

Maximum Diversity Problem (MDP): Implementación de GRASP y Path-relinking.

Carles Vicent Adam Castañer, Luna Xinping Moreno Gómez, y Enrique Sayas Bailach¹

Abstract Este trabajo aborda el Maximum Diversity Problem (MDP), un problema de optimización combinatoria cuya complejidad NP-hard motiva el uso de técnicas metaheurísticas. Se implementa el algoritmo GRASP (Greedy Randomized Adaptive Search Procedure) y se analiza su rendimiento con y sin la incorporación de la técnica de Path-relinking. A través de la calibración del parámetro α y la selección de soluciones élite basadas en calidad y diversidad, se explora el impacto de esta combinación sobre 15 instancias del MDP. Los resultados muestran que GRASP ofrece un mejor rendimiento en escenarios de tiempo limitado, aunque se sugiere que una versión más optimizada de Path-relinking podría mejorar los resultados en instancias más grandes.

¹ Carles Vicent Adam Castañer
Estudiante de Ciencia de Datos en la Universidad de Valencia, e-mail:
caracas2@alumni.uv.es

Luna Xinping Moreno Gómez
Estudiante de Ciencia de Datos en la Universidad de Valencia, e-mail:
luxinmo@alumni.uv.es

Enrique Sayas Bailach
Estudiante de Ciencia de Datos en la Universidad de Valencia, e-mail:
ensabai@alumni.uv.es

1 Introducción

El *Maximum Diversity Problem* (MDP) es un problema de optimización combinatoria que consiste en seleccionar un subconjunto de elementos desde un conjunto más amplio, de manera que se maximice la diversidad entre los elementos seleccionados. Este problema es relevante en múltiples contextos prácticos, como la conformación de equipos diversos, la ubicación de instalaciones, el diseño de portafolios y la planificación de productos, entre otros, donde es deseable que los elementos elegidos sean lo más distintos posible entre sí.

Debido a su complejidad, el *MDP* es un problema *NP-hard*, lo que significa que no es viable resolverlo de forma exacta para instancias grandes en un tiempo razonable. Por esta razón, se recurre a métodos heurísticos que permiten encontrar buenas soluciones en un tiempo aceptable. Uno de estos métodos es GRASP (*Greedy Randomized Adaptive Search Procedure*), una metaheurística basada en la generación iterativa de soluciones utilizando una combinación de estrategias aleatorias y búsqueda local.

Sin embargo, los resultados de GRASP pueden mejorarse utilizando técnicas adicionales. Una de ellas es *Path-relinking*, un enfoque que busca intensificar la búsqueda conectando soluciones élite ya conocidas mediante trayectorias en el espacio de soluciones. Esta técnica permite explorar nuevas soluciones potencialmente mejores al aprovechar la información contenida en soluciones previamente generadas.

Este trabajo tiene como propósito explorar el impacto de aplicar *Path-relinking* como complemento al algoritmo GRASP para resolver el Maximum Diversity Problem. Para ello, se evaluarán distintas configuraciones del parámetro alfa de GRASP, se implementará el mecanismo de *Path-relinking* para generar caminos entre soluciones élite, y se realizará una comparación empírica de los resultados obtenidos en ambos casos sobre un conjunto de 15 instancias del problema.

2 GRASP (Greedy Randomized Adaptive Search Procedure)

El algoritmo GRASP es una metaheurística diseñada para abordar problemas de optimización combinatoria, especialmente aquellos donde los métodos exactos resultan computacionalmente inviables debido a la complejidad o al tamaño del espacio de búsqueda. Este algoritmo combina estrategias codiciosas con componentes aleatorios y adaptativos, lo que le permite generar soluciones de buena calidad en un tiempo razonable.

El funcionamiento de GRASP se basa en un proceso iterativo que consta de dos fases principales: la construcción de una solución inicial y la mejora de esta mediante búsqueda local. En la fase constructiva, el algoritmo genera una solución factible paso a paso utilizando un enfoque codicioso aleatorizado. En cada paso se construye una Lista de Candidatos Restringida (RCL), que contiene un subconjunto de los mejores elementos posibles según una función heurística. A partir de esta lista, se selecciona aleatoriamente un elemento, lo que

introduce diversidad en las soluciones generadas. Posteriormente, en la fase de búsqueda local, GRASP explora el vecindario de la solución actual en busca de mejoras. Este proceso de refinamiento continúa hasta alcanzar un óptimo local. Todo el procedimiento se repite múltiples veces, guardando siempre la mejor solución encontrada a lo largo de las iteraciones.

En nuestro trabajo, aplicaremos GRASP a ambos conjuntos pero les dejaremos un tiempo de ejecución diferentes, ya que en uno de ellos se aplicará PR posteriormente para poder realizar la comparativa.

2.1 Calibrar Alpha

Durante el proceso de calibración, se evaluaron múltiples valores del parámetro α dentro del algoritmo GRASP aplicado al Maximum Diversity Problem, con el fin de determinar su impacto en la calidad de las soluciones obtenidas. Los experimentos se realizaron sobre las 15 instancias proporcionadas, considerando distintos tiempos de ejecución para asegurar robustez en los resultados. Como se observa en la Figura 1, el valor $\alpha = 0.1$ se destacó como el más eficaz, presentando la menor desviación media porcentual en comparación con otras configuraciones evaluadas. Este comportamiento sugiere que una selección más restrictiva durante la fase de construcción (menor α) favorece soluciones de mayor diversidad.

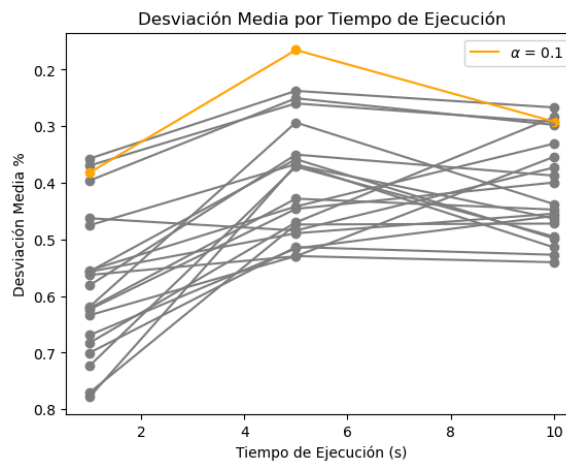


Figura 1. Determinar el mejor valor de α .

3 Path-relinking

Path-Relinking es una estrategia de intensificación ampliamente utilizada en metaheurísticas para aprovechar el conocimiento previamente adquirido sobre soluciones de alta calidad. Originalmente introducido en el contexto de Scatter Search, esta técnica se ha adaptado con

éxito a múltiples entornos heurísticos, como GRASP, para reforzar la explotación de regiones prometedoras del espacio de búsqueda.

El objetivo principal de este proceso consiste en generar trayectorias en el espacio de soluciones que conectan una solución de partida con otra considerada como guía. A lo largo de esta trayectoria, se van explorando soluciones intermedias que incorporan, de manera gradual, características de la solución guía, con el propósito de encontrar posibles configuraciones mejoradas que no habían sido encontradas en la fase inicial de construcción.

En nuestro trabajo, aplicamos Path-Relinking como una segunda fase del procedimiento GRASP, aprovechando un conjunto de soluciones élite extraídas de múltiples ejecuciones del constructivo aleatorizado y la búsqueda local durante diversos periodos de tiempo.

Cada una de estas soluciones representa un buen equilibrio entre calidad (presentan un valor objetivo alto) y diversidad (tienen estructura distinta), lo que favorece trayectorias PR con grandes oportunidades de mejora.

3.1 Selección conjunto de soluciones de élite

El conjunto de soluciones élite se forma durante la fase GRASP, que se ejecuta durante un tiempo limitado (en nuestro caso, se opta por la mitad del presupuesto total de tiempo). En lugar de seleccionar las mejores soluciones según su valor objetivo, implementamos una opción híbrida de selección basada en calidad y diversidad.

El proceso comienza seleccionando la mejor solución encontrada. A continuación, se añaden iterativamente aquellas soluciones que, además de tener buena calidad, sean lo más diferentes posible del conjunto élite ya formado. Esta disimilitud se mide utilizando la **diferencia simétrica** entre los conjuntos de elementos seleccionados por cada solución.

Cada candidato se evalúa con una puntuación que combina su valor objetivo con su mínima distancia al conjunto élite actual, ponderada por un parámetro λ :

$$\text{score}(s) = \text{of}(s) + \lambda \cdot \min_{e \in E} \{\text{distancia}(s, e)\}$$

Donde $\text{of}(s)$ es el valor objetivo de la solución candidata y $\text{distancia}(s, e)$ corresponde al número de elementos diferentes respecto a cada solución e del conjunto élite E . En nuestros experimentos, utilizamos $\lambda=1$, lo que dará igual importancia a calidad y diversidad.

Este enfoque garantiza que el conjunto élite esté compuesto por soluciones de alta calidad según su valor objetivo, pero también estructuralmente diversas, lo cual consideramos es crucial para que Path-Relinking sea efectivo.

3.2 Mecanismo de Path-Relinking

El procedimiento de Path-Relinking se aplica entre todos los pares de soluciones del conjunto élite, en ambas direcciones: desde una solución de partida hacia una solución guía, y viceversa. La trayectoria se construye mediante un proceso iterativo de intercambios, en el que se eliminan elementos de la solución actual y se incorporan elementos de la guía que aún no han sido incluidos.

Por tanto, en cada paso de este recorrido:

- Se consideran posibles intercambios entre elementos candidatos a añadir y a eliminar.
- Se emplea una medida heurística basada en la suma de distancias desde cada nodo candidato al resto de la actual solución para evaluar el impacto de cada intercambio.
- Aplicando un enfoque lista restringida de candidatos (RCL) similar al de GRASP y parametrizado por un valor β , se opta por un intercambio que maximice la mejora esperada.

La función *path_relinking(...)* calcula eficientemente el cambio en la función objetivo (*delta_of*) sin reevaluar toda la solución, lo que permite mantener una ejecución rápida incluso en instancias grandes (en nuestro experimento hasta 500 elementos).

El algoritmo continúa, o bien hasta que se alcance un tiempo máximo previamente marcado (como ocurre en nuestro experimento), o hasta que la solución actual sea igual a la guía, es decir, todos los elementos diferentes hayan sido probados. Durante este recorrido, se almacena y retorna la mejor solución encontrada.

A continuación, mostramos un pseudocódigo simplificado del procedimiento de PR usado en nuestro código:

```
def path_relinking(inicio, guia, beta):
    actual = copy(inicio)
    mejor = actual
    while elementos_diferentes(actual, guia):
        candidatos = generar_intercambios(actual, guia)
        rcl = filtrar_por_beta(candidatos, beta)
        (a, r) = seleccionar_aleatorio(rcl)
        actual = aplicar_intercambio(actual, a, r)
        if evaluar(actual) > evaluar(mejor):
            mejor = actual
    return mejor
```

El esquema anterior resume el enfoque implementado, que ha sido probado en las 15 instancias que disponíamos. Se aplica, como ya se ha comentado, un límite de tiempo a la fase de Path-Relinking, igual al usado en la fase GRASP, que asegura una comparación justa entre ambos enfoques.

4 Resultados

Para comparar el rendimiento entre GRASP y GRASP+Path-relinking, se han ejecutado todas las instancias para un tiempo de ejecución de 60 segundos. En el caso de GRASP + Path-relinking, se ha destinado 30 segundos a la creación del conjunto de soluciones elite y los 30 segundos restantes para la búsqueda de soluciones intermedias.

En la gráfica se observa que para todos los problemas de tamaño 100 ambas implementaciones convergen a la misma solución. Sin embargo, el tiempo medio de ejecución para GRASP+PR es de 30 segundos, lo que indica que realmente sólo destina al Path-relinking 1 segundo.

Por otra parte, en los problemas de tamaño 500, se alcanza un peor comportamiento por parte de GRASP+PR. Esto se puede deber a que el tiempo destinado al Path-relinking es menor del necesario para encontrar mejores soluciones intermedias, o que el conjunto de elite definido no sea realmente el adecuado.

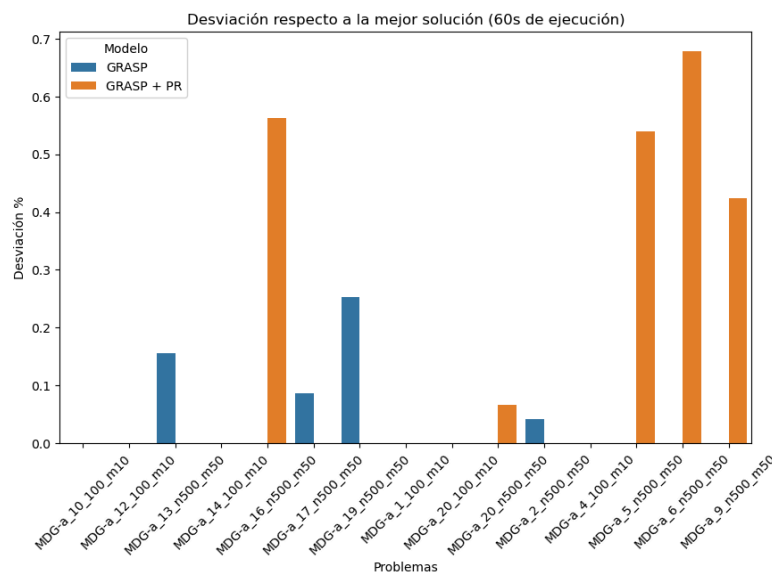


Figura 2. Desviación respecto a la mejor solución.

A continuación se muestra una tabla con los resultados obtenidos.

| | | GRASP | GRASP + PR |
|--------------|------------|----------|------------|
| n=100 | Desv. | 0 % | 0 % |
| | # Mejor | 6 | 6 |
| | Tiempo (s) | 60 | 31 |
| n=500 | Desv. | 0.0597 % | 0.25 % |
| | # Mejor | 5 | 4 |
| | Tiempo (s) | 60.05 | 60.93 |

Tabla 1. Resultados para GRASP y GRASP+PR.

5 Conclusiones

Tras comparar ambas implementaciones, se aprecia un mejor rendimiento del algoritmo GRASP. No obstante, con una versión más optimizada del Path-relinking se podría llegar a mejores resultados. En el Jupyter Notebook adjunto se muestran todos los pasos realizados para la selección del parámetro α , así como para la obtención de los datos de la tabla.

Bibliografía

Resende, M. G. C., Ribeiro, C. C., Glover, F., & Martí, R. (2010). Scatter search and path-relinking: Fundamentals, advances, and applications. En M. Gendreau & J.-Y. Potvin (Eds.), Handbook of Metaheuristics (pp. 87–107). Springer. https://doi.org/10.1007/978-1-4419-1665-5_4

Universidad de Sevilla. (s.f.). Capítulo 6 [Tesis de grado, Universidad de Sevilla]. <https://biblus.us.es/bibing/proyectos/abreproy/70317/fichero/Capitulo+6.pdf>