

TOMA DE DECISIONES CIENTÍFICA EN LA INGENIERÍA DE SOFTWARE MEDIANTE INTELIGENCIA COMPUTACIONAL Y ANÁLISIS DE DATOS

Carlos Casanova, Manuel Chichi, Leonardo Hoet, 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, hoetl@frcu.utn.edu.ar, pereyraf, pradol, rottolig, schabe, debattistaa}@frcu.utn.edu.ar

RESUMEN

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. La falta de conocimientos respecto de enfoques realistas para resolver problemas de la IS y la falta de herramientas software que auxilien a los tomadores de decisiones utilizando tales enfoques son dos carencias que pueden explicar las dificultades en esta adopción. Las líneas de investigación aquí propuestas tienden a suplir ambas. Para esta tarea se propone la utilización de tanto técnicas comprendidas en lo que se conoce como Inteligencia Computacional (IC), dentro de las cuales se encuentran la teoría de conjuntos difusos, las redes neuronales y la computación evolutiva, como también de herramientas de la Ciencia de Datos, incluyendo técnicas de aprendizaje automático, estadísticas y visualización de datos, entre otros. Estas técnicas son capaces 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, Inteligencia Computacional, Ciencia de Datos, 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. 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 [1]. 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 [1]. El proceso de software es complejo y, como todos los procesos intelectuales y creativos, confía en que las personas tomen decisiones con criterio.

1.1. 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 cuenta 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.

1.2. 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 [2]. 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.

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 [2].

1.3. Inteligencia Computacional

La Inteligencia Computacional (IC) toma su inspiración de la naturaleza para desarrollar sistemas inteligentes basados en computadoras. Los pilares principales que componen a la IC son las Redes Neuronales, que modelan aspectos del funcionamiento del cerebro, los sistemas difusos, que modelan la forma en la que las personas describen el mundo a su alrededor, y computación evolutiva, que modela aspectos sobre la variación y selección natural en la bioesfera [3]. Estos múltiples métodos no son competitivos entre sí, sino que son complementarios y pueden ser utilizados juntos para resolver un problema dado.

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 IC, 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.

1.4. Ciencia de datos

La ciencia de datos, minería de datos, descubrimiento de conocimiento y otros términos similares, buscan agrupar un conjunto técnicas, herramientas y metodologías relacionadas con diferentes

aspectos del procesamiento de datos, tales como la recolección, la limpieza, el procesado, análisis y extracción de patrones implícitos a partir de los mismos [4].

Si bien una gran variedad de las técnicas utilizadas en el marco de este gran paraguas conceptual se pueden catalogar como parte de la inteligencia computacional, se nutre además de otras disciplinas como la estadística y la visualización de datos para extraer información valiosa y novedosa de repositorios de datos para asistir a la toma de decisiones [5].

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

2.1. 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 [6]. 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 [7]. En consecuencia, 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.

2.2. Mecanismos de captación de preferencias

Existen múltiples enfoques de optimización multiobjetivo, entre ellos los basados en preferencias [8]. 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, siendo los componentes analizables de estos mecanismos principalmente tres: la cantidad de interacciones con el usuario, qué información se le solicita, y en qué momento del proceso de optimización.

2.3. Ingeniería de Software basada en búsquedas conducida por los datos

Los datos producidos por los procesos de la ingeniería de software constituyen una fuente de conocimiento que puede ser explotada y utilizada para la resolución de los problemas, sirviendo de soporte a los algoritmos de optimización y actividades llevadas a cabo en el contexto de la Ingeniería de Software basada en Búsqueda (ISBB) [9], [10]. Son ejemplos de esta sinergia la comprensión de las preferencias de los tomadores de decisiones cuando estas no se encuentran explícitas a partir de las interacciones de los mismos con los algoritmos de búsqueda, o la posibilidad de explicar los resultados obtenidos por los algoritmos de optimización para su mejor comprensión y análisis.

En consecuencia, se propone la integración de algoritmos y técnicas del área del análisis de datos en los mecanismos utilizados para la exploración de las soluciones de la ingeniería de software basada en búsqueda para incrementar la usabilidad de los mismos.

2.4. 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 herramientas software que soporten la toma de decisiones, que sean usables por administradores no expertos en IC. Es razonable suponer que si esta herramienta se integra a herramientas que la industria ya utiliza tendrá mayores posibilidades de ser incorporada.

2.5. Paradigmas de Cómputo Alternativos

Cada vez se requiere mayor poder de procesamiento para abordar los distintos modelos de decisión descriptos y su consecuente análisis. Es por eso que se propone la utilización de paradigmas de computación paralela y cuántica (simulada) para suplir esta necesidad creciente de poder de cómputo. La computación paralela se ha aplicado ampliamente en el campo de la IC [11] y la cuántica posee como una de sus fortalezas la resolución de problemas de optimización mediante el algoritmo de temple cuántico y otros métodos relacionados [12].

3. 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 [13], 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 [14]. También se ha resuelto el problema de priorización de requerimientos de

software utilizando relaciones de preferencia difusa [15]. 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. En esta línea un primer prototipo de una aplicación para la exploración de frentes de Pareto ha sido construido haciendo uso de la librería Dash para Python 3.7. Este prototipo incorpora la posibilidad de carga de las soluciones de los algoritmos de búsqueda, su procesamiento mediante algoritmos de aprendizaje no supervisado y su presentación en un lenguaje comprensible para el tomador de decisiones haciendo uso de descriptores estadísticos y herramientas de visualización de información. Todo este proceso se puede realizar de forma iterativa, facilitando la exploración de las soluciones y consolidando las preferencias del tomador de decisiones. Se busca continuar con el desarrollo y validación de este prototipo software. 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.

4. FORMACIÓN DE RECURSOS HUMANOS

La investigación presentada constituye las líneas fundacionales de un nuevo grupo de investigación dentro de la UTN-FRCU, el GIICIS. 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 han defendido exitosamente dos prácticas profesionales supervisadas.

5. BIBLIOGRAFÍA

- [1] I. Sommerville, *Software Engineering*, 9.^a ed. Pearson Education, Inc., 2011.
- [2] F. S. Hillier y G. J. Lieberman, *Introducción a la Investigación de operaciones*. McGRAW-HILL/INTERAMERICANA EDITORES, S.A. DE C.V., 2010.
- [3] J. M. Keller, D. Liu, y D. B. Fogel, *Fundamentals of computational intelligence: Neural networks, fuzzy systems, and evolutionary computation*. John Wiley & Sons, 2016.
- [4] C. C. Aggarwal, *Data Mining*. Cham: Springer International Publishing, 2015.
- [5] J. Han, M. Kamber, y J. Pei, *Data Mining: Concepts and Techniques*. Elsevier Inc., 2012.
- [6] 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.
- [7] 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.
- [8] T. N. Ferreira, S. R. Vergilio, y J. T. de Souza, «Incorporating user preferences in search-based software engineering: A systematic mapping study», *Information and Software Technology*, vol. 90, pp. 55-69, 2017.
- [9] V. Nair *et al.*, «Data-Driven Search-based Software Engineering», en *2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR)*, 2018.
- [10] T. Colanzi, W. Assunção, ... S. V.-I. and, y U. 2020, «The Symposium on Search-Based Software Engineering: Past, Present and Future», *Information and Software Technology*, vol. 127, 2020.
- [11] B. S. Mishra y S. Dehuri, «Parallel computing environments: A review», *IETE Technical Review (Institution of Electronics and Telecommunication Engineers, India)*, vol. 28, n.º 3, pp. 240-247, may 2011.
- [12] A. Das y B. K. Chakrabarti, *Quantum annealing and related optimization methods*. Springer Science & Business Media, 2005.
- [13] 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.
- [14] 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.
- [15] 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.