. Yale Law Journal 1985.
14. *Wendell Wallach, Colin Allen*. Moral Machines: Teaching Robots Right from Wrong. Kindle Edition, Oxford University Press 2008.
15. Losing Humanity, a case against killer robots. Humans Right Watch 2012.
16. Shaking The Foundations: The Human Rights Implications of Killer Robots. Humans Right Watch 2014.
17. *V. Müller, T. Simpson*. Autonomous Killer Robots Are Probably Good News. PhilPapers 2015.
18. The Forgotten Victims of Obama's Drone War. The New York Times 2013
<http://www.nytimes.com/2013/05/23/opinion/the-forgotten-victims-of-obamas-drone-war.html>
19. Guerra. Diccionario de la real academia española, edición digital. - <http://dle.rae.es/?id=JoNx0nS>
20. *Martin Fowler*. The New Methodology. Blog post, 2005. - <http://martinfowler.com/articles/newMethodology.html>
21. *Alex Faickney Osborne*. Applied Imagination: Principles and Procedures of Creative Problem Solving. Ed. Scribner 1953.
22. *J. Bogado, B. García*. Reflexiones iniciales acerca de la validez ética de la utilización de técnicas de minería de datos sobre datos personales en la búsqueda de terroristas. ETHICOMP 2012.
23. *M. B. García, W. Fleischman, J. Bogado*. Una conversación con Matthew Sher sobre privacidad y la amistad. ETHICOMP 2013.
24. *Aristóteles*. Ética a Nicómaco. Ed Gredos 2010.