



Title: Ingeniería de Software Basada en Búsqueda en Líneas de Productos de Software

Authors: TRUJILLO-TZANAHUA, Guadalupe Isaura, CORTÉS-VERDÍN, Karen y JUÁREZ-MARTÍNEZ, Ulises

Editorial label ECORFAN: 607-8695
BCIERMMI Control Number: 2020-04
BCIERMMI Classification (2020): 211020-0004

Pages: 15
RNA: 03-2010-032610115700-14

ECORFAN-México, S.C.
143 – 50 Itzopan Street
La Florida, Ecatepec Municipality
Mexico State, 55120 Zipcode
Phone: +52 1 55 6159 2296
Skype: ecorfan-mexico.s.c.
E-mail: contacto@ecorfan.org
Facebook: ECORFAN-México S. C.
Twitter: @EcorfanC

www.ecorfan.org

Holdings		
Mexico	Colombia	Guatemala
Bolivia	Cameroon	Democratic
Spain	El Salvador	Republic
Ecuador	Taiwan	of Congo
Peru	Paraguay	Nicaragua

Introduction

Methodology

Results

Annexes

Conclusions

References

Introducción

- Las empresas deben innovar sus producto o introducir nuevos para poder permanecer en el mercado.
- Existe una necesidad de evolucionar para poder satisfacer las necesidades del mercado.
- Las empresas adoptan nuevos enfoques para generación automática y personalización en masa del software: Líneas de Productos de Software (LPS) o Multilíneas de Productos de Software (MPL).
- Los enfoques de LPS y MPL buscan generar productos a través de la reutilización y de la automatización de procesos.
- Por otra parte, surge la necesidad de explorar nuevas técnicas para la resolución de problemas y toma de decisiones complejas en el desarrollo.

Introducción

- Así, surge la SBSE (*Search-Based Software Engineering*) que consiste en la formulación de problemas de búsqueda y aplicación de optimización para su solución en el contexto del software.
- Dado que la gestión de MPL implica múltiples variantes, no es factible configurar, implementar o probar todos los productos.
- Por lo tanto, se presenta la investigación y aplicación una variante del problema de la mochila para la configuración de productos de una MPL.
- Se formula el problema de configuración como un problema de la mochila multidimensional y se resuelve con una técnica de SBSE.

Metodología

La aplicación de SBSE en el desarrollo de una MPL consta de 5 etapas:

1) Identificación del problema

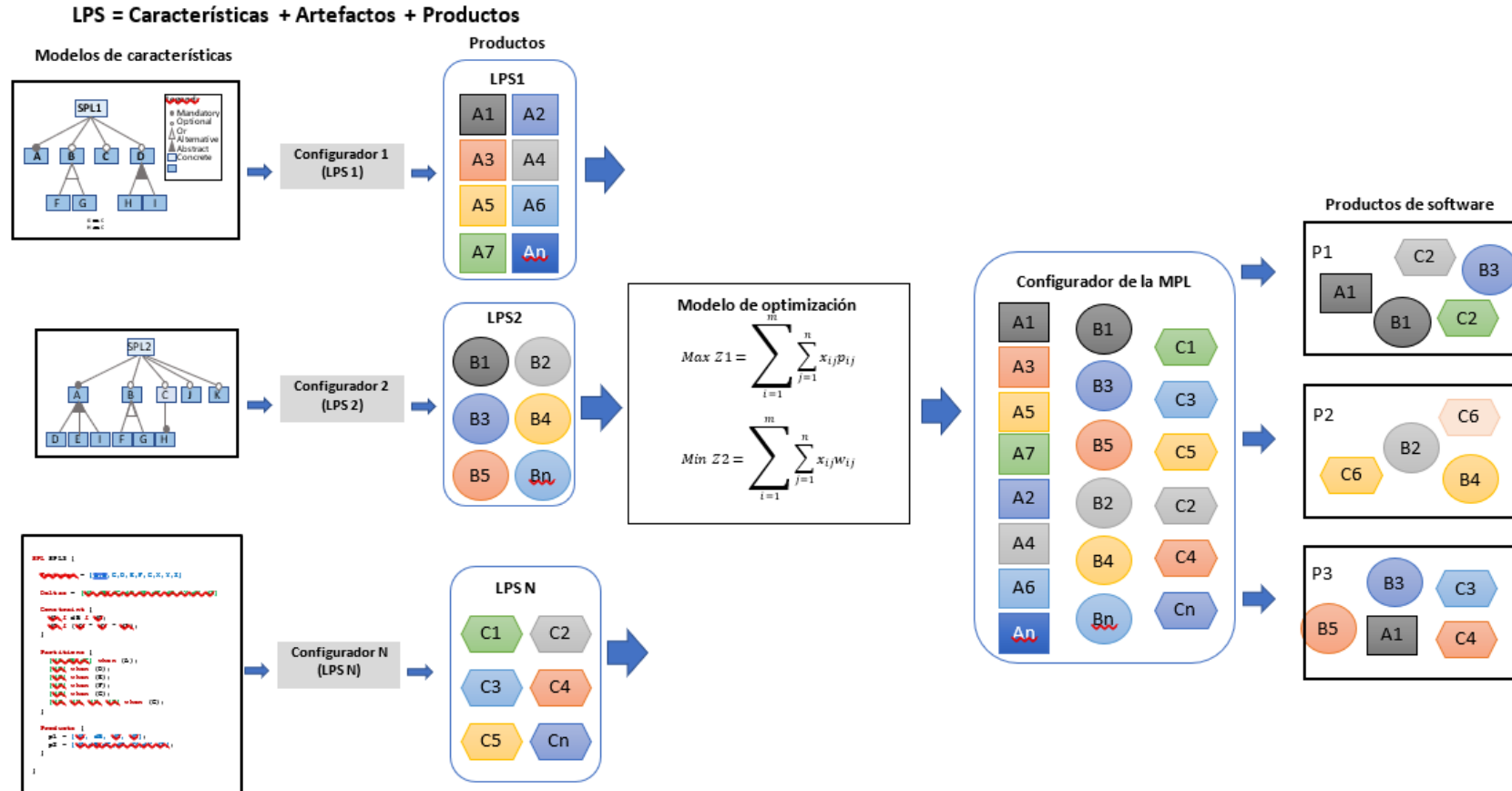
- 1) Problema de optimización combinatoria no lineal y multi-objetivo
- 2) Número finito de Líneas de Productos de Software
- 3) Cantidad definida de características, cada característica tiene diversos valores y pertenece a diferentes modelos de características
- 4) Problema de tipo NP por lo que se elige el modelo de la mochila para su solución (*Multiple-Choice Knapsack Problem*)

2) Formalización del problema

Dado m modelos de características y una especificación de requisitos, una configuración de productos es la selección de características subdivididas en m LPS.

Metodología

En consecuencia, si un producto de software en particular se define por n características, con cada característica $f_{ij} \in N_i$, la cual tiene una recompensa de selección P_{ij} por reutilización y un costo de desarrollo w_{ij} , es posible modelar el problema de configuración de productos en una MPL como un caso especial de MMKP.



Metodología

3) Definición del modelo de optimización

$Maxz_1, Minz_2$

Función objetivo que selecciona características de cada LPS con el máximo beneficio (reutilización):

$$Max Z = \sum_{i=1}^m \sum_{j=1}^n x_{ij} p_{ij}$$

Mientras que el costo del desarrollo no exceda la restricción del presupuesto:

$$\sum_{i=1}^m \sum_{j \in N} w_{ij} x_{ij} \leq W$$

Metodología

Función objetivo que selecciona características de cada LPS con el costo mínimo de desarrollo

$$\text{Min } Z = \sum_{i=1}^m \sum_{j=1}^n x_{ij} w_{ij}$$

Mientras que la reutilización debe exceder el límite de reutilización:

$$\sum_{i=1}^m \sum_{j \in N} w_{ij} p_{ij} \leq P$$

$$\sum x_{ij} = 1, \forall i \in \{1, 2 \dots n\}$$

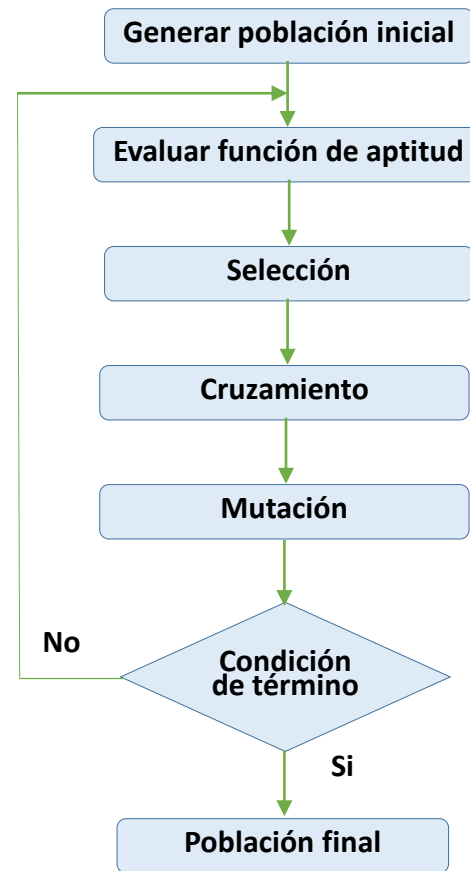
Metodología

Datos	
F	Característica
N	Número de características
M	Número de LPS = Número de modelos de características
I	Índice de una característica
J	Índice de una LPS
w_{ij}	Costo de desarrollo de la característica i si se asigna a la LPS j
p_{ij}	Beneficio o puntaje de la característica i si se asigna a la LPS j por reutilización
Variables	
x_{ij}	Variable igual a 1 si la característica i se asigna a la LPS j; 0 en caso contrario
Restricciones	
W	Presupuesto disponible
P	Valor límite mínimo de reutilización requerido
MA	Conjunto de características obligatorias
O	Conjunto de características opcionales
XOR	Conjunto de todas las características alternativas exclusivas
OR	Conjunto de todas las características alternativas no exclusivas

Metodología

4) Solución empleando una técnica de SBSE (algoritmo genético)

El problema fue resuelto de forma multi-objetivo mediante un algoritmo genético desarrollado en Python



Resultados

El algoritmo se implementó en Python y se probó en una PC con procesador Intel ® Core ™ i7 2.8 GHz, 16 GB y S.O. Microsoft Windows 10.

Para evaluar el rendimiento del modelo se utilizaron dos métricas: valor de conveniencia para la reutilización y costo.

El modelo de optimización se ejecutó 10 veces bajo los mismos parámetros.

Para cada ejecución, los valores iniciales de las variables de optimización se cambiaron para iniciar la búsqueda desde diferentes puntos y observar si la evolución se comporta de la misma manera.

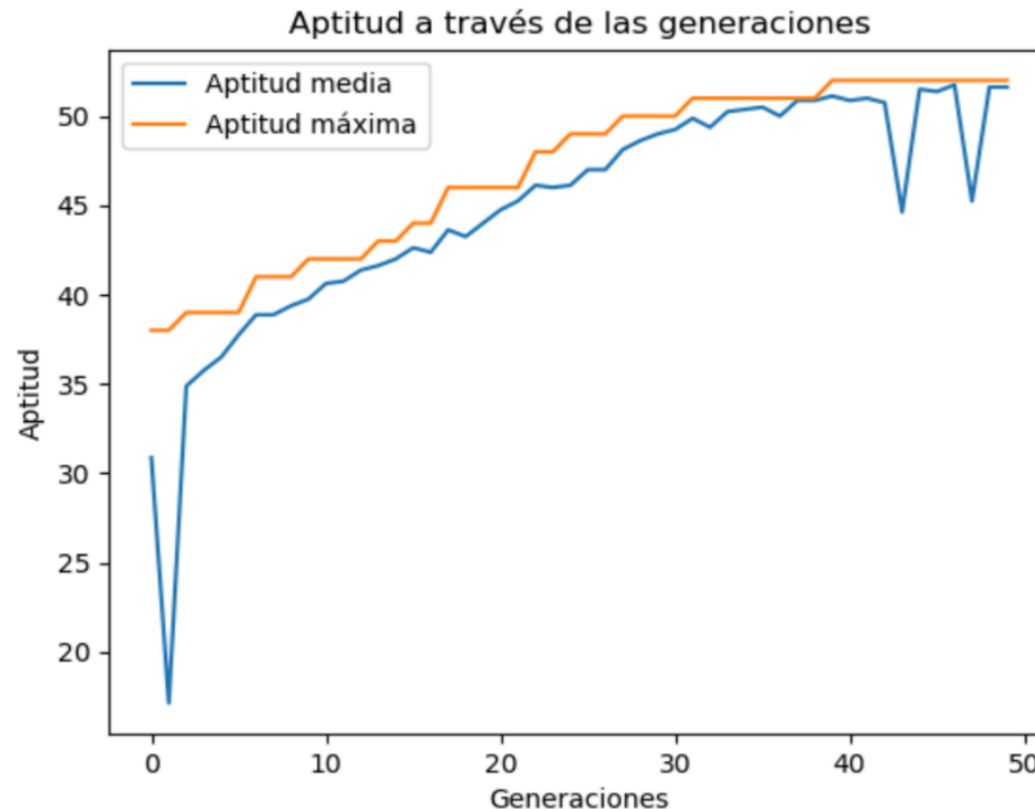
Las 10 ejecuciones dieron resultados mayores a los requeridos, por lo que se concluye que el modelo se comporta adecuadamente para el escenario propuesto.

Resultados

Población final o selección óptima de del conjunto de selecciones óptimas:

1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	4	4	
1	1	1	1	1	1	0	1	0	1	1	0	1	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1	0	0	1	0	0	0	0	0	0	1	1	1	1

Ejemplo de los Frentes de Pareto obtenidos en las pruebas realizadas al algoritmo genético:



Conclusiones

- Existe demanda de nuevos enfoques para la configuración y generación de sistemas más complejos, evolución y mantenimiento como las Líneas de Productos de Software y Multilíneas de Productos de Software.
- La SBSE, en el caso de las LPS, es un medio que guía y ayuda a los ingenieros de aplicaciones, especialmente en configuración, depuración o pruebas.
- En este caso, el algoritmo genético en Python facilitó la selección y combinación de características de múltiples LPS, dando lugar a nuevos productos.

Conclusiones

- Así, mediante técnicas de SBSE se pueden diseñar productos que mejor se adapten los requisitos, en base a objetivos de optimización de costo y reutilización.
- Al optimizar la selección de características en la fase de configuración, se logran beneficios en la calidad y desempeño del producto así como mayor beneficio económico al ajustar el costo de producción.
- Trabajos futuros: 1) Agregar variables adicionales al modelo de optimización para abarcar otros productos de software y, poder así, abarcar segmentos del mercado y 2) utilizar el caso de estudio y resolver el modelo de optimización con otras técnicas de SBSE

Referencias

- Alsawalqah, H. I., Kang, S., & Lee, J. (2014). A method to optimize the scope of a software product platform based on end-user features. *Journal of Systems and Software*, 98, 79–106. <https://doi.org/10.1016/j.jss.2014.08.034>
- Asadi, M., Soltani, S., Gasevic, D., Hatala, M., & Bagheri, E. (2014). Toward automated feature model configuration with optimizing non-functional requirements. *Information and Software Technology*, 56(9), 1144–1165. <https://doi.org/10.1016/j.infsof.2014.03.005>
- Clements, P. C., & Northrop, L. (2001). *Software Product Lines: Practices and Patterns*. Retrieved from <http://dl.acm.org/citation.cfm?id=501065>
- Connolly, D., Martello, S., & Toth, P. (1991). Knapsack Problems: Algorithms and Computer Implementations. *The Journal of the Operational Research Society*, 42(6), 513. <https://doi.org/10.2307/2583458>
- dos Santos Neto, P. de A., Britto, R., Rabêlo, R. de A. L., Cruz, J. J. de A., & Lira, W. A. L. (2016). A hybrid approach to suggest software product line portfolios. *Applied Soft Computing Journal*, 49, 1243–1255. <https://doi.org/10.1016/j.asoc.2016.08.024>

Referencias

Holl, G., Grünbacher, P., & Rabiser, R. (2012). A systematic review and an expert survey on capabilities supporting multi product lines. *Information and Software Technology*, 54(8), 828–852. <https://doi.org/10.1016/j.infsof.2012.02.002>

Lienhardt, M., Damiani, F., Donetti, S., & Paolini, L. (2018). Multi Software Product Lines in the Wild. *Proceedings of the 12th International Workshop on Variability Modelling of Software-Intensive Systems - VAMOS 2018*, 89–96. <https://doi.org/10.1145/3168365.3170425>

Tan, L., Lin, Y., Ye, H., & Zhang, G. (2013). Improving product configuration in software product line engineering. *Conferences in Research and Practice in Information Technology Series*, 135, 125–134. Retrieved from <http://crpit.com/confpapers/CRPITV135Tan.pdf>



ECORFAN®

© ECORFAN-Mexico, S.C.

No part of this document covered by the Federal Copyright Law may be reproduced, transmitted or used in any form or medium, whether graphic, electronic or mechanical, including but not limited to the following: Citations in articles and comments Bibliographical, compilation of radio or electronic journalistic data. For the effects of articles 13, 162,163 fraction I, 164 fraction I, 168, 169,209 fraction III and other relative of the Federal Law of Copyright. Violations: Be forced to prosecute under Mexican copyright law. The use of general descriptive names, registered names, trademarks, in this publication do not imply, uniformly in the absence of a specific statement, that such names are exempt from the relevant protector in laws and regulations of Mexico and therefore free for General use of the international scientific community. BCIERMMI is part of the media of ECORFAN-Mexico, S.C., E: 94-443.F: 008- (www.ecorfan.org/ booklets)