Synergía ISSN L 2953-2973 Vol.4, No.2 Nov 2025- Abril 2026
https://revistas.up.ac.pa/index.php/synergia

Synergía pp.264-276

Aplicación del algoritmo NSGA-II en la resolución multiobjetivo del problema de la suma de subconjuntos

Application of the NSGA-II algorithm in the multi-objective resolution of the sum of subsets problem

Noriel Cosme Toribio

Universidad de Panamá. Facultad de Ciencias Naturales, Exactas y Tecnología, Panamá
noriel.cosmet@up.ac.pa https://orcid.org/0009-0004-7252-461X

Abraham De Sedas

Universidad de Panamá. Facultad de Ciencias Naturales, Exactas y Tecnología, Panamá
abraham.desedas@up.ac.pa https://orcid.org/0009-0006-7170-8649

Daniel Sánchez Díaz

Universidad de Panamá. Facultad de Ciencias Naturales, Exactas y Tecnología, Panamá
daniel-a.sanchez@up.ac.pa https://orcid.org/0009-0008-4326-5734

***Autor de correspondencia: (abraham.desedas@up.ac.pa)**

Fecha de recepción: 9/07/2025 Fecha de aceptación: 29/09/2025

DOI https://doi.org/10.48204/synergia.v4n2.8548

Resumen

El estudio presenta el algoritmo NSGA-II como una herramienta eficiente y óptima para resolver el problema de la suma de subconjuntos (SSP), que es un problema binario de mochila con diversas aplicaciones en áreas como la gestión de inversiones, la planificación de la producción y el diseño de circuitos electrónicos. El NSGA-II es un algoritmo genético multiobjetivo que utiliza técnicas de selección, cruce y mutación, junto con un enfoque de clasificación no dominado, para evolucionar una población de soluciones y obtener un conjunto de soluciones no dominadas, conocido como el frente de Pareto. El estudio describe detalladamente el funcionamiento del algoritmo, incluyendo los operadores genéticos y el enfoque de clasificación no dominado. Además, se presentan resultados experimentales que demuestran la eficacia y eficiencia del algoritmo en la resolución del problema SSP. En general, se proporciona una base sólida para comprender los fundamentos y aplicaciones del algoritmo NSGA-II en la optimización multiobjetivo.

Palabras clave: subconjuntos, cruce, mutación, algoritmo, genética.

Abstract

The study presents the NSGA-II algorithm as an efficient and optimal tool for solving the Subset Sum Problem (SSP), which is a binary knapsack problem with diverse applications in areas such as investment management, production planning, and electronic circuit design. NSGA-II is a multi-objective genetic algorithm that uses selection, crossover, and mutation techniques, along with a non-dominated sorting approach, to evolve a population of solutions and obtain a set of non-dominated solutions known as the Pareto front. The study provides a detailed description of the algorithm's functioning, including the genetic operators and the non-dominated sorting approach. Furthermore, experimental results are presented to demonstrate the effectiveness and efficiency of the algorithm in solving the SSP. Overall, the study provides a solid foundation for understanding the fundamentals and applications of the NSGA-II algorithm in multi-objective optimization.

Keywords: subsets, crossover, mutation, algorithm, genetic.

Introducción

El problema de la suma de subconjuntos (SSP) es un problema de binario de mochila. Este problema tiene varias aplicaciones. Algunas de ellas son:

1. Optimización de carteras: En la gestión de inversiones, el problema de la suma de subconjuntos se utiliza para seleccionar una combinación óptima de activos para maximizar el rendimiento o minimizar el riesgo, teniendo en cuenta restricciones como límites de inversión y diversificación.
2. Planificación de la producción: En la planificación de la producción, el problema de la suma de subconjuntos se utiliza para seleccionar un conjunto óptimo de productos o elementos para cumplir con la demanda de los clientes y maximizar la eficiencia de los recursos disponibles.

3. Diseño de circuitos electrónicos: en el diseño de circuitos electrónicos, el problema de la suma de subconjuntos se utiliza para seleccionar un conjunto óptimo de componentes electrónicos para cumplir con ciertas especificaciones, como la potencia consumida o la velocidad de funcionamiento, mientras se minimiza el costo.

Estas son algunas de las aplicaciones más comunes del problema de la suma de subconjuntos.

Donde el problema de la suma de subconjuntos (SSP) se plantea de la siguiente manera: Sea un conjunto W de n enteros y C un entero grande dado. El objetivo es encontrar un subconjunto S (de W) cuya suma de elementos es próximo a C, sin excederse.

Función objetivo

$$P(\vec{x}) = \sum_{i=1}^{n} w_i x_i$$

donde $\vec{x} = (x_1, x_2, x_3, \dots, x_n)$ es una solución factible.

Sujeto a

$$\sum_{i=1}^{n} w_i x_i \leq C$$

donde $x_i \in \{0, 1\}$.

La solución óptima es un vector que maximiza la función objetivo, es decir, un vector factible que corresponde al conjunto con la máxima suma.

Garey y Johnson (1979) demostraron que un problema específico pertenece a la clase NP-completo, lo cual plantea desafíos significativos debido a la complejidad computacional involucrada. Dado que las técnicas de optimización clásicas no son eficaces para resolver problemas NP-completos, se han desarrollado nuevos métodos. Entre estos métodos se encuentran los algoritmos genéticos, que ofrecen una alternativa basada en la evolución biológica y la genética para abordar problemas complejos. No existe una definición de algoritmo genético única, algunas de ellas son:

1) Genetic Algorithms in Search, Optimization, and Machine Learning (Algoritmos Genéticos en Búsqueda, Optimización y Aprendizaje Automático) por David E. Goldberg (1989): Según Goldberg, los algoritmos genéticos son métodos de búsqueda basados en la mecánica de la evolución y la genética. Estos algoritmos emplean una población de posibles soluciones y utilizan operadores genéticos, como reproducción, mutación y recombinación, para generar nuevas soluciones a partir de las existentes. Mediante la aplicación de un proceso de selección, los algoritmos genéticos guían la evolución de la población hacia soluciones de mejor calidad a lo largo del tiempo.

2) Genetic Algorithms: Principles and Perspectives - A Guide to GA Theory (Algoritmos Genéticos: Principios y Perspectivas - Una Guía sobre la Teoría de los AG) por Colin R. Reeves (1993): Reeves define los algoritmos genéticos como una clase de algoritmos de búsqueda estocástica que operan en una población de individuos codificados genéticamente. Estos individuos representan posibles soluciones a un problema dado. El proceso de evolución en los algoritmos genéticos incluye la selección de individuos prometedores, la aplicación de operadores genéticos y la evaluación de la calidad de las soluciones. A través de múltiples generaciones, se espera que los algoritmos genéticos converjan hacia soluciones óptimas o aceptables.

3) Handbook of Genetic Algorithms (Manual de Algoritmos Genéticos) por Lawrence Davis (1991): Davis describe los algoritmos genéticos como un enfoque de búsqueda y optimización basado en los principios de la evolución y la genética. Estos algoritmos mantienen una población de soluciones candidatas y utilizan operadores genéticos, como la reproducción sexual, la mutación y la recombinación, para generar nuevas soluciones. La selección, que se basa en el rendimiento de los individuos en una función objetivo, determina qué soluciones sobreviven y se reproducen en cada generación. A medida que se repite este ciclo

evolutivo, los algoritmos genéticos buscan mejorar la calidad de las soluciones a lo largo del tiempo.

De todas ellas podemos formular una definición intuitiva de los algoritmos genéticos que son técnicas de optimización y búsqueda inspirada en la evolución biológica. Se utilizan para resolver problemas complejos y encontrar soluciones aproximadas o subóptimas en una amplia variedad de dominios.

El algoritmo NSGA-II (Non-dominated Sorting Genetic Algorithm II) es uno de los métodos propuestos para resolver este problema. El NSGA-II es un algoritmo genético multiobjetivo ampliamente utilizado para resolver problemas de optimización con múltiples objetivos. Fue propuesto por Deb et al. (2002) en el artículo "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II" publicado en el año 2002 en la revista IEEE Transactions on Evolutionary Computation.

En el artículo, los autores presentan NSGA-II como una mejora del NSGA original, diseñada para abordar problemas de optimización multiobjetivo de manera más eficiente y efectiva. El algoritmo NSGA-II utiliza una combinación de técnicas genéticas, como selección, cruce y mutación, junto con un enfoque de clasificación no dominado para evolucionar una población de soluciones y obtener un conjunto de soluciones no dominadas, conocido como el frente de Pareto.

El artículo describe detalladamente el funcionamiento del algoritmo NSGA-II, incluyendo la selección de padres, el operador de cruza, el operador de mutación y el enfoque de clasificación no dominado. También se presentan resultados experimentales que demuestran la eficacia y eficiencia del algoritmo en la resolución de problemas de optimización multiobjetivo.

Deb et al. (2002) introdujeron NSGA-II como una herramienta valiosa en el campo de la optimización multiobjetivo y ha sido ampliamente citado y utilizado en la comunidad científica. El artículo proporciona una base sólida para comprender los fundamentos y aplicaciones del algoritmo NSGA-II en la resolución de problemas de optimización con múltiples objetivos.

El seudocódigo básico del algoritmo NSGA-II es

1. Inicializar la población $P(t)$ de tamaño n.
2. Evaluar el fitness de los individuos en $P(t)$.
3. Calcular el rango no dominado y la distancia de crowding para todos los individuos en $P(t)$.
4. Crear una nueva población $Q(t)$ vacía.
5. Repetir hasta que $Q(t)$ esté llena:
    a. Seleccionar dos padres mediante torneo binario desde $P(t)$.
    b. Aplicar operadores de cruzar y mutación para generar dos hijos.
    c. Evaluar el fitness de los hijos.
    d. Añadir los hijos a $Q(t)$.
6. Combinar $P(t)$ y $Q(t)$ en una población $P(t+1)$ de tamaño $2n$.
7. Calcular el rango no dominado y la distancia de crowding para todos los individuos en $P(t+1)$.
8. Seleccionar los mejores N individuos de acuerdo con el rango no dominado y la distancia de crowding.
9. Incrementar t en 1.
10. Si se ha alcanzado el número máximo de generaciones, detenerse; de lo contrario, volver al paso 2.

El objetivo de este artículo es aplicar el algoritmo NSGA-II para resolver el problema SSP y, además, ilustrar el uso de la Matemática para resolver un problema multiobjetivo.

Materiales y métodos

Población y muestra

Este estudio fue de carácter exploratorio y experimental. La población estuvo conformada por instancias del problema de la suma de subconjuntos (SSP) que fueron generadas computacionalmente, simulando diferentes combinaciones de subconjuntos con respecto a un valor objetivo predeterminado. La muestra correspondió a un conjunto de ejecuciones del algoritmo NSGA-II bajo diferentes configuraciones de parámetros, como tamaño de población, número de generaciones y probabilidades de cruce y mutación.

Variables

Las variables consideradas en este estudio fueron:

- El número de elementos en el conjunto original,
- El valor objetivo (target) a alcanzar,
- Las soluciones generadas (subconjuntos seleccionados),
- La suma total de cada subconjunto, y
- La distancia entre dicha suma y el valor objetivo.

Procedimiento

Primero se definieron múltiples instancias del problema SSP, especificando el conjunto de entrada y el valor objetivo. Posteriormente, se implementó el algoritmo NSGA-II en un entorno de programación controlado, asegurando la replicabilidad de los resultados. Las ejecuciones siguieron la estructura clásica del algoritmo: inicialización de una población aleatoria, evaluación de la función objetivo, aplicación de operadores genéticos (cruza, mutación), y ordenamiento no dominado para construir el frente de Pareto.

Durante las simulaciones, se registraron las soluciones no dominadas generadas en cada iteración. Se evaluó su desempeño en términos de dos objetivos: maximizar la suma de elementos sin exceder el valor objetivo, y minimizar la cantidad de elementos utilizados.

Para el análisis de resultados, se graficó el frente de Pareto y se examinó la evolución de las soluciones a través de las generaciones.

Este enfoque permitió estudiar el comportamiento del algoritmo frente a distintas complejidades del problema SSP, así como validar su eficacia como técnica de optimización multiobjetivo.

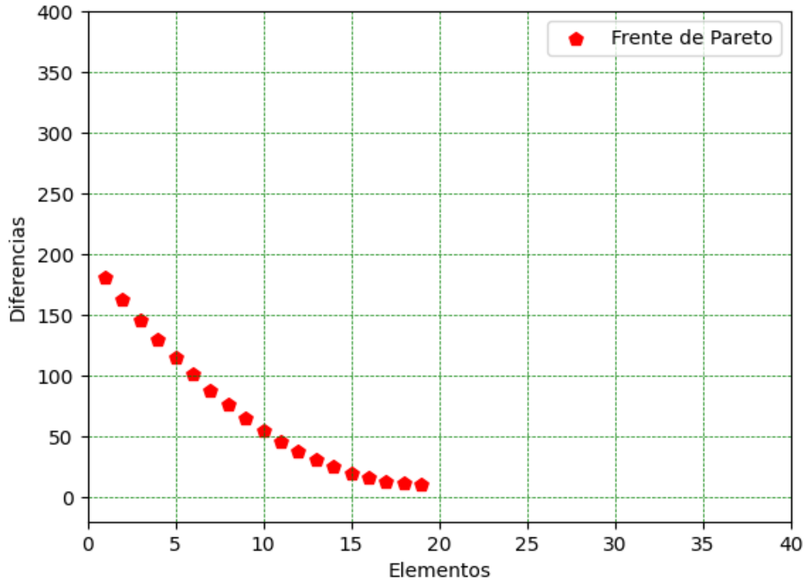
Resultados y discusión

La Figura 1 presenta el frente de Pareto obtenido mediante la implementación del algoritmo NSGA-II, siguiendo el enfoque propuesto por Van Veldhuizen y Lamont (1998). En dicha figura, se observa cómo las soluciones no dominadas se distribuyen a lo largo de un gradiente que refleja el equilibrio entre dos objetivos en conflicto: maximizar la suma total de los elementos seleccionados y minimizar la diferencia con respecto al valor objetivo, sin excederlo. A medida que el número de elementos en el subconjunto seleccionado incrementa, se tiende a una mayor aproximación al valor objetivo, lo cual indica una mejora en la calidad de la solución. Sin embargo, esta relación no es estrictamente lineal, ya que la inclusión de más elementos puede llevar también a soluciones subóptimas si no se consideran adecuadamente las combinaciones. De forma inversa, cuando se seleccionan subconjuntos con menor cantidad de elementos, la suma total tiende a alejarse más del valor objetivo, lo que evidencia la tensión inherente en este problema multiobjetivo: buscar una solución compacta (con pocos elementos) que al mismo tiempo sea lo más cercana posible al objetivo deseado. Esta dinámica es precisamente la que captura el frente de Pareto, mostrando la frontera de eficiencia en la cual ninguna solución es mejor en ambos objetivos simultáneamente, sino que cada punto representa un compromiso óptimo entre ellos.

Este resultado evidencia la capacidad del algoritmo NSGA-II para generar soluciones diversificadas y eficientes que permiten al tomador de decisiones elegir según criterios contextuales, como el costo computacional o la penalización por sobrecarga de elementos en una aplicación práctica. En consecuencia, el frente de Pareto no solo visualiza el rendimiento del algoritmo, sino que también ofrece una herramienta para analizar la estructura del problema y la distribución de soluciones viables en el espacio de búsqueda.

Figura 1.

Frente de Pareto



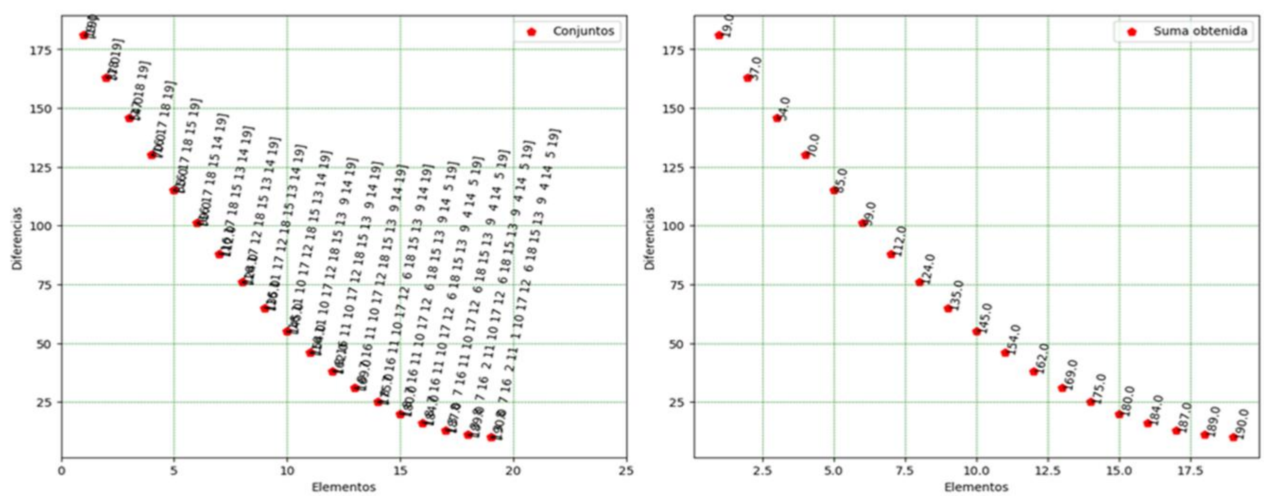
La Figura 2 proporciona información complementaria sobre las soluciones generadas por el algoritmo. En la gráfica ubicada a la derecha, se representa la suma total de los elementos de cada subconjunto perteneciente al frente de Pareto. Se aprecia una tendencia creciente: a medida que se incorporan más elementos en los subconjuntos, la suma correspondiente se aproxima progresivamente al valor objetivo. Este comportamiento es consistente con la naturaleza del problema, ya que agregar elementos ofrece mayor capacidad para acercarse al umbral deseado, aunque también incrementa el riesgo de sobrepasarlo. En el extremo inferior de esta gráfica, se identifican subconjuntos cuya suma se ajusta con precisión al valor objetivo, lo que representa soluciones óptimas dentro de las restricciones impuestas.

Por otro lado, la figura situada a la izquierda visualiza los elementos individuales que conforman cada subconjunto presente en el frente de Pareto. Esta representación permite analizar la composición estructural de las soluciones, revelando patrones de inclusión recurrentes entre subconjuntos eficaces. La variabilidad entre subconjuntos muestra cómo diferentes combinaciones pueden alcanzar niveles similares de desempeño, lo que refuerza la utilidad del NSGA-II al ofrecer múltiples soluciones eficientes entre las cuales se puede seleccionar de acuerdo con criterios adicionales (como la cardinalidad, el peso computacional, o preferencias del usuario).

En conjunto, ambas gráficas permiten no solo validar el correcto funcionamiento del algoritmo, sino también comprender la dinámica interna del problema multiobjetivo, proporcionando una perspectiva más rica y detallada sobre las soluciones no dominadas generadas por el NSGA-II.

Figura 2.

Suma de las soluciones y elementos de cada subconjunto



Se puede destacar la eficacia del algoritmo en encontrar soluciones no dominadas, representadas por el frente de Pareto, que equilibran diferentes objetivos, como maximizar la suma de elementos y cumplir con restricciones de capacidad. El NSGA-II supera las limitaciones de las técnicas de optimización clásicas al abordar problemas NP-completos, como el SSP. Los resultados experimentales muestran cómo el frente de Pareto obtenido se acerca cada vez más a la suma objetivo a medida que aumenta el número de elementos en el subconjunto. Además, se observa cómo la suma incrementa a medida que se añaden más elementos, alcanzando el valor deseado en el extremo inferior del frente de Pareto. Esto demuestra que el algoritmo NSGA-II es capaz de encontrar soluciones óptimas y eficientes para el SSP. Este método es importante por su aplicabilidad en otros problemas de optimización multiobjetivo.

Conclusiones

El algoritmo NSGA-II ha demostrado ser altamente efectivo y eficiente en la resolución del problema de la suma de subconjuntos (Subset Sum Problem, SSP), una variante del problema de la mochila y clasificado como NP-completo. Gracias a su enfoque evolutivo y su capacidad para generar un conjunto diverso de soluciones no dominadas, el NSGA-II permite abordar problemas multiobjetivo con múltiples restricciones de manera simultánea y balanceada. En el contexto del SSP, el algoritmo logra construir frentes de Pareto que representan soluciones óptimas en términos del compromiso entre maximizar la suma de subconjuntos y minimizar la desviación respecto al valor objetivo, sin sobrepasarlo.

La aplicación de NSGA-II no solo facilita la obtención de soluciones robustas, sino que también proporciona herramientas de análisis útiles para la toma de decisiones, ya que permite visualizar la diversidad de soluciones y comprender su estructura interna. Además, su flexibilidad y escalabilidad lo convierten en un enfoque adecuado para problemas de gran complejidad y con múltiples objetivos en conflicto, como los que se encuentran en campos como la gestión de carteras financieras (Marinakis & Marinaki, 2010), la planificación de la producción (Deb et al., 2002), y el diseño de sistemas electrónicos (Coello Coello, 2006).

Este trabajo contribuye a la literatura proporcionando un estudio aplicado y reproducible del NSGA-II en un problema clásico de optimización combinatoria. Asimismo, sienta las bases para futuras investigaciones que busquen adaptar o hibridar este algoritmo con otras técnicas metaheurísticas, como los algoritmos de colonia de hormigas o de enjambre de partículas, con el fin de mejorar aún más la eficiencia de búsqueda en espacios de soluciones de alta dimensionalidad.

El NSGA-II se posiciona como una herramienta poderosa dentro del repertorio de algoritmos evolutivos para la optimización multiobjetivo, con una sólida fundamentación teórica y un amplio respaldo empírico en aplicaciones reales y simuladas.

Agradecimiento

Expresamos nuestro sincero agradecimiento al Magíster Julio Trujillo, del Departamento de Matemática de la Universidad de Panamá, por sus valiosas recomendaciones, orientación metodológica y constante guía a lo largo del desarrollo de esta investigación. Su experiencia, compromiso y apoyo fueron fundamentales para la culminación exitosa de este estudio.

Referencias bibliográficas

Coello Coello, C. A. (2006). Evolutionary multi-objective optimization: a historical view of the field. IEEE Computational Intelligence Magazine, 1(1), 28–36. https://doi.org/10.1109/MCI.2006.1597056

Davis, L. (1991). Handbook of Genetic Algorithms. Van Nostrand Reinhold.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2), 182-197.

Garey, M.R. and Johnson, D.S. (1979). Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman & Co., New York.

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Professional.

Marinakis, Y., & Marinaki, M. (2010). A hybrid genetic–particle swarm optimization algorithm for the vehicle routing problem. Expert Systems with Applications, 37(2), 1446–1455. https://doi.org/10.1016/j.eswa.2009.06.106

Reeves, C. R. (1993). Genetic Algorithms: Principles and Perspectives - A Guide to GA Theory. Springer.

Van Veldhuizen, D. A., & Lamont, G. B. (1998). Evolutionary computation and convergence to a pareto front. In Late breaking papers at the genetic programming 1998 conference (pp. 221-228).