

TOMA DE DECISIONES MULTICRITERIO EN PROBLEMAS DE LA INGENIERÍA DE SOFTWARE UTILIZANDO COMPUTACIÓN BLANDA

Carlos Casanova, Manuel Chichi, María Luján Gabioud, Fernando Pereyra Rausch, Lucas Prado, Giovanni Daián Rottoli, Esteban Schab, Anabella De Battista

Grupo de Investigación sobre Inteligencia Computacional e Ingeniería de Software, Departamento Ingeniería en Sistemas de Información, Fac. Reg. Concepción del Uruguay, Universidad Tecnológica Nacional
Entre Ríos, Argentina

{casanovac, chichim, gabioudl, pereyraf, pradol, rottolig, schabe, debattistaa}@frcu.utn.edu.ar

RESUMEN

La Ingeniería de Software (IS) como la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software. Sin embargo, la adopción de herramientas formales que complementen la experiencia y el buen juicio en las distintas actividades de un proceso de desarrollo de software todavía es un pendiente dentro la industria del software, particularmente en la Argentina. Dos carencias que pueden explicar esto son, por un lado, la falta de conocimientos respecto de enfoques realistas para resolver problemas relativos a la IS, y por otro, la falta de herramientas software que auxiliien a los tomadores de decisiones utilizando tales enfoques. Las líneas de investigación aquí propuestas tienden a suplir ambas carencias. Para esta tarea se propone la utilización de técnicas comprendidas en lo que se conoce como computación blanda (*soft computing*), dentro de las cuales se encuentran la Teoría de Conjuntos Difusos, las Redes Neuronales y los métodos de Búsqueda Heurística y Metaheurística. La *soft computing* es capaz de brindar la flexibilidad necesaria para crear métodos y modelos que sean tolerantes a la imprecisión, la falta de información y la aproximación, características que le son propias a los contextos de decisión en la IS.

Palabras clave: Ingeniería de Software, Optimización, Soft Computing, Teoría de Conjuntos Difusos, Metaheurísticas, Preferencias.

CONTEXTO

El presente trabajo se desarrolla en el ámbito del Grupo de Investigación sobre Inteligencia Computacional e Ingeniería de Software (GIICIS), perteneciente al Departamento Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional, Facultad Regional Concepción del Uruguay.

1. INTRODUCCIÓN

El Vocabulario de la Ingeniería de Sistemas y Software de ISO/IEC/IEEE (SEVOCAB) define a la ingeniería de software como la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software; es decir, la aplicación de la ingeniería al software [1]. El Software es un bien intangible que cumple un doble rol. Es un producto y al mismo tiempo es el vehículo para entregar el producto más importante de nuestro tiempo: la información.

El software se desarrolla o modifica con intelecto: no se manufactura en el sentido clásico. Los costos del software se concentran en la ingeniería. Esto significa que los proyectos de software no pueden administrarse como si fueran proyectos de manufactura.

En este contexto, un proceso de software es un conjunto de actividades relacionadas que conducen a la producción de un producto de software [2]. Sin embargo, no es una prescripción rígida de cómo elaborarlo. Por el contrario, debe ser ágil y adaptable (al problema, al proyecto, al equipo y a la cultura organizacional). Por tanto, un proceso adoptado para un proyecto puede ser significativamente distinto de otro adoptado

para otro proyecto [2]. El proceso de software es complejo y, como todos los procesos intelectuales y creativos, confía en que las personas tomen decisiones con criterio.

2. ESTADO DE LA INDUSTRIA EN ARGENTINA

Aún así, en la República Argentina la industria del software, de un tiempo a esta parte, ha ido ganando progresivamente visibilidad e importancia para la economía del país. Un estímulo importante que ha incentivado el crecimiento de esta industria es el Régimen de Promoción de Software, instituido por la Ley 25.922 y con vigencia hasta el 31 de diciembre de 2019, el cual se reconoce como un instrumento que resultó eficiente para la promoción de dicha industria. Las actividades de la Economía del Conocimiento son las más dinámicas de la economía argentina: entre 2007 y 2017 generaron 65% más empleo que en el resto de las actividades productivas y sus operaciones crecieron en el mismo periodo un 70%, frente a un crecimiento general del 12%. Además, los salarios son un 35% más elevados que en el resto de los sectores. Actualmente, emplean a 215 mil personas y según el informe [3] del año 2017 de Argenconomics (entidad conformada por empresas prestadoras de servicios basados en el conocimiento, relacionadas con los mercados externos) las exportaciones de sectores vinculados a la Economía del Conocimiento ascendieron a USD 6.168 millones de dólares en ese año, convirtiéndose en el tercer rubro exportador del país, sólo detrás de la soja y sus derivados, y los cereales, y en el segundo exportador de Servicios Basados en Conocimiento (SBC) a nivel regional. En el primer semestre de 2019, según datos de la Secretaría de Transformación Productiva sobre la base del INDEC [4], los tres principales complejos exportadores fueron el cerealero/oleaginoso (US\$ 13.596 millones), el automotriz (US\$ 3802 millones) y el de los SBC (US\$ 3385 millones).

3. PROBLEMAS QUE SURGEN EN EL CONTEXTO DE LA IS

La comunidad científica viene realizando esfuerzos en la definición realista de

problemas propios de la ingeniería de software. El primer paso en cualquier metodología científica es la definición del problema. Sin una definición apropiada del problema, se corre el riesgo de encontrar una solución elegante para un problema que no existe. Características comunes que pueden identificarse son las siguientes:

- Hay una necesidad de **balancear objetivos que compiten** entre sí, es decir, se clasifican como problemas de optimización multi-objetivo.
- Existe la necesidad de tratar con **información incompleta, imprecisa o inconsistente** desde la formulación de los problemas. Este escenario es propio de entornos donde se desenvuelven los seres humanos. Por caso, puede darse que no se cuente con suficiente información para estimar con un grado de certeza aceptable ciertos parámetros. También puede darse la falta de acuerdo entre varios expertos al realizar una estimación, o que la misma sea establecida de forma imprecisa, por ejemplo, puede requerirse que el acoplamiento entre dos módulos sea “bajo” o que ciertos *stakeholders* resulten “muy favorecidos” en la próxima versión.
- Existen muchas soluciones potenciales, usualmente caracterizadas por una **explosión combinatoria** de las variables de decisión.

4. INVESTIGACIÓN OPERATIVA

La Investigación de Operaciones (IO) es un enfoque científico en la toma de decisiones que busca el mejor diseño y operación de un sistema, por lo regular en condiciones que requieren la asignación de recursos escasos [5]. Como su nombre lo indica, el objetivo de esta disciplina implica “investigar sobre las operaciones”. En consecuencia, esta disciplina se aplica a la problemática relacionada con la conducción y la coordinación de actividades en una organización.

La IO incluye el término investigación porque utiliza un enfoque similar al que se aplica en las áreas científicas establecidas. El proceso comienza por la observación cuidadosa y la formulación del problema, lo cual incluye la recolección de los datos pertinentes. El

siguiente paso es la construcción de un modelo científico —generalmente matemático— con el cual se intenta abstraer la esencia del problema real. En esta etapa se propone la hipótesis de que el modelo será una representación tan precisa de las características esenciales de la situación, que permitirá que las conclusiones —soluciones— que se obtengan sean válidas también para el problema real. Después se llevan a cabo los experimentos adecuados para probar esta hipótesis, para modificarla si es necesario y para verificarla en determinado momento. En cierto sentido, la IO involucra la investigación científica creativa de las propiedades fundamentales de las operaciones. Sin embargo, es más que esto. La IO se ocupa también de la administración práctica de la organización. Por lo tanto, para tener éxito, también debe proporcionar conclusiones claras que el tomador de decisiones pueda usar cuando sea necesario [5].

5. SOFT COMPUTING

La *soft computing* difiere de la “*hard computing*” o computación convencional en que, a diferencia de esta última, es tolerante a la imprecisión, la incertidumbre, la verdad parcial y la aproximación. En efecto, el modelo a seguir para la *soft computing* es la mente humana. El principio guía de la *soft computing* es: aprovechar la tolerancia a la imprecisión, la incertidumbre, la verdad parcial y la aproximación para conseguir tratabilidad, robustez y soluciones de bajo costo. Las técnicas de la *soft computing* incluyen la lógica difusa, algoritmos genéticos, redes neuronales artificiales, aprendizaje de máquinas y sistemas expertos [6]. Estos múltiples métodos no son competitivos entre sí, sino que son complementarios y pueden ser utilizados juntos para resolver un problema dado.

La precisión, el rigor y la certeza son atributos que implícitamente establecen un compromiso: se está en posesión de toda la información necesaria para poder encontrar la mejor solución posible para el problema dado. En efecto, la complejidad de ciertos problemas produce que no sea posible utilizar técnicas

tradicionales para abordarlos si lo que se pretende son buenas soluciones en lapsos razonables. Más aún, para problemáticas cercanas a la realidad, las hipótesis y supuestos de las mismas pueden hacer imposible la aplicación de técnicas de computación convencional, generalmente debido a la presencia de formas específicas de incertidumbre en los parámetros. La *soft computing*, en tal caso, apunta a resolver los problemas aprovechando la imprecisión y la incertidumbre presente en el proceso de toma de decisiones, brindando las “soluciones más satisfactorias” en tal contexto de toma de decisiones.

6. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

6.1. Formulación realista de problemas de la IS como problemas de búsqueda/optimización

A nivel de industria de software mundial, ninguna técnica formal ha tenido un impacto significativo [7]. Una de las razones que puede explicar este fenómeno es que los modelos y algoritmos que se utilizan no tienen en cuenta desde su formulación, en su gran mayoría, la incertidumbre inherente al proceso de desarrollo de software (y por lo tanto, las soluciones obtenidas resultan de una vida útil casi efímera), o bien, si la contemplan, los modelos y resultados son difíciles de entender para una persona no experta en optimización o inteligencia artificial [8]. En este sentido, se busca formular modelos matemáticos utilizando mecanismos formales de captación de incertidumbre (como la teoría de conjuntos difusos, o la de posibilidades) que resulten adecuados para la realidad de un proceso de desarrollo de software. El primer paso, por tanto, consiste en formular el problema como uno de búsqueda/optimización. En este sentido, se trabaja junto a expertos de la industria del software en la especificación de problemas de la IS, con el objetivo de darle utilidad a las soluciones que se pueden obtener con los distintos algoritmos disponibles.

6.2. Mecanismos de captación de preferencias

Existen múltiples enfoques de optimización multiobjetivo, entre ellos los basados en preferencias. Estos suponen que puede incluirse información del tomador de decisiones para guiar la búsqueda hacia regiones prometedoras del espacio de soluciones. Idear mecanismos adecuados para captar esta información no es una tarea trivial, y se trabaja en el estudio de los mecanismos disponibles, y el diseño e implementación de nuevos mecanismos para mejorar principalmente la usabilidad de los métodos. Los componentes analizables de estos mecanismos son principalmente tres: la cantidad de interacciones con el usuario, qué información se le solicita, y en qué momento del proceso de optimización [9].

6.3. Integración con herramientas existentes de gestión de proyecto / proceso / producto

Como ya se ha dicho, los modelos y resultados de esta línea de investigación pueden resultar difíciles de entender para una persona no experta en optimización o inteligencia artificial. Se pretende, por lo tanto, construir una herramienta software que soporte la toma de decisiones que sea usable por administradores no expertos en *soft computing*. Es razonable suponer que si esta herramienta se integra a herramientas que la industria ya utiliza tendrá mayores posibilidades de ser incorporada.

7. RESULTADOS OBTENIDOS/ESPERADOS

Hasta el momento se han resuelto satisfactoriamente varios problemas, aunque ninguno de ellos está agotado. Entre ellos se encuentran el problema del próximo lanzamiento (*Next Release Problem*, NRP), resuelto mediante múltiples algoritmos evolutivos como NSGA-II, IBEA y una versión novedosa de PSO denominada FMOPSO [10], desarrollada por los autores, escrita en C++. Otro problema es el de la planificación de lanzamientos (*Release*

Planning Problem, RPP), una extensión del NRP, en este caso utilizando programación matemática difusa, incorporando restricciones blandas [11]. También se ha resuelto el problema de priorización de requerimientos de software utilizando relaciones de preferencia difusa [12]. Para esto se desarrolló un prototipo en Octave que todavía debe ser mejorado.

Lograr un primer prototipo de interfaz usable para usuarios no expertos en este tipo de técnicas es un objetivo esperado. Este prototipo debe incorporar la posibilidad de realizar la carga de los datos necesarios de los problemas y la inducción de las preferencias de los usuarios. Además se proyecta poder construir *plugins* o mecanismos de interoperabilidad con *suites* existentes usadas en la industria.

Finalmente, se espera contar al final del proyecto con una colección de modelos altamente cohesivos que brinden información para facilitar la toma de decisiones relativas a distintos problemas de la Ingeniería de Software.

8. FORMACIÓN DE RECURSOS HUMANOS

La investigación presentada constituye las líneas fundacionales de un nuevo grupo de investigación dentro de la UTN-FRCU, el GIICIS. Un investigador ha sido invitado a realizar una investigación posdoctoral en el Grupo de Investigación en Modelos de Decisión y Optimización, Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada, España. Dos investigadores se encuentran realizando su tesis de doctorado. Además participan en el proyecto dos becarios alumnos de la carrera Ingeniería en Sistemas de Información que inician su formación en la investigación, dos alumnos realizan su práctica supervisada y uno su trabajo de fin de carrera en el contexto de esta investigación. En el marco de este proyecto ya se ha defendido exitosamente una práctica profesional supervisada y hay otra en inminente presentación.

9. BIBLIOGRAFÍA

- [1] Sweb. E. Board, *Guide to the software engineering body of knowledge (SWEBOK (R)): Version 3.0*. IEEE Computer Society, 2014.
- [2] I. Sommerville, *Software Engineering*, 9.^a ed. Pearson Education, Inc., 2011.
- [3] Argenconomics, «Informe Economía del Conocimiento 2017», 2018. [En línea]. Disponible en: <http://www.argencon.org/nota628-Argencon-realizo-su-Asamblea-Anual-y-presento-su-nuevo-informe-Argenconomics>. [Accedido: 17-feb-2020].
- [4] S. de la T. P. Dirección Nacional de Análisis y Estadísticas Productivas, Subsecretaría de Desarrollo y Planeamiento Productivo, «Argentina productiva: economía del conocimiento», 2019.
- [5] F. S. Hillier y G. J. Lieberman, *Introducción a la Investigación de operaciones*. McGRAW-HILL/INTERAMERICANA EDITORES, S.A. DE C.V., 2010.
- [6] L. A. Zadeh, «Fuzzy Logic, Neural Networks, and Soft Computing», *Fuzzy Systems*, vol. 37, n.º 3, pp. 77-84, 1994.
- [7] D. C. C. Peixoto, G. R. Mateus, y R. F. Resende, «The issues of solving staffing and scheduling problems in software development projects», en *Proceedings - International Computer Software and Applications Conference*, 2014.
- [8] J. M. Rojas y G. Fraser, «Is search-based unit test generation research stuck in a local optimum?», en *Proceedings - 2017 IEEE/ACM 10th International Workshop on Search-Based Software Testing, SBST 2017*, 2017, n.º c, pp. 51-52.
- [9] T. N. Ferreira, S. R. Vergilio, y J. T. de Souza, «Incorporating user preferences in search-based software engineering: A systematic mapping study», *Inf. Softw. Technol.*, vol. 90, pp. 55-69, 2017.
- [10] C. Casanova *et al.*, «Aproximación del Frente Pareto-Óptimo de un Problema NRP Bi-Objetivo mediante un Algoritmo basado en Enjambres de Partículas», en *Anales VI Seminario Argentina-Brasil de Tecnologías de la Información y la Comunicación*, 2018.
- [11] C. Casanova, F. Pereyra Rausch, y L. Prado, «Modelo de Planificación de Lanzamientos de Software utilizando Restricciones Blandas», en *VII Congreso Nacional de Ingeniería Informática / Sistemas de Información*, 2019.
- [12] M. L. Gabioud y C. Casanova, «Priorización en Ingeniería de Requerimientos con Preferencias Difusas», en *VII Congreso Nacional de Ingeniería Informática / Sistemas de Información*, 2019.