

Algoritmos de Optimización aplicados a Documentación Arquitectónica con múltiples Stakeholders

Alumno

Christian Paulo Villavicencio

Director

Dr. J. Andrés Díaz Pace

**Facultad de Ciencias Exactas
Universidad Nacional del Centro de la Provincia de Buenos Aires
UNCPBA**



Agosto 2013

1. Motivación

Durante el desarrollo de software, la arquitectura de software y su documentación son factores importantes para el éxito del proyecto [1]. El documento de arquitecturas (SAD, por sus siglas en inglés) establece un canal de comunicación entre los stakeholders del proyecto. Se denomina *stakeholders* a las distintas personas u organizaciones involucradas en un sistema (por ejemplo: arquitectos, desarrolladores, gerentes, usuarios finales, entre otros). Cada stakeholder puede caracterizarse en función de sus intereses, necesidades, habilidades y tipos de documentos que maneja, los cuales están generalmente vinculados al rol que cumpla el stakeholder en el desarrollo del software. Dado que existen varios stakeholders para un SAD, suele ocurrir que los intereses y necesidades de cada uno de ellos estén en conflicto con los del resto.

Una problemática asociada al SAD es que generalmente apunta a un lector genérico, y por ende, tiende a ser extenso y proveer varios detalles técnicos; de modo que, por lo general, puede ocurrir que no siempre atienda las necesidades de información específicas de cada stakeholder e incluso lo sobrecargue con datos que no necesita o tiene interés en conocer en el momento de la consulta. Adicionalmente, el documentador debe planificar la generación de los contenidos del SAD y/o la edición de los ya existentes, de forma tal de satisfacer a los stakeholders y evitar el desperdicio de recursos documentando “más de lo necesario”.

En trabajos anteriores [2], se ha formulado la gestión del SAD como un problema de optimización combinatoria y se ha desarrollado una herramienta de asistencia denominada Arquitectonic que trabaja sobre un ambiente Wiki. El objetivo a optimizar consiste en seleccionar un conjunto de “tareas básicas de documentación” (o plan de documentación) sujeto a restricciones de costo y de consistencia, y tratando a su vez de satisfacer la mayoría de los intereses (preferencias) de los stakeholders. Si bien se han obtenido resultados preliminares satisfactorios, es necesario realizar un estudio más sistemático del problema de optimización en cuestión y los posibles algoritmos para resolverlo.

2. Trabajo Propuesto

En este trabajo se propone estudiar e implementar distintos algoritmos que pueden ser aplicados para resolver el problema de optimizar la generación de contenidos de un SAD. Una formulación básica del problema planteado como una variante del “problema de la mochila” se da en [3]. Una solución óptima es aquella que provee una lista de tareas a realizar para actualizar la versión actual del SAD que genere la mayor mejora en la utilidad de dicho SAD, teniendo en cuenta restricciones de tiempo y recursos. Dicha utilidad se mide en función de la

satisfacción de los stakeholders individuales respecto al grado de completitud del SAD que se logró con la actualización.

Los datos vinculados a los stakeholders y a sus preferencias serán generados tomando como base el método de documentación “Views & Beyond” desarrollado por el SEI [4], que incluye un conjunto de templates para el SAD y, además, proporciona guías que ayudan a relacionar vistas arquitectónicas predeterminadas (por ejemplo: módulos, componentes y conectores, despliegue) con intereses de tipos de stakeholders.

La investigación (y eventual generalización) del problema de la mochila aplicado a la gestión del SAD con múltiples stakeholders implica considerar tanto algoritmos exactos como aproximados (heurísticas), debido a los posibles costos en términos de complejidad computacional. En este sentido, es interesante estudiar tradeoffs entre: optimalidad de la solución, tiempo de ejecución del algoritmo, y objetivos de satisfacción de cada uno de los stakeholders.

En particular, una técnica interesante y aplicable en ese contexto son los algoritmos genéticos. Un algoritmo genético [5] es un tipo de algoritmo que basa su funcionamiento en la evolución biológica ya que hace evolucionar una población de individuos sometiéndola a “eventos aleatorios” (cruces genéticos y mutaciones) semejantes a los que ocurren en la evolución biológica, así como también a una selección, de acuerdo con algún criterio, que servirá para determinar cuáles son los individuos más adaptados (los sobrevivientes) y cuáles son los menos aptos (los descartados). Este tipo de algoritmos resulta interesante puesto que si se mapea correctamente los elementos del problema planteado es posible encontrar soluciones cercanas a la óptima si tener una complejidad computacional elevada como la que tendría, por ejemplo, un backtracking, aunque todo dependerá de cómo esté pensado e implementado el algoritmo.

Una vez implementados los distintos tipos de algoritmos serán probados con un conjunto de SAD’s predeterminado y se analizarán los resultados obtenidos.

3. Cronograma de Trabajo

Las actividades principales a desarrollar son las siguientes:

- Relevamiento bibliográfico en las áreas de arquitecturas de software, documentación, teoría de stakeholders, y algoritmos de optimización. Tiempo estimado: 1 mes.
- Implementación de algoritmos de optimización (genéticos, exactos, etc.) adecuados para el problema. Tiempo estimado: 2 meses.
- Integración de los algoritmos de optimización con la herramienta Architectonic. Implementación de funcionalidades adicionales que puedan derivarse de este trabajo (por ejemplo: carga de datos de los stakeholders, configuración, mejoras en la presentación de resultados, etc.).
Tiempo estimado: 1 mes.

- Ejecución de pruebas con datasets y distintas configuraciones de parámetros en los algoritmos. Ajuste del comportamiento de los algoritmos. Tiempo estimado: 2 meses.
- Redacción de un informe del trabajo realizado. Tiempo estimado: 4 meses - en paralelo con las actividades anteriores.

4. Referencias

- [1] J. Andres Diaz-Pace, Matias Nicoletti, Silvia Schiaffino. "Towards Software Architecture Documents matching Stakeholders' Interests". Presentado en ADNTIIC 2011. Berlin, pp. 176-185.
- [2] Luis Emiliano Sanchez y Juan Armella. "Una herramienta para asistir la documentación de arquitecturas de software". A ser publicado en Proceedings de EST 2013 (16 Concurso de Trabajos Estudiantiles), 42 JAIIO - SADIO. Córdoba, Argentina. Septiembre 2013.
- [3] J. Andres Diaz-Pace, Matias Nicoletti, Silvia Schiaffino, Emiliano Sánchez, Christian Villavicencio (2013). "A Stakeholder-centric Optimization Strategy for Architectural Documentation". A ser presentado en 3rd International Conference on Model & Data Engineering (MEDI 2013).
- [4] Clements, Bachmann, Bass, Garlan, Ivers, Little, Merson, Nord, Stafford. "Documenting Software Architectures: Views and Beyond (Second Edition)". October 15, 2010. ISBN-10: 0321552687. ISBN-13: 978-0321552686.
- [5] Goldberg, D. E., (2007), Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., 2007, ISBN: 0201157675.
- [6] Yuanyuan Zhang, Mark Harman, Anthony Finkelstein, S. Afshin Mansouri. "Comparing the Performance of Metaheuristics for the Analysis of Multi-Stakeholder Tradeoffs in Requirements Optimisation". Publicado en la revista "Information and Software Technology", volumen 53, número 7, páginas 761-773. Julio 2011.