

Exploración *a posteriori* Basada en Clustering del Frente Pareto- óptimo aplicado al Problema del Próximo Lanzamiento

Giovanni Daián Rottoli, Carlos Casanova, Esteban Schab, Anabella De Battista
Grupo de Investigación sobre Inteligencia Computacional e Ingeniería de Software
Departamento Ingeniería en Sistemas de Información
Universidad Tecnológica Nacional, Facultad Regional Concepción del Uruguay
Ing. Pereira 676 - Concepción del Uruguay - E.R. - Argentina.
{rottolig, casanovac, schabe, debattistaa}@frcu.utn.edu.ar

Resumen

La ingeniería de software basada en búsqueda propone la formulación de problemas de la ingeniería de software como problemas de optimización, generalmente con múltiples objetivos. Para estos problemas existen múltiples soluciones que deben ser analizadas por los tomadores de decisiones para seleccionar aquella que más se adecúe a sus necesidades, pudiendo este no tener preferencias previas sobre la mejor solución a ser seleccionada. Para ello, este artículo propone un acercamiento basado en agrupamiento que permite la exploración a posteriori del espacio de soluciones, describiendo las mismas no solo utilizando el valor de sus objetivos, sino también la configuración de los elementos que conforman las mismas. Se presenta una prueba de concepto del método propuesto utilizando el frente Pareto-Óptimo solución de una instancia del Problema del Próximo Lanzamiento.

Introducción

La Ingeniería de Software estudia la problemática relacionada al desarrollo, la operación y el mantenimiento del software desde un punto de vista científico. Sin embargo, es usual encontrar casos en los que se utiliza un enfoque artesanal de desarrollo de software, dependiente en gran medida de la experiencia de los actores del proceso para lograr productos que cumplan con requerimientos y características varias. En ese contexto el líder de proyecto enfrenta numerosas dificultades tanto para lograr dichas características, como para construir, operar o mantener el software y que éste constituya un producto de calidad. Las dificultades radican en que estos procesos tienden a estar caracterizados por un gran número de restricciones y objetivos interrelacionados que compiten entre sí.

Para abordar este tipo de problemas, la Ingeniería de Software Basada en Búsqueda (ISBB) o *Search Based Software Engineering* (SBSE) propone reformularlos

como problemas de optimización, usualmente de naturaleza combinatoria y con múltiples objetivos, para luego aplicar algoritmos de búsqueda que sean capaces de encontrar una solución de compromiso. En este sentido, la ISBB es una disciplina que busca brindar soporte a los ingenieros de software para lograr productos de calidad a través de la búsqueda de soluciones óptimas para este tipo de problemas de optimización [1].

Problemas de optimización multi-objetivo

Respecto de los algoritmos de búsqueda, la ISBB ha propuesto desde sus comienzos la aplicación de algoritmos de tipo metaheurístico [1]. Esta propuesta se sustenta en varias características que son inherentes a los problemas que surgen en el contexto de la IS: espacios de búsqueda combinatorios, necesidad de balancear múltiples objetivos que compiten entre sí, existencia de incertidumbre o imprecisión, entre otros. Los algoritmos metaheurísticos resultan adecuados en este contexto por brindar estrategias de búsqueda genéricas que encuentran soluciones en tiempos razonables, sin requerir el cumplimiento de propiedades como linealidad o convexidad, aunque con la imposibilidad de asegurar cuán lejos se está de la solución óptima real.

En los problemas de optimización multi-objetivo varios atributos se emplean como funciones objetivo que compiten entre sí, definiendo sobre el espacio de soluciones un orden parcial, la relación de **dominancia**, donde existen pares de soluciones que no son comparables en principio. Por ejemplo, en un problema con dos objetivos, dos soluciones donde la primera es mejor en el primer objetivo y peor en el segundo respecto de la solución restante, resultan no comparables.

El conjunto de soluciones en la que ninguna domina a las demás se conoce como Frente Pareto-Óptimo, o simplemente Frente de Pareto. Dos ejemplos de algoritmos ampliamente utilizados que buscan frentes de Pareto basándose en la relación de dominancia son NSGA-II [2] y SPEA2 [3].

Enfoques basados en preferencias

En general el tomador de decisiones necesita seleccionar sólo una solución del Frente de Pareto. Con el objetivo de facilitar este proceso de selección surgen los Algoritmos Basados en Preferencias (ABP) [4], los cuales incorporan las preferencias, intuición o emoción del tomador de decisiones en el proceso de optimización. Para esto se hace uso de mecanismos que consultan al tomador de decisiones para extraer información que permita guiar la búsqueda hacia una región de interés. Según en qué etapa del algoritmo se incorporan las preferencias, estos mecanismos se clasifican en *a priori*, interactivos y *a posteriori*.

Los mecanismos de incorporación de preferencias *a priori* se utilizan antes de iniciar el proceso de generación de soluciones (búsqueda). Dentro de esta clasificación pueden encontrarse los Procesos de Jerarquía Analítica [5], utilizados para encontrar los pesos asociados a cada objetivo y componer una única función objetivo.

Los mecanismos interactivos de incorporación de preferencias, por otro lado, se utilizan durante la búsqueda, es decir, mientras se generan las soluciones intermedias, embebidos en el algoritmo de búsqueda. En este punto es de especial interés el estudio de cuántas veces se requiere de la atención del tomador de decisiones. El ABP interactivo más utilizado en la ISBB es el IGA (*Interactive Genetic Algorithm*) [4], encontrándose aplicaciones en Ingeniería de Requerimientos, Refactorización, Remodularización y Diseño de Interfaces de Usuario, entre otros.

Por último, los mecanismos *a posteriori* se utilizan una vez finalizado el proceso en cuestión, y generalmente se reducen a búsquedas dentro de frentes Pareto-óptimos (o aproximaciones de éstos). Este tipo de enfoques es el menos utilizado en el estado del arte [4].

Aportes y contribuciones

En este trabajo se propone un mecanismo de captación de preferencias *a posteriori* mediante el empleo de agrupamiento jerárquico de soluciones complejas y análisis estadístico sobre su estructura, con el objetivo de auxiliar al tomador de decisiones en la selección de una única solución que le sea satisfactoria. La propuesta es aplicada al frente de Pareto de una instancia del Problema del Próximo Lanzamiento (*Next Release Problem*, NRP) con dos objetivos, un problema de Ingeniería de Requerimientos.

El resto del trabajo se estructura como sigue. Se describe formalmente el NRP y se presenta el conjunto de datos utilizado. Luego, se describe formalmente el mecanismo de captación de preferencias, lo cual incluye las funciones de similitud necesarias para aplicar el algoritmo de agrupamiento jerárquico, seguidas de los descriptores utilizados para caracterizar los grupos y la forma en la que el tomador de decisiones puede visualizar

los resultados. Se presentan los resultados de aplicar el mecanismo propuesto y se discuten las ventajas y desventajas del mismo contrastándolo con otros trabajos. Finalmente, se presentan las conclusiones más relevantes y las referencias utilizadas.

El Problema del Próximo Lanzamiento

Un aspecto clave de cualquier proyecto de desarrollo grande es determinar un conjunto de requerimientos apropiado para las partes interesadas (*stakeholders*). El problema del próximo lanzamiento, originalmente propuesto por Bagnall, Rayward-Smith y Whitley (2001) [6], formaliza esta cuestión y consiste en hallar un subconjunto de requerimientos que maximice un atributo deseado, tal como el beneficio, sujeto a una cota superior de algún otro atributo, usualmente el costo. Eventualmente, también puede estar restringido por dependencias entre los requerimientos: precedencia, simultaneidad, elegir no más que k de n requerimientos, entre otros.

Este problema es reducible al problema de la mochila (*knapsack problem*) [6] y constituye, por lo tanto, un problema de optimización combinatoria NP-hard. En la versión bi-objetivo del NRP [7], la restricción de cota superior del costo se transforma en un segundo objetivo, el cual se encuentra en conflicto con el primero.

Es posible formular el problema NRP como un modelo de Programación Lineal Entera, como sigue [8]:

Sean $X = [x_1, x_2, \dots, x_n]$ y $Y = [y_1, y_2, \dots, y_m]$ los vectores binarios de decisión que representan la inclusión de los requerimientos, del 1 al n , y la satisfacción de los *stakeholders*, del 1 al m . Sea $C = [c_1, c_2, \dots, c_n]$ el vector de costos asociados a los requerimientos, y $B = [b_1, b_2, \dots, b_m]$ los beneficios asociados a cada *stakeholder*. Sea P la **relación de precedencia** formada por los pares (i, j) , tales que el requerimiento i debe ser seleccionado siempre que se selecciona el requerimiento j . Sea además I la **relación de interés** formada por los pares (i, k) , donde el *stakeholder* k ha solicitado el requerimiento i . Así, se puede modelar el problema NRP como sigue:

$$\begin{aligned} \max f(Y) &= \sum_{i=1}^m b_i \cdot y_i \\ \min g(X) &= \sum_{j=1}^n c_j \cdot x_j \end{aligned}$$

sujeto a

$$x_i \geq x_j, \forall (i, j) \in P$$

$$x_i \geq y_k, \forall (i, k) \in I$$

$$X \in \{0,1\}^n, Y \in \{0,1\}^m$$

Descripción del problema

Una vez obtenido el frente de Pareto para un problema de optimización multi-objetivo, el tomador de decisiones debe seleccionar una solución de dicho frente para implementar en el contexto en el que se encuentra, para lo cual debe compararlas y optar por aquella que más se adecúe a sus preferencias.

Generalmente esta comparativa se realiza a partir de la evaluación de los valores de la función objetivo de cada una de las soluciones, cuestión que resulta tediosa al estar el frente compuesto por numerosas opciones distintas. Por otro lado, las soluciones son complejas *per se*, consistiendo en múltiples elementos que se interrelacionan entre sí (e.g. múltiples requerimientos interdependientes en el NRP), por lo que no es trivial la búsqueda de soluciones con base en la configuración misma de la solución. Adicionalmente, el tomador de decisiones puede no tener en claro sus preferencias en el momento de la exploración del frente, por lo que es menester el brindar herramientas de información para dirigir sus esfuerzos.

Freire et al. (2019) [9] menciona que no existen formas de facilitar el análisis de soluciones complejas en el frente de Pareto, como es en el caso del NRP. Además, propone la utilización de algoritmos de agrupamiento utilizando como medida de similitud entre las soluciones la distancia euclídea entre los valores de los objetivos, mas no analiza la similitud de las soluciones de manera estructural, considerando las características de configuración de las mismas.

En este trabajo, entonces, se propone la utilización de algoritmos de *clustering* para el agrupamiento de soluciones complejas para un problema de la ISBB que aporta conocimiento y herramientas visuales al tomador de decisiones para optar por las alternativas más adecuadas para su situación particular.

Solución propuesta

Considerando la ausencia de información previa sobre las preferencias del tomador de decisiones, se propone la utilización del modelo que se ilustra en la Figura 1.

El proceso comienza con la obtención y preparación del frente de Pareto. Este paso dependerá del problema en cuestión, de la representación de las soluciones, y del algoritmo utilizado para hallarlas. Cabe remarcar que el conjunto de soluciones no necesariamente deben ser las soluciones óptimas reales, sino que es posible utilizar un conjunto de soluciones cuasi-óptimas o aproximadas.

A partir de estas soluciones, se debe codificar cada una de las mismas de forma tal que conste por cada una de ella los valores de sus objetivos asociados y las características de la solución misma. Por ejemplo, en caso del problema del próximo lanzamiento, tanto el costo como el beneficio obtenido son dos atributos de la tabla de datos, así como

los requerimientos seleccionados para su implementación y el conjunto de stakeholders satisfechos.

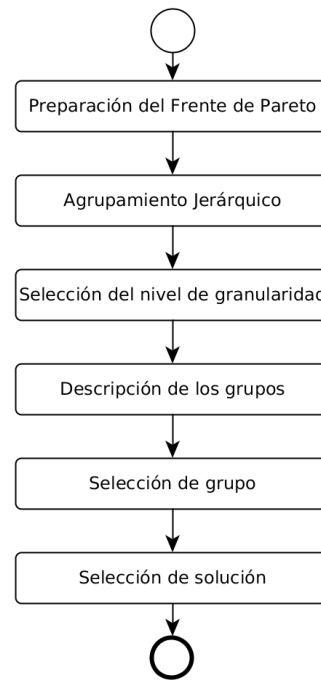


Figura 1. Proceso de exploración *a posteriori* de frentes de Pareto utilizando agrupamiento de datos

Posteriormente, se aplica un algoritmo de agrupamiento jerárquico sobre el frente de Pareto preparado para acotar la búsqueda a regiones de interés [10]. Para esto es necesario la utilización de una medida de similitud que contemple tanto los valores de los objetivos como también la configuración de la solución. Esta medida, entonces, dependerá del problema abordado. En la próxima sección, Prueba de Concepto, se propone una medida para el Problema del Próximo Lanzamiento.

La característica principal del algoritmo de agrupamiento jerárquico es que no brinda inicialmente un conjunto de particiones del conjunto de datos, como hacen algoritmos basados en particiones o en densidades, permitiendo observar primeramente las relaciones de similitud (con herramientas de visualización como dendogramas) y pudiendo decidir luego cuál es el parámetro de granularidad adecuado para materializar los grupos. Esta característica resulta deseable en el contexto de una búsqueda *a posteriori* debido a que el tomador de decisiones puede observar el comportamiento del frente antes de acotarlo.

Una vez obtenidas las particiones del frente de Pareto con base en la medida de similitud adecuada, es posible describir los grupos mediante descriptores estadísticos o bien mediante un proceso de descubrimiento de reglas de pertenencia a grupos, haciendo uso de algoritmos de aprendizaje automático [11]. Esta descripción le permite al tomador de decisiones explorar las características de cada una de las secciones del frente y decidir por ahondar en una de las secciones que le resulte de mayor interés.

Una vez reducido el número de soluciones para analizar, es posible seleccionar una para su implementación sin incurrir en fatiga para el tomador de decisiones. Sin embargo, si el grupo seleccionado para su exploración aún posee un número considerable de soluciones, es posible repetir la estrategia hasta que la cantidad de opciones sea manejable.

Mediante este procedimiento se acota la búsqueda a una región del frente de Pareto brindando información no solo sobre las características de las soluciones en función de los valores de los objetivos, sino que además se contempla la configuración de las mismas en el contexto del problema.

Prueba de concepto

Para ilustrar la solución propuesta en la sección anterior, este apartado aborda una prueba de concepto utilizando un frente de Pareto de una instancia del Problema del Próximo Lanzamiento, algoritmos de agrupamiento jerárquico, descriptores estadísticos y herramientas de visualización para asistir al tomador de decisiones en la selección de una solución adecuada para su situación particular.

Conjunto de datos

Se dispone de un frente de Pareto exacto con 385 soluciones de una instancia del problema del próximo lanzamiento con 140 requerimientos y 100 *stakeholders*. La instancia del problema ha sido estudiada por los autores previamente [12] y el frente de Pareto exacto se halló utilizando el enfoque de ϵ -restricciones descrito en [13]. Este conjunto de datos se encuentra en disponible en [14].

Por cada solución no dominada se detalla un identificador entero secuencial, el beneficio que se obtiene al implementar los requerimientos de la solución, el costo asociado a dicha implementación, un vector booleano *req* donde $req[i] = 1$ si el requerimiento *i* se incluye en la solución, y $req[i] = 0$ en caso contrario, y un vector booleano *stk* donde $stk[j] = 1$ si la solución satisface al *stakeholder* *j*, y $stk[j] = 0$ en caso contrario (Tabla 1).

Tabla 1. Atributos de las soluciones del frente Pareto-óptimo.

Atributo	Formato	Descripción
i	Entero	Identificador.
p	Real	Beneficio asociado a la solución.
c	Real	Costo de implementación de la solución.
req	Vector booleano	Requerimientos seleccionados para su implementación.
stk	Vector booleano	<i>Stakeholders</i> satisfechos con la implementación de la solución.

Procedimiento

Para la realización de la exploración del frente de Pareto se implementó un script en Python 3.7 de la distribución Anaconda, utilizando las librerías Numpy, Pandas, Scipy y Seaborn [15].

Agrupamiento de soluciones

Con el propósito de realizar un agrupamiento de las soluciones del frente de Pareto se ha tenido en cuenta la similitud de las mismas con base en sus atributos costo, beneficio, requerimientos y *stakeholders*.

En primer lugar, la similitud entre el costo y el beneficio se ha capturado mediante el uso de la distancia Euclídea entre estos atributos para las dos soluciones comparadas, tal como se observa en la ecuación 1.

Por otro lado, la similitud entre dos conjuntos de requerimientos (o conjuntos de *stakeholders*) se ha medido comparando la cantidad de requerimientos (o *stakeholders*) en común que estos poseen. Para esto, se hace uso del operador booleano XOR y se aplica el producto escalar del vector resultado consigo mismo (ecuaciones 2 y 3).

Es importante señalar que tanto los valores de costo y beneficio, como la distancias asociadas a los requerimientos y a los *stakeholders* deben ser escalados en el mismo intervalo, con el fin de evitar que uno de estos atributos sesgue los resultados del proceso de agrupamiento.

La similitud entre dos soluciones, entonces, se calcula como la suma algebraica de las distancias anteriormente mencionadas (Ecuación 4).

$$\delta_{obj}(x, y) = \sqrt{(c_x - c_y)^2 + (p_x - p_y)^2} \quad (1)$$

$$\delta_{req}(x, y) = [req_x \otimes req_y]^2 / |req_x| \quad (2)$$

$$\delta_{stk}(x, y) = [stk_x \otimes stk_y]^2 / |stk_x| \quad (3)$$

$$Dist(x, y) = \delta_{obj}(x, y) + \delta_{req}(x, y) + \delta_{stk}(x, y) \quad (4)$$

Descripción de los grupos

Por cada *cluster* hallado se extrajeron descriptores estadísticos. Se calculó el promedio y la desviación estándar de los valores de los atributos costo y beneficio de sus elementos miembros. Adicionalmente, se contó por cada requerimiento la frecuencia de consideración de los requerimientos, esto es, la cantidad de soluciones que consideraba incluir cada una de las características del software. Lo mismo se calculó para los *stakeholders*, contando por cada uno de ellos la cantidad de soluciones del grupo que lo satisface.

Visualización de resultados para exploración

Como soporte para la representación de los resultados parciales para la exploración de las soluciones del frente de Pareto, se utiliza un dendograma (diagrama de árbol) para el agrupamiento jerárquico y la selección del nivel de granularidad.

Adicionalmente, por cada uno de los grupos materializados se obtienen histogramas con las frecuencias de consideración de requerimientos y de satisfacción de *stakeholders*.

Segunda iteración

Se repitió el procedimiento sobre los resultados del *cluster* número 2, filtrando las soluciones del frente de Pareto que pertenecen a este grupo. La misma función de distancia fue utilizada, por lo que es posible utilizar el resultado del agrupamiento anterior y obtener la nueva partición solo de los datos seleccionados. Esto no requiere un cálculo extra.

Resultados

Primeramente, se obtuvo una segmentación de las soluciones del frente de Pareto considerado como entrada. Esta segmentación puede observarse en forma de árbol en el dendograma de la figura 2.

Como ya ha sido mencionado con anterioridad, la cantidad de soluciones del frente de Pareto no permite una visualización adecuada de su localización en el árbol; sin embargo es posible explorar con detalle las ramificaciones del mismo mediante el uso de *zooms* en las zonas de interés, cuestión que se ejemplifica en la figura 3.

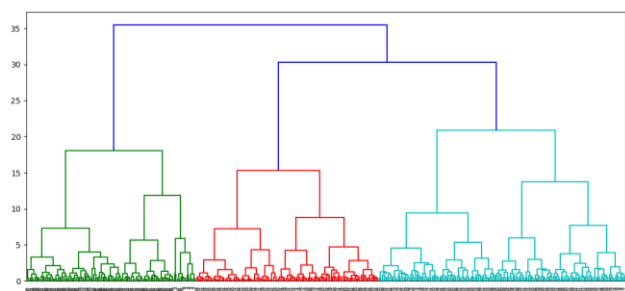


Figura 2. Dendograma obtenido del proceso de agrupamiento jerárquico.

Posteriormente, tras el particionamiento de la jerarquía obtenida en el paso anterior, se identificaron cuatro clusters, cantidad que fue determinada tras la exploración del dendograma mencionado. Los grupos resultantes poseen 109, 118, 75 y 84 soluciones.

Los descriptores referidos al costo y al beneficio para cada uno de los grupos se pueden observar en la Tabla 2, mientras que la frecuencia de consideración de los requerimientos y la frecuencia de satisfacción de los *stakeholders* se puede observar en las figuras 4 y 5 respectivamente.

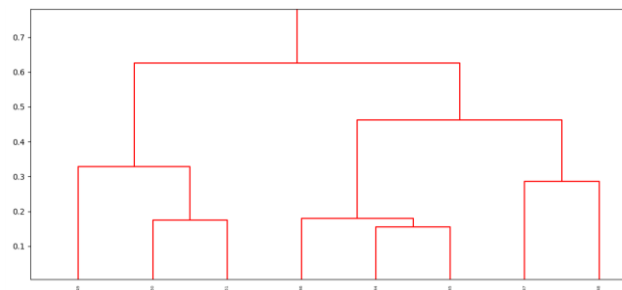


Figura 3. Dendograma de subárbol del agrupamiento jerárquico. Se observan nueve soluciones del frente de Pareto.

Tabla 2. Descripción de los atributos costo (c) y beneficio (p) por cada grupo

Grupo	Cantidad	Attr	Descriptor	Valor
1	109	p	Prom	2384,7
			s.d.	229,0
		c	Prom	576,0
			s.d.	64,6
2	118	p	Prom	474,2
			s.d.	241,1
		c	Prom	83,6
			s.d.	49,4
3	75	p	Prom	1145,8
			s.d.	135,7
		c	Prom	241,7
			s.d.	34,3
4	84	p	Prom	1689,3
			s.d.	187,4
		c	Prom	386,3
			s.d.	52,7

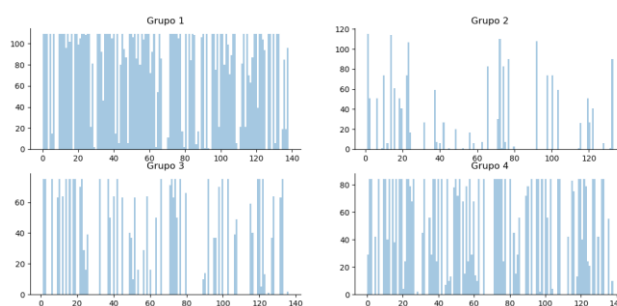


Figura 4. Histogramas de frecuencia de consideración de requerimientos en las soluciones de cada grupo.

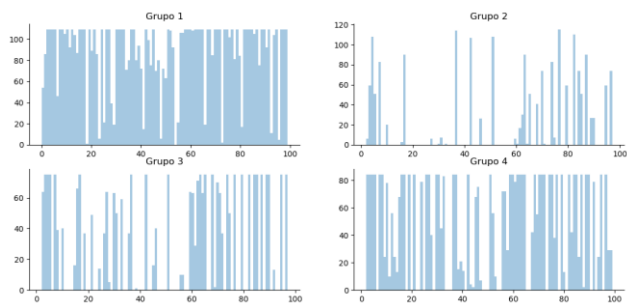


Figura 5. Histogramas de frecuencia de la satisfacción de stakeholders en las soluciones de cada grupo.

El grupo etiquetado con el número 1 se caracteriza por poseer valores promedio de costo y de beneficio elevados con una dispersión de aproximadamente el 10% de los mismos. El grupo, además, posee una gran cantidad de requerimientos contemplados para su implementación, como resulta razonable, y se encuentra a su vez asociada con la satisfacción de elevado número de stakeholders.

El grupo 2, por su parte, se encuentra en el extremo opuesto del frente de Pareto, al poseer los valores promedio de sus atributos más bajos. Sin embargo, la dispersión de estos valores resulta más alta (aproximadamente 50% del valor promedio del beneficio y 60% del valor promedio del costo), por lo que su análisis puede resultar más costoso para los tomadores de decisiones. No obstante, la cantidad de requerimientos considerados en este grupo para su implementación es menor, lo que puede ser una característica útil si se requiere de un proyecto de rápida ejecución, a expensas de satisfacer a un número menor de stakeholders.

Adicionalmente, el grupo 3 y 4 poseen valores de costo y beneficio en rangos intermedios a los grupos previamente mencionados. El grupo 3, que posee valores de costo y beneficio promedio más bajos contempla, como es esperado en una primera instancia, un número menor de requerimientos que el grupo 4. Ambos grupos poseen valores relativos de desviación estándar similares.

Habiendo supuesto que el grupo 2 resulta de interés para el negocio, y habiendo aplicado el método propuesto nuevamente, tal como se detalla en la sección anterior, se obtuvo el agrupamiento jerárquico detallado en la figura 6.

Posteriormente, se seleccionaron 4 grupos de este resultado, al igual que en en la etapa previa, pero cualquier número de grupos pudo haber sido elegido, dependiendo de los intereses del tomador de decisiones. Estos grupos se describieron de la misma forma que los grupos anteriores, obteniéndose los valores detallados en la tabla 3 y en las figuras 7 y 8.

Una vez más la segmentación es lineal (debido a las características del frente), siendo el subgrupo 1 el de las soluciones con mayor beneficio y mayor costo, con una desviación estándar aproximada del 3% y 5% del valor promedio respectivamente, y teniendo además la menor cantidad de elementos (16,1%). El subgrupo 4, por otro lado, presenta los valores más bajos de beneficio y costo,

con una desviación estándar considerablemente mayor (aproximadamente 50% en ambos casos). Este subgrupo, además, contiene la mayor cantidad de elementos (35,5%). En el rango intermedio se encuentran el subgrupo 2, con un 28,8% de los elementos, y el subgrupo 3, con un 19,4% de las soluciones de la porción del frente de Pareto analizada.

Con respecto a la configuración de la solución, los subgrupos poseen un orden decreciente en la cantidad de requerimientos contemplados, siendo el subgrupo 1 el que más requerimientos incluye, y el subgrupo 4 el que menos incluye. El mismo comportamiento se evidencia en el caso de los stakeholders.

Sin embargo, el grupo 2 y 3 incluye soluciones que no consideran a los requerimientos comprendidos entre los requerimiento 25 y el 66, por lo que se requiere especial atención en este aspecto para elegir la solución. No obstante, en este caso resulta más cómodo el análisis de las opciones por la escasa cantidad de elementos que posee cada cluster.

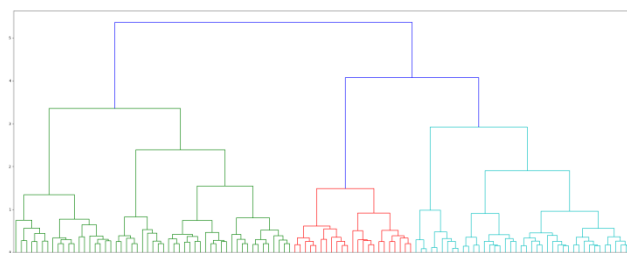


Figura 6. Dendograma obtenido de agrupar jerárquicamente los elementos del grupo 2 obtenido previamente.

Tabla 3. Descripción de los atributos costo (c) y beneficio (p) por cada subgrupo

Subgrupo	Cantidad	Attr.	Descriptor	Valor
1	19	p	Prom	813.7
			s.d.	32.9
		c	Prom	157.5
			s.d.	8.4
2	34	p	Prom	644.8
			s.d.	71.4
		c	Prom	116.6
			s.d.	16.1
3	23	p	Prom	423.4
			s.d.	47.7
		c	Prom	70.7
			s.d.	9.7
4	42	p	Prom	210.4
			s.d.	99.1
		c	Prom	30.4
			s.d.	16.1

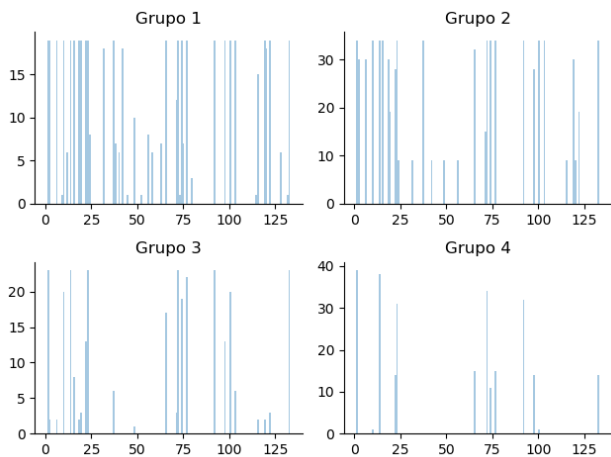


Figura 7. Histogramas de frecuencia de consideración de requerimientos en las soluciones de cada subgrupo.

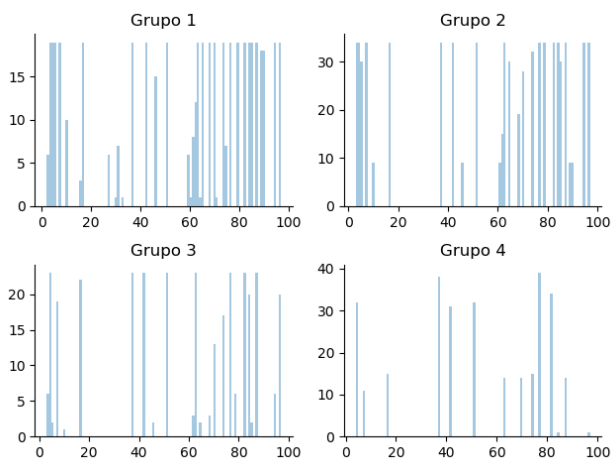


Figura 8. Histogramas de frecuencia de la satisfacción de stakeholders en las soluciones de cada subgrupo.

Discusión

Mediante el agrupamiento de las soluciones según distintos criterios de interés para los tomadores de decisiones es posible concentrar los esfuerzos de la exploración de las mismas en pos de la selección de aquella más adecuada a la situación particular de la organización de desarrollo de software.

Estos criterios son factibles de ser integrados en una única función de distancia que permita comparar las soluciones, quedando abierta la posibilidad de diseñar la misma según las necesidades de los tomadores de decisiones.

Por otro lado, el agrupamiento jerárquico permite generar una única vez el árbol de similitud y decidir posteriormente, en función de la configuración del resultado, cuántos grupos serán analizados, o bien, realizar de forma interactiva la exploración de los distintos niveles de granularidad, obteniendo diferentes resultados de forma dinámica.

La utilización de descriptores para estos grupos, ya sea de naturaleza estadística como en el caso presentado en este artículo, o basados en aprendizaje automático [11] si las características del frente de Pareto lo ameritan, hace de la exploración de los grupos una actividad no demandante.

Adicionalmente, la frecuencia con la que se consideran los requerimientos o los stakeholders resulta útil cuando el criterio de selección de soluciones contempla la cantidad de requerimientos a implementar o los stakeholders a satisfacer. De esta forma, un project manager puede preferir las soluciones que brinden respuesta a las necesidades de cierto grupo (o cantidad) de clientes, o que incluya ciertas funcionalidades.

Este método puede aplicarse de manera iterativa al realizar una vez más una segmentación de soluciones usando como entrada un grupo de interés seleccionado previamente. De esta forma, se puede reducir los candidatos hasta que quede una cantidad de soluciones adecuada para su análisis detallado.

Conclusiones

En el presente trabajo se introduce un método de exploración *a posteriori* de frente Pareto-Óptimo mediante agrupamiento de las soluciones basado en no solo el valor de los objetivos del problema de optimización considerado sino también la similitud dada por la configuración de cada una de las soluciones.

Este método permite al tomador de decisiones que no posee preferencias previas sobre la opción a seleccionar, explorar el frente de Pareto de acuerdo a las características que poseen las soluciones.

Se ha probado la propuesta sobre el frente de Pareto de una instancia del problema del próximo lanzamiento (NRP), pudiendo filtrar las soluciones del frente hasta niveles de granularidad manejables para su análisis por parte de un tomador de decisiones.

En futuros trabajos se buscará incorporar en la medida de similitud diseñada para tal fin cuestiones como la dependencia entre los componentes de las soluciones, que resultan de suma relevancia para los profesionales de la industria del software. Además, se realizarán pruebas exhaustivas sobre la descripción de los grupos generados mediante el uso de algoritmos de aprendizaje automático.

Adicionalmente se prevé el desarrollo de un software interactivo para la exploración de frente Pareto-Óptimo basado en la técnica propuesta, y su validación en escenarios reales mediante juicio de expertos.

Agradecimientos

El trabajo de investigación presentado en este artículo ha sido financiado parcialmente por el proyecto SIUTICU0005297TC “Enfoques de Optimización Multiobjetivo basados en Preferencias para problemas de

la Ingeniería de Software”. Se extienden nuestros agradecimientos al Ing. Luciano Bracco por el desarrollo de las herramientas para la obtención de los datos utilizados en estos experimentos.

[15]Rottoli, Giovanni Daián; Casanova, Carlos. “Clustering-Based NRP Pareto Front Exploration”. Código fuente. 2019. v.p.conaiisi.2019. Doi: 10.5281/zenodo.3404161. Disponible en:<https://github.com/GdRottoli/ClusteringBasedParetoFrontExploration.git>

Referencias

- [1] Harman, Mark, and Bryan F. Jones. "Search-based software engineering." *Information and software Technology* 43.14 (2001): 833-839.
- [2] Deb, Kalyanmoy, et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II." *IEEE transactions on evolutionary computation* 6.2 (2002): 182-197.
- [3] Zitzler, Eckart, Marco Laumanns, and Lothar Thiele. "SPEA2: Improving the strength Pareto evolutionary algorithm." *TIK-report* 103. 2001.
- [4] Ferreira, Thiago Nascimento, Silvia Regina Vergilio, and Jerffeson Teixeira de Souza. "Incorporating user preferences in search-based software engineering: A systematic mapping study." *Information and Software Technology* 90 (2017): pp. 55-69.
- [5] Saaty, Thomas L. "A scaling method for priorities in hierarchical structures." *Journal of mathematical psychology* 15.3 (1977): 234-281.
- [6] Bagnall, Anthony J., Victor J. Rayward-Smith, and Ian M. Whitley. "The next release problem." *Information and software technology* 43.14 (2001): 883-890.
- [7] Zhang, Yuanyuan, Mark Harman, and S. Afshin Mansouri. "The multi-objective next release problem." *Proceedings of the 9th annual conference on Genetic and evolutionary computation*. ACM, 2007.
- [8] N. Veerapen, G. Ochoa, M. Harman, and E. K. Burke, “An Integer Linear Programming approach to the single and bi-objective Next Release Problem,” *Information and Software Technology*, 2015.
- [9] Freire, Willian Marques, et al. "Supporting decision makers in search-based product line architecture design using clustering." *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*. Vol. 1. IEEE, 2019.
- [10] Han, Jiawei, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.
- [11] García-Martínez, Ramón, Paola Britos, and Dario Rodríguez. "Information mining processes based on intelligent systems." *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, Berlin, Heidelberg, 2013.
- [12] Casanova, C., et al. “Fuzzy Bi-Objective Particle Swarm Optimization for Next Release Problem”. *International Conference on Software Engineering and Knowledge Engineering*. (2019): pp. 509-512. Doi: 10.18293/SEKE2019-082
- [13] Miettinen, Kaisa (1998). *A Posteriori Methods*. En *Nonlinear Multiobjective Optimization*. Springer US. doi: 10.1007/978-1-4615-5563-6_4
- [14]Rottoli, Giovanni Daián; Casanova, Carlos, “Frente Pareto-Óptimo de instancia NRP”, Mendeley Data, v1. 2019. <http://dx.doi.org/10.17632/dwg6k63n9g.1>