

**Diseño y Aplicación de una Metodología Heurística
en la Resolución de Problemas de Optimización Complejos**

María de los Angeles Harita Rascon

ADVERTIMENT. L'accés als continguts d'aquesta tesi queda condicionat a l'acceptació de les condicions d'ús establertes per la següent llicència Creative Commons:  <https://creativecommons.org/licenses/?lang=ca>

ADVERTENCIA. El acceso a los contenidos de esta tesis queda condicionado a la aceptación de las condiciones de uso establecidas por la siguiente licencia Creative Commons:  <https://creativecommons.org/licenses/?lang=es>

WARNING. The access to the contents of this doctoral thesis it is limited to the acceptance of the use conditions set by the following Creative Commons license:  <https://creativecommons.org/licenses/?lang=en>



Escuela de Ingeniería
Departamento de Arquitectura de Computadores
y Sistemas Operativos

Diseño y Aplicación de una Metodología Heurística en la Resolución de Problemas de Optimización Complejos

María de los Angeles Harita Rascon

Universidad Autònoma de Barcelona

Noviembre 2025



Arquitectura de Computadores y Sistemas Operativos

**Diseño y Aplicación de una Metodología
Heurística en la Resolución de Problemas de
Optimización Complejos**

Tesis presentada por **María de los Angeles Harita Rascón**, para optar al grado de Doctor por la *Universitat Autònoma de Barcelona*. Este trabajo ha sido desarrollado en el departamento de Arquitectura de Computadores y Sistemas Operativos (DACSO - CAOS) bajo la dirección de la Dra. Dolores Isabel Rexachs del Rosario y el Dr. Emilio Luque Fadón.

Directores

Autor

Acknowledgements

Este proyecto de Tesis ha sido realizado con la colaboración de la Agencia Estatal de Investigación (AEI), en España y el Fondo Europeo de Desarrollo Regional (FEDER), Unión Europea (UE) bajo el contrato con número de referencia PID2020-112496GB-I00 y título **Computación de Altas Prestaciones Eficiente y Segura para Aplicaciones de Servicios de Salud Inteligentes**. También bajo el contrato con número de referencia PID2023-147955NB-I00 y título **Computación de Altas Prestaciones y Algoritmos Heurísticos para la Gestión de Sistemas Sanitarios Inteligentes**. Además, ha sido parcialmente financiado por la Fundación Escuelas Universitarias Gimbernat (EUG).

Dedicatoria

A Dios, ante todo, por sostenerme, por darme fuerzas cuando no las encontraba y por abrir caminos donde parecía no haberlos. A quienes me han brindado su apoyo incondicional y, sobre todo, su voto de confianza, mi más sincero agradecimiento. Sin su fe en mí, este camino habría sido mucho más difícil de transitar.

A mi familia, el pilar fundamental de mi vida, gracias por su paciencia, su comprensión y su amor constante. A mis padres, siempre presentes y apoyándome incluso en la distancia. A mis hijos, quienes son mi mayor alegría y mi motor diario: la pequeña Edurne, que comenzó esta aventura siendo apenas un bebé y ha tenido que compartir estos años con una madre absorbida por un trabajo monumental, Y a los mayores, María y Álvaro por su sensibilidad, su madurez y por recordarme siempre lo que realmente importa. Sepan que, aunque fue un sacrificio, también fue un acto de lleno de esperanza.

A Álvaro, por su apoyo, y por estar incluso en los momentos más duros en medio del caos: la pandemia, los chicos, las mudanzas, el trabajo, la tesis, y hasta cuando me rompí el pie.

A mis tutores, Lola y Emilio, por su paciencia infinita, su guía académica y su confianza en mi y en mi trabajo incluso cuando yo misma dudaba. Agradezco muchísimo cada corrección, cada comentario y cada impulso para seguir avanzando. Gracias por su dedicación.

A mis compañeros y al grupo de investigación, HPC4EAS, por el ambiente de aprendizaje, por las conversaciones distendidas, por los ánimos en los momentos complicados y por compartir este recorrido lleno de retos y aprendizajes.

Gracias a todos por caminar conmigo, y por recordarme que cada paso valía la pena.

Abstract

Optimization has become a cornerstone of *High-Performance Computing* (HPC) and modern computational science, enabling the efficient resolution of increasingly complex and large-scale problems. As complex systems, such as simulation models, logistic networks, or operational management frameworks, grow in dimensionality and uncertainty, for instance by incorporating additional decision variables, traditional deterministic or exhaustive search methods often become computationally infeasible. In this context, heuristic and metaheuristic approaches have emerged as flexible, adaptive, and evolving strategies capable of providing high-quality solutions within reasonable computational times. These methods do not aim to guarantee the absolute optimum but to approximate it efficiently by balancing exploration and exploitation in vast and irregular search spaces.

This doctoral research presents the development of a heuristic methodology *Montecarlo Clustering Search Algorithm (MCSA)* designed to address multiobjective combinatorial optimization problems under realistic conditions. The proposed approach integrates two complementary mechanisms:

- *Montecarlo* sampling, which performs a global and stochastic exploration of the solution space, and
- *clustering-based* refinement, which identifies promising regions and guides the search toward better solutions through iterative exploitation.

Starting from this simple yet powerful idea, sampling and grouping, the methodology evolves into a versatile optimization framework capable of handling high-dimensional and heterogeneous problems. Its performance was validated through *benchmark* mathematical functions, demonstrating its reliability and adaptability across different optimization landscapes.

Finally, the methodology was integrated with a simulation model of an Emergency Department (ED), allowing the algorithm to evaluate the real impact of each configuration under variable operational conditions. This integration highlights the practical applicability, robustness, and transferability of the proposed framework to real-world decision-making environments. Overall, this work contributes a generalizable and interpretable optimization methodology that bridges the gap between theoretical exploration and real-world application, advancing the role of heuristics in HPC-oriented optimization and complex system analysis.

Resum

L'optimització s'ha convertit en una pedra angular del *High-Performance Computing* (HPC) i de la ciència computacional moderna, permetent la resolució eficient de problemes cada vegada més complexos i de gran escala. A mesura que els sistemes complexos, com ara models de simulació, xarxes logístiques o marcs de gestió operativa, augmenten en dimensionalitat i incertesa, per exemple en incorporar un major nombre de variables de decisió, els mètodes deterministes o de cerca exhaustiva sovint esdevenen computacionalment inviables.

En aquest context, els enfocaments heurístics i metaheurístics han emergit com a estratègies flexibles, adaptatives i en evolució, capaces d'oferir solucions d'alta qualitat en temps de càlcul raonables. Aquests mètodes no pretenen garantir l'òptim absolut, sinó apropar-s'hi de manera eficient, equilibrant l'exploració i l'explotació en espais de cerca amplis i irregulars.

Aquesta recerca doctoral presenta el desenvolupament d'una metodologia heurística, anomenada *Montecarlo Clustering Search Algorithm* (MCSA), dissenyada per abordar problemes d'optimització combinatòria multiobjectiu en condicions realistes. La metodologia proposada integra dos mecanismes complementaris:

- el mostreig *Montecarlo*, que realitza una exploració global i estocàstica de l'espai de solucions, i
- el refinament basat en *clustering*, que identifica regions prometedores i guia la cerca cap a millors solucions mitjançant una explotació iterativa.

A partir d'aquesta idea senzilla però potent, mostrejar i agrupar, la metodologia evoluciona fins a convertir-se en un marc d'optimització versàtil, capaç de tractar problemes d'alta dimensionalitat i naturalesa heterogènia. El seu rendiment ha estat validat mitjançant funcions matemàtiques *benchmark*, demostrant la seva fiabilitat i adaptabilitat en diferents paisatges d'optimització.

Finalment, la metodologia s'ha integrat amb un model de simulació d'un Servei d'Urgències Hospitalàries (SUH), permetent que l'algorisme avalui l'impacte real de cada configuració sota condicions operatives variables. Aquesta integració posa de manifest la seva aplicabilitat pràctica, robustesa i transferibilitat a entorns reals de presa de decisions. En conjunt, aquest treball aporta una metodologia d'optimització generalitzable i interpretable que connecta l'exploració teòrica amb l'aplicació real, avançant en el paper de les heurístiques en l'optimització orientada a HPC i en l'anàlisi de sistemes complexos.

Resumen

La optimización se ha convertido en una piedra angular del *High-Performance Computing* (HPC) y de la ciencia computacional moderna, permitiendo la resolución eficiente de problemas cada vez más complejos y de gran escala. A medida que los sistemas complejos, tales como modelos de simulación, redes logísticas o de gestión operativa, aumentan en dimensionalidad e incertidumbre, por ejemplo al incorporar más variables de decisión, los métodos deterministas o de búsqueda exhaustiva suelen volverse computacionalmente inviables.

En este contexto, los enfoques heurísticos y metaheurísticos han emergido como estrategias flexibles, adaptativas y en constante evolución, capaces de proporcionar soluciones de alta calidad en tiempos de cómputo razonables. Estos métodos no buscan garantizar el óptimo absoluto, sino aproximarlos de manera eficiente, equilibrando exploración y explotación en espacios de búsqueda amplios e irregulares.

Esta investigación doctoral presenta el desarrollo de una metodología heurística denominada *Montecarlo Clustering Search Algorithm* (MCSA), diseñada para abordar problemas de optimización combinatoria multiobjetivo en condiciones realistas. La metodología propuesta integra dos mecanismos complementarios:

- el muestreo *Montecarlo*, que realiza una exploración global y estocástica del espacio de soluciones, y
- el refinamiento basado en *clustering*, que identifica regiones prometedoras y orienta la búsqueda hacia mejores soluciones

mediante una explotación iterativa.

A partir de esta idea sencilla pero potente, muestrear y agrupar, la metodología evoluciona hasta convertirse en un marco de optimización versátil, capaz de abordar problemas de alta dimensionalidad y naturaleza heterogénea. Su rendimiento fue validado mediante funciones matemáticas *benchmark*, demostrando su fiabilidad y adaptabilidad a distintos paisajes de optimización.

Finalmente, la metodología se integró con un modelo de simulación de un Servicio de Urgencias Hospitalarias (SUH), lo que permitió que el algoritmo evaluara el impacto real de cada configuración bajo condiciones operativas variables. Esta integración pone de manifiesto su aplicabilidad práctica, robustez y transferibilidad a entornos reales de toma de decisiones. En conjunto, este trabajo aporta una metodología de optimización generalizable e interpretable que conecta la exploración teórica con la aplicación real, avanzando en el papel de las heurísticas en la optimización orientada a HPC y en el análisis de sistemas complejos.

Índice general

Índice general	XIII
Índice de figuras	XV
Índice de tablas	XVII
1. Introducción General	1
1.1. Contexto	1
1.2. La optimización en HPC	2
1.2.1. Importancia de la Optimización en Informática: Tipos . . .	3
1.3. Aplicaciones y Métodos de Optimización en Investigación	5
1.3.1. Optimización y las Ciencias Computacionales	7
1.3.2. Algunos retos en la Optimización Moderna	9
1.4. Estructura del Trabajo	14
Glosario del Capítulo	19
2. Objetivos de la Investigación	21
2.1. Objetivo General	22
2.2. Hipótesis de la investigación	22
2.2.1. Objetivos Específicos	23
Glosario del Capítulo	28
3. Marco Teórico	33
3.1. Contexto	34
3.2. Motivación de la propuesta de Optimización	35

ÍNDICE GENERAL

3.2.1.	Definición Formal y Componentes Clave de la Optimización	36
3.2.2.	Elementos de la Optimización	37
3.3.	Métodos de Resolución en Problemas de Optimización	39
3.4.	Heurísticas: métodos iterativos y métodos exploratorios	40
3.4.1.	Heurísticas y Metaheurísticas en la Resolución de Problemas Complejos	41
3.5.	Fundamentación Teórica y Diseño del Método Propuesto	42
3.6.	Trabajo Relacionado	43
3.7.	Fundamentos del Algoritmo base	44
3.7.1.	Limitaciones del modelo de referencia	46
3.7.2.	Evaluación del algoritmo base a través de <i>benchmarks</i>	49
3.7.3.	Validación Experimental	53
3.7.4.	Análisis y Discusión	56
3.8.	Conclusiones del Capítulo	61
	Glosario del Capítulo	64
4.	Metodología propuesta	67
4.1.	Presentación del Algoritmo	69
4.2.	Parametrización de MCSA	72
4.3.	MCSA Fase 1: Exploración	74
4.3.1.	Definición del espacio de búsqueda y generación de muestras aleatorias	76
4.4.	MCSA Fase 2: Explotación	79
4.4.1.	Reducción del Mapa de Montecarlo: Procedimiento de Barrido/Corte	80
4.4.2.	Algoritmo de clustering DBSCAN y justificación del análisis basado en densidad	82
4.4.3.	Definición de regiones potenciales	84
4.4.4.	Formalización	88
4.5.	Evaluación del MCSA	89
4.5.1.	Resultados Experimentales	92
4.5.2.	Análisis y Discusión de los Resultados	95
4.6.	Conclusiones del Capítulo	98

Glosario del Capítulo	101
5. Aplicación de la Metodología y del Modelo propuesto: El problema de la mochila	107
5.1. Justificación de la propuesta	108
5.2. Objetivo metodológico de la propuesta	109
5.3. Antecedentes y Trabajos Relacionados	110
5.4. Formulación: el Problema de la Mochila Binaria (<i>KP01</i>)	112
5.4.1. Definición formal del KP01	112
5.4.2. Métodos clásicos de resolución y Herramientas modernas de resolución	113
5.4.3. Ventajas de MCSA como estrategia alternativa	114
5.5. Implementación del algoritmo MCSA en el KP01	116
5.5.1. Antecedentes	116
5.6. Aplicación del MCSA al KP01	119
5.6.1. Evaluación del MCSA en el KP01	122
5.6.2. Resultados Experimentales	124
5.7. Discusión de los Resultados	131
5.8. Adaptación de la metodología al problema de la mochila multiobjetivo (MOKP)	133
5.8.1. Definición formal del Problema de la Mochila Multiobjetivo	134
5.8.2. Resultados de la implementación	136
5.9. Conclusión del capítulo	137
Glosario del Capítulo	140
6. Aplicación del Modelo en la gestión del personal en un Servicio de Urgencias Hospitalarias utilizando la simulación	143
6.1. Justificación y contexto de la propuesta de aplicación	144
6.2. Objetivo de la propuesta	148
6.3. Metodología para la configuración de personal en un Servicio de Urgencias	149
6.4. Simulación como herramienta de evaluación y soporte a la optimización	151

ÍNDICE GENERAL

6.5.	Hipótesis	156
6.6.	Estado del Arte	157
6.7.	El heurístico: aplicación del MCSA al problema de gestión del personal sanitario	161
6.7.1.	Integración de MCSA con el simulador de urgencias	163
6.7.2.	Formulación del problema: Fase de Exploración mediante MCSA	164
6.7.3.	Restricciones	173
6.7.4.	Fase de Explotación	174
6.8.	Validación Experimental	178
6.8.1.	Diseño Experimental	180
6.8.2.	Resultados Experimentales	181
6.9.	Validación Extendida: Integración de Variables de Recursos Humanos y Físicos	184
6.9.1.	Configuración y Ejecución	187
6.9.2.	Discusión de los Resultados	190
6.10.	Conclusiones del Capítulo	193
	Glosario del Capítulo	196
7.	Conclusiones, principales aportaciones y líneas abiertas	199
7.1.	Hallazgos y Conclusiones	199
7.2.	Líneas Futuras	201
	Apéndice	204
A.	Resultados adicionales	207
A.1.	Pruebas adicionales sobre funciones <i>benchmark</i>	207
A.2.	Resultados detallados: Boxplots de las funciones del CEC2017 . . .	213
	Nomenclatura	215
	Referencias	223

Índice de figuras

1.1. Taxonomía de los métodos de Optimización (adaptada de [50]). . .	7
1.2. Metodología para resolver un problema de optimización.	9
3.1. Modelo conceptual del algoritmo base. Un problema inicial, ubicado en un espacio continuo es discretizado. Aquí inicia el proceso de muestreo aleatorio y reducción de espacio, buscando Regiones Factibles donde finalmente se llevará a cabo una búsqueda exhaustiva localizada.	46
3.2. Función benchmark Sphere agrupada mediante K-Means utilizando diferentes valores de k	55
3.3. Agrupamiento de la función benchmark Ackley mediante K-Means utilizando distintos valores de k , mostrando la variación en las particiones del dominio	56
3.4. Representación de la función Hölder Table y su partición en clusters mediante K-Means para diferentes valores de k . Se observan los cambios en la segmentación del paisaje según la granularidad del clustering, destacando cómo valores mayores de k permiten identificar regiones más locales y abruptas de la función.	57
3.5. Aplicación de K-Means a la función Michalewicz utilizando diferentes valores de k . A medida que aumenta el número de clusters, el algoritmo captura con mayor detalle las regiones estrechas que contienen los mínimos locales profundos, facilitando su identificación en un paisaje altamente multimodal.	57

ÍNDICE DE FIGURAS

3.6. Función benchmark Sphere agrupada mediante K-Means y diferentes k	58
3.7. Clusterización de la función Ackley mediante K-Means basada en los valores de $f(x)$. El pico global queda aislado en un cluster, mostrando cómo esta estrategia permite concentrarse en regiones de interés y reducir efectivamente el espacio de búsqueda. Se muestran variaciones al utilizar diferentes valores de k	60
3.8. Clusterización de la función Sphere mediante K-Means utilizando los valores de la función objetivo $f(x)$. Se observan estructuras concéntricas que reflejan los niveles de la función y facilitan la identificación del mínimo global. Se muestran variaciones al utilizar diferentes valores de k	60
3.9. Clusterización de la función Hölder Table mediante K-Means usando los valores de $f(x)$. Aunque se identifican algunas regiones con valores deseables, la alta multimodalidad y la presencia de múltiples mínimos locales dificultan la reducción efectiva del dominio. Se pueden ver variaciones al utilizar diferentes valores de k	61
3.10. Aplicación de K-Means a la función Michalewicz utilizando los valores de $f(x)$. La función multimodal extrema genera clusters que no permiten acotar fácilmente el espacio de búsqueda, evidenciando limitaciones cuando se trabaja con mínimos estrechos y múltiples regiones relevantes. Se muestran variaciones al utilizar diferentes valores de k	61
4.1. Diagrama de flujo de la heurística MCSA, ilustrado mediante dos casos de aplicación (<i>benchmark</i>) [44]. Este esquema proporciona una visión general de los principales componentes y del flujo del MCSA del proceso de optimización.	70

4.2. La nube de puntos es una representación en la que cada punto del gráfico corresponde a una muestra $(x_1, x_2, f(x))$. La dispersión inicial refleja la cobertura del espacio; a medida que avanza la Exploración y se activa el criterio de parada, definido como la condición que detiene la Exploración cuando se alcanza la precisión deseada en la media funcional o la cobertura máxima del espacio, el proceso finaliza evitando cálculos innecesarios y optimizando la eficiencia del muestreo.	71
4.3. Esquema general de la Fase de Exploración del MCSA.	74
4.4. Comparativa de técnicas de agrupamiento.	78
4.5. Diagrama de flujo para la determinación de la línea de corte en la Fase de Explotación del MCSA, combinando segmentación por intervalos y evaluación iterativa por DBSCAN.	81
4.6. Ejemplo de corte y agrupamiento sobre una función benchmark compuesta.	82
4.7. Clasificación de muestras en el algoritmo DBSCAN para $\text{minPts} = 6$. Los puntos núcleo (core), de borde (border) y de ruido (noise) se identifican en función de su densidad local.	83
4.8. Ejemplo del corte aplicado sobre el mapa de Montecarlo para la función de Michalewicz, mostrando la línea de corte seleccionada tras el barrido por niveles. Se pueden observar dos regiones en la parte inferior de la línea de corte, que serán seleccionadas como regiones potenciales.	86
4.9. Proceso de construcción de regiones potenciales en dos iteraciones consecutivas del MCSA. Las regiones mostradas corresponden a las dos zonas destacadas en la figura anterior, vistas en el plano $[X, Y]$ (vista superior), identificando los picos de la función de Michalewicz como regiones potenciales seleccionadas para el refinamiento. (1) Selección de la mejor muestra en cada cluster, (2) cálculo de la longitud del cluster en cada dimensión, (3) definición de la nueva zona de búsqueda centrada en el óptimo local.	87

ÍNDICE DE FIGURAS

4.10. Exploración del espacio de soluciones mediante muestreo Montecarlo y refinamiento de soluciones sobre las funciones: Ackley y Sphere.	93
4.11. Aplicación de la metodología sobre las funciones: XinSheYang y Levi N.13.	95
5.1. Flujo desde el proceso de muestreo hasta la evaluación de cada muestra que promueve la creación de un mapa de Montecarlo	120
5.2. Representación del proceso de distribución y ordenación de 6 elementos en dos dimensiones respetando el tamaño del problema. .	121
5.3. Representación del proceso de organización del espacio de búsqueda antes de la Fase de Exploración. Se propone un reordenamiento ascendente según el valor de la función objetivo (índice de Montecarlo). Finalmente, el cubo conceptual ilustra cómo se asignan estas combinaciones a lo largo de los ejes del espacio multidimensional, en preparación para la búsqueda mediante MCSA.	126
5.4. Transición a Fase de Explotación. a)Proceso de corte para reducir el espacio de búsqueda y b) Aplicación de DBSCAN para reducir el espacio, seleccionar las muestras más viables y posteriormente aplicar K-means.	126
5.5. Representación de una muestra final generada por MCSA. Cada celda destacada indica la combinación de elementos seleccionada en una dimensión específica. La muestra completa está compuesta por la integración de estas combinaciones, una por dimensión, lo que define una solución candidata dentro del espacio de búsqueda. . . .	127
6.1. Proceso de clasificación de pacientes según la urgencia de su atención médica en un Servicio de Urgencias (elaboración propia adaptada de figura de referencia general.	147
6.2. Este esquema ilustra cómo la simulación desempeña un papel clave al conectar la búsqueda abstracta (MCSA) con la evaluación del rendimiento en el mundo real.	161

6.3.	Diagrama de flujo metodológico que muestra la interacción entre el modelo basado en agentes (ABM) y el algoritmo MCSA. El esquema representa el bucle de retroalimentación en el que MCSA propone configuraciones de personal como entrada al simulador. Este realiza la simulación de cada configuración y genera los indicadores clave de rendimiento (KPI(s)), los cuales guían la siguiente iteración del proceso de Exploración. La estructura refleja la lógica central del marco de optimización.	164
6.4.	Diagrama que ilustra el proceso de muestreo realizado por el algoritmo MCSA para generar configuraciones de personal, seguido por su evaluación mediante el modelo basado en agentes (ABM), el cual devuelve los valores correspondientes de los indicadores clave de rendimiento (KPI(s)). MCSA genera y refina iterativamente nuevas muestras hasta alcanzar la convergencia.	169
6.5.	Visualización del espacio de búsqueda antes y después de la detección de regiones densas, es decir, áreas con alta concentración de muestras. Para llevar a cabo el proceso de corte efectivo, primero se divide el mapa a lo largo del eje $f(\mathbf{x})$ para localizar regiones de alta densidad.	175
6.6.	Una vez identificadas las regiones densas, se aplica el algoritmo de K-Means para segmentar los datos en cuatro clusters	177
6.7.	Flujo de pacientes a través del ED. El sistema se compone de ocho procesos interdependientes, numerados en los círculos, que regulan el tránsito de los pacientes. Las áreas A y B representan flujos diferenciados para pacientes urgentes y no urgentes, respectivamente, cada una con personal independiente. Modelo de: [61].	179
A.1.	Determinación del número óptimo de clusters (k) para las muestras Montecarlo de la función Eggholder mediante el método de la silueta. El valor máximo de puntuación se obtiene para $k = 4$, lo que confirma que cuatro clusters son los más representativos de la estructura de los datos.	209

A.2. Visualización de la clusterización de la función <i>benchmark</i> Eggholder usando K-Means con $k = 4$. Se observa que los centroides detectados no reflejan correctamente la estructura de los datos provenientes del muestreo Montecarlo, dividiendo un mismo conglomerado en múltiples clusters aparentemente arbitrarios y agrupando puntos alejados en la misma clase.	210
A.3. Resultados de K-Means con ($k=7$) a un conjunto de puntos de muestreo en un espacio de decisión bidimensional (X,Y), típico de las regiones óptimas de la función Eggholder.	211
A.4. Visualización de la clusterización de la función <i>benchmark</i> Eggholder usando K-Means con $k = 40$. Se observa la aparición de múltiples centroides que no representan adecuadamente la estructura del mapa de Montecarlo, generando clusters fragmentados y dispersos. Esta configuración es además costosa desde el punto de vista computacional y no ofrece mejoras significativas en la interpretación de la estructura de los datos. . . .	212
A.5. Boxplots de los resultados del MCSA para las funciones benchmark del CEC2017, que evidencian la robustez del algoritmo frente a la variabilidad derivada de múltiples ejecuciones.	214
A.6. Distribución de los resultados obtenidos por el MCSA en las funciones benchmark del CEC2017, mostrando la consistencia del algoritmo frente a la variabilidad entre ejecuciones.	214

Índice de tablas

1.1. Glosario del Capítulo	19
2.1. Glosario del Capítulo	28
3.1. Funciones <i>benchmark</i> utilizadas en la literatura de optimización. . .	52
3.2. Funciones <i>benchmark</i>	52
3.3. Glosario del Capítulo	64
4.1. Resultados de la validación utilizando funciones clásicas de <i>benchmark</i> . 92	
4.2. Resultados de las pruebas con funciones benchmark del CEC2017. .	96
4.3. Glosario del Capítulo	101
5.1. Distribución multidimensional de elementos en el espacio de búsqueda para la instancia <i>EX01</i> del problema de la mochila, con 24 elementos.	123
5.2. Solución para la instancia <i>EX01</i> . Se representan las posibles combinaciones de los elementos de la dimensión 1 (de un total de 8). 125	
5.3. Resultados de MCSA de la instancia <i>EX01</i> resuelto en 8 dimensiones. Se muestra la mejor solución obtenida, destacando el peso total asociado a ella que cumple con la restricción del problema. 128	
5.4. Detalle de las instancias a resolver: de <i>EX01</i> a <i>EX06</i>	129
5.5. Resultados para diferentes instancias del KP01: de <i>EX01</i> a <i>EX06</i> , resueltas por medio del MCSA	130
5.6. Resultados obtenidos para la instancia <i>EX01</i> adaptada como problema multiobjetivo con distintos factores de importancia. . . .	136

ÍNDICE DE TABLAS

5.7. Glosario del Capítulo	140
6.1. Comparación de técnicas de simulación en la gestión hospitalaria. En el presente estudio, se utiliza ABM implementada en NetLogo por su flexibilidad, portabilidad y capacidad de integración con entornos de alto rendimiento computacional (HPC).	152
6.2. Valores mínimos y máximos del eje para cada tipo de personal (Sr y Jr son dimensiones separadas). La restricción $Sr + Jr \leq 10$ se aplica por categoría.	181
6.3. Resumen de parámetros clave del problema y métricas obtenidas con el heurístico MCSA. Incluye la definición del espacio de búsqueda, tiempos de simulación, esfuerzo de muestreo y configuración eficiente identificada (asociada al LoS).	182
6.4. Configuración final de personal identificada por MCSA. Cada coordenada indica el número de profesionales asignados por rol y área, correspondientes a las dimensiones exploradas durante la optimización.	184
6.5. Definición del espacio de búsqueda multidimensional utilizado en la heurística MCSA. La tabla muestra todas las variables de decisión (personal médico y recursos físicos), junto con sus límites completos del eje y los límites operativos ajustados aplicados durante el proceso de optimización.	185
6.6. Resumen de los principales parámetros y resultados del proceso de optimización mediante MCSA. La tabla presenta la dimensionalidad del problema, el tamaño efectivo del espacio de búsqueda, y el número total de evaluaciones necesarias para identificar una configuración óptima. El resultado final se expresa en forma de coordenada, representando una solución factible que alcanzó el menor valor observado de <i>LoS</i>	189
6.7. Comparación entre la Asignación de Personal y Recursos de la Configuración base del personal sanitario y la Configuración encontrada por el MCSA.	190

6.8. Comparación entre los KPI(s) asociados a la configuración base del hospital y la solución propuesta por el método.	192
6.9. Glosario del Capítulo	196

ÍNDICE DE TABLAS

Capítulo 1

Introducción General

1.1. Contexto

El avance de la ciencia y la ingeniería en la actualidad está estrechamente vinculado al desarrollo de capacidades computacionales de alto rendimiento (HPC - High-Performance Computing). Estas tecnologías permiten abordar problemas de gran escala que involucran cantidades masivas de datos, simulaciones complejas y modelos matemáticos de alta dimensionalidad. Desde la predicción del clima hasta la bioinformática, el diseño industrial y la inteligencia artificial, HPC se ha convertido en un habilitador fundamental del descubrimiento científico y la innovación tecnológica.

Uno de los campos donde el uso de HPC resulta particularmente relevante es la optimización. La búsqueda de soluciones eficientes a problemas que requieren la minimización o maximización de funciones bajo ciertas restricciones es una necesidad transversal en numerosas disciplinas.

En este contexto, el presente trabajo se enmarca en la intersección entre la computación de alto rendimiento y la optimización matemática. Su principal objetivo es el diseño y aplicación de una metodología heurística capaz de abordar problemas de optimización combinatoria en espacios multidimensionales discretos.

1. INTRODUCCIÓN GENERAL

El enfoque propuesto integra técnicas de muestreo aleatorio y agrupamiento para explorar eficazmente espacios de búsqueda complejos y costosos de evaluar, como aquellos presentes en aplicaciones reales.

Aunque el algoritmo se desarrolla con un enfoque general, su aplicabilidad se demuestra en casos de estudio concretos. Entre ellos destacan el problema de la mochila y la optimización de recursos en un servicio de atención médica. En estos escenarios, se requiere evaluar configuraciones de gran tamaño y tomar decisiones bajo múltiples restricciones, donde métodos clásicos no ofrecen soluciones prácticas en tiempos razonables.

De esta forma, el trabajo combina fundamentos teóricos de optimización con técnicas de programación de alto rendimiento y un enfoque aplicado, ofreciendo una contribución metodológica con potencial de aplicación en diferentes dominios. La metodología desarrollada busca ser escalable, flexible y orientada a la resolución de problemas reales donde se requieren soluciones de buena calidad en un tiempo computacional razonable.

1.2. La optimización en HPC

La optimización es un concepto fundamental en diversas ramas de la ciencia y la ingeniería, y en particular en informática. Se refiere al proceso de encontrar la mejor solución posible para un problema dentro de un conjunto dado de restricciones y condiciones. Formalmente, consiste en identificar un vector x^* dentro de un conjunto factible $\mathcal{D}$ que minimice o maximice una función objetivo $f(x)$, es decir:

$$x^* = \arg \min_{x \in S} f(x) \quad \text{o} \quad x^* = \arg \max_{x \in S} f(x) \quad (1.1)$$

Este proceso busca mejorar algún aspecto de un sistema, como el tiempo de ejecución, el uso de recursos o el rendimiento general, con el fin de alcanzar el mejor resultado posible bajo las condiciones establecidas.

Con el avance acelerado de la tecnología y la creciente complejidad de los sistemas

1. INTRODUCCIÓN GENERAL

informáticos, la optimización adquiere un papel esencial en el diseño y desarrollo de software y hardware. Su aplicación abarca desde la mejora de algoritmos hasta la optimización de procesos computacionales en infraestructuras de red, bases de datos y sistemas distribuidos.

Además, la optimización no solo permite mejorar la eficiencia y el desempeño de los sistemas, sino que también facilita la toma de decisiones en contextos donde las soluciones no son evidentes o donde el espacio de búsqueda es muy amplio y complejo. Este aspecto es especialmente relevante en problemas donde el costo computacional de la búsqueda exhaustiva es prohibitivo, lo que hace necesaria la utilización de técnicas aproximadas y heurísticas.

Por otro lado, la informática moderna se enfrenta a problemas cada vez más complejos debido al crecimiento exponencial de la cantidad de datos generados y la necesidad de procesarlos de manera eficiente. En este contexto, la optimización se ha consolidado como una herramienta fundamental para mejorar el rendimiento y la eficiencia de los sistemas informáticos, tanto a nivel de hardware como de software, así como para facilitar la toma de decisiones en entornos complejos.

En este capítulo se presentan los fundamentos y el contexto general de la optimización en informática, destacando su importancia, sus principales tipos y los retos que enfrenta en la actualidad. Asimismo, se introduce la problemática abordada en este trabajo y su relevancia en el ámbito de la investigación aplicada.

1.2.1. Importancia de la Optimización en Informática: Tipos

Hay diversos tipos de optimización que abordan distintos aspectos dentro de la informática. A continuación, se describen los principales tipos de optimización que tienen un impacto notable en este campo:

La **optimización de hardware**, está relacionada con la mejora de la arquitectura de los sistemas informáticos para reducir el consumo energético, mejorar la velocidad de procesamiento o maximizar la eficiencia en el uso

1. INTRODUCCIÓN GENERAL

de componentes. Esto incluye la optimización de microprocesadores, sistemas embebidos, dispositivos móviles y centros de datos. La reducción del consumo energético, el aumento de la velocidad de procesamiento y la eficiencia en el uso de componentes físicos son objetivos clave en esta área, especialmente en entornos con limitaciones de tamaño y recursos, como los dispositivos portátiles.

La **optimización de software**, por otro lado, se centra en mejorar el rendimiento de las aplicaciones a través de técnicas como la optimización del código fuente, la gestión eficiente de memoria, la paralelización de procesos y la reducción de la latencia en sistemas interactivos. Esto permite que los programas sean más rápidos, consuman menos recursos y ofrezcan una mejor experiencia al usuario final.

La **optimización de redes y comunicaciones** es un área que se ocupa de la mejora de la transmisión de datos y la gestión de redes de comunicación, abordando problemas típicos como la congestión de la red, la asignación eficiente (dinámica) de recursos de ancho de banda y la reducción de la latencia en la transmisión de información entre dispositivos. La optimización en este ámbito es importante para garantizar la calidad del servicio en redes modernas, como Internet, redes móviles y sistemas de comunicación en la nube.

La **optimización algorítmica**, por otro lado, busca diseñar y mejorar algoritmos para hacerlos más eficientes en términos de tiempo de ejecución y uso de recursos computacionales. Este tipo de optimización es importantísima en áreas como la búsqueda, la clasificación, el procesamiento de grandes volúmenes de datos y la resolución de problemas complejos. La mejora en los algoritmos contribuye directamente a la escalabilidad y robustez de los sistemas informáticos.

La **optimización en inteligencia artificial y aprendizaje automático** ha cobrado especial relevancia, ya que busca mejorar la precisión de los modelos y algoritmos, minimizando el error en la toma de decisiones y maximizando el rendimiento general del modelo en tareas como clasificación, predicción y reconocimiento de patrones. Los métodos de optimización permiten entrenar modelos más efectivos, y también adaptar sistemas inteligentes a entornos dinámicos y cambiantes.

1. INTRODUCCIÓN GENERAL

Resumiendo, la optimización en Informática, y orientada al ámbito de la investigación es indispensable no solo para maximizar el rendimiento de los sistemas informáticos, sino también para resolver una amplia variedad de problemas complejos, en especial aquellos en los que las soluciones no son evidentes o se encuentran en espacios de búsqueda grandes y en múltiples dimensiones, pasando por la toma de decisiones, lo que hace que la búsqueda de la solución óptima sea un reto. Su aplicación transversal abarca desde el diseño del hardware hasta la inteligencia artificial, reflejando su relevancia en el desarrollo tecnológico actual.

1.3. Aplicaciones y Métodos de Optimización en Investigación

En la investigación científica, muchos problemas complejos, que abarcan disciplinas como la simulación de procesos físicos, la investigación operativa y la inteligencia artificial, se abordan mediante el uso de métodos de optimización. Estos problemas suelen involucrar grandes cantidades de datos y variables interrelacionadas, lo que requiere encontrar soluciones que maximizan o minimizan un conjunto de parámetros, como el costo, el tiempo, la eficiencia o la precisión. La optimización se ha convertido en una herramienta para el diseño y la mejora de sistemas y modelos en una amplia variedad de áreas, desde la planificación de recursos hasta el análisis predictivo. Algunos ejemplos incluyen:

Optimización en la simulación científica. Las simulaciones que modelan fenómenos complejos, como la dinámica molecular, la predicción del clima o la ingeniería de sistemas, requieren optimización para ajustar parámetros y mejorar la precisión de los resultados obtenidos, mientras se minimizan los costos computacionales.

Optimización en la toma de decisiones. En áreas como la investigación operativa, la optimización se utiliza para resolver problemas relacionados con la asignación de recursos, la planificación de rutas, la programación de tareas o la

1. INTRODUCCIÓN GENERAL

maximización de la eficiencia en los sistemas productivos.

Optimización en el diseño de experimentos. En la investigación experimental, la optimización ayuda a diseñar experimentos de manera eficiente, seleccionando las condiciones adecuadas para maximizar los resultados mientras se minimizan los recursos y el tiempo invertido.

Aunque la optimización ofrece enormes beneficios en investigación, también presenta limitaciones significativas. La mayoría de los problemas de optimización no tienen una solución exacta debido a la complejidad del espacio de soluciones, la presencia de restricciones y la naturaleza no lineal de muchos problemas reales. En investigaciones avanzadas, donde el espacio de búsqueda puede ser multidimensional y altamente no lineal, la optimización global es necesaria, ya que los métodos clásicos de optimización local pueden quedar atrapados en soluciones subóptimas. Esto lleva al uso de algoritmos de optimización aproximada.

La optimización combinatoria se aplica en problemas donde se deben seleccionar combinaciones óptimas de elementos bajo restricciones, como en el caso de la asignación de recursos en la investigación operativa. Los algoritmos son fundamentales para abordar este tipo de dificultades, ya que permiten encontrar soluciones cercanas al óptimo de manera eficiente, incluso cuando los problemas son altamente no lineales, tienen muchas dimensiones o están sujetos a restricciones complejas. Por otro lado, la optimización continua está patente refiriéndose a aquellos problemas en los que las variables pueden tomar cualquier valor dentro de un rango y se busca encontrar el mínimo o máximo de una función continua, que puede ser lineal o no lineal. La optimización continua es común en la simulación de procesos físicos, donde las relaciones entre las variables no siguen una ley lineal, o en modelos de aprendizaje automático, donde se busca optimizar parámetros de manera que se ajusten a los datos eficientemente.

Como se ve en la Figura 1.1, este organigrama presenta la clasificación de los métodos de optimización, dividida en dos grandes áreas: Combinatoria y Continua. Dentro de la optimización combinatoria, se encuentran los métodos Exactos, que garantizan la solución óptima, y también métodos Aproximados, que incluyen

1. INTRODUCCIÓN GENERAL

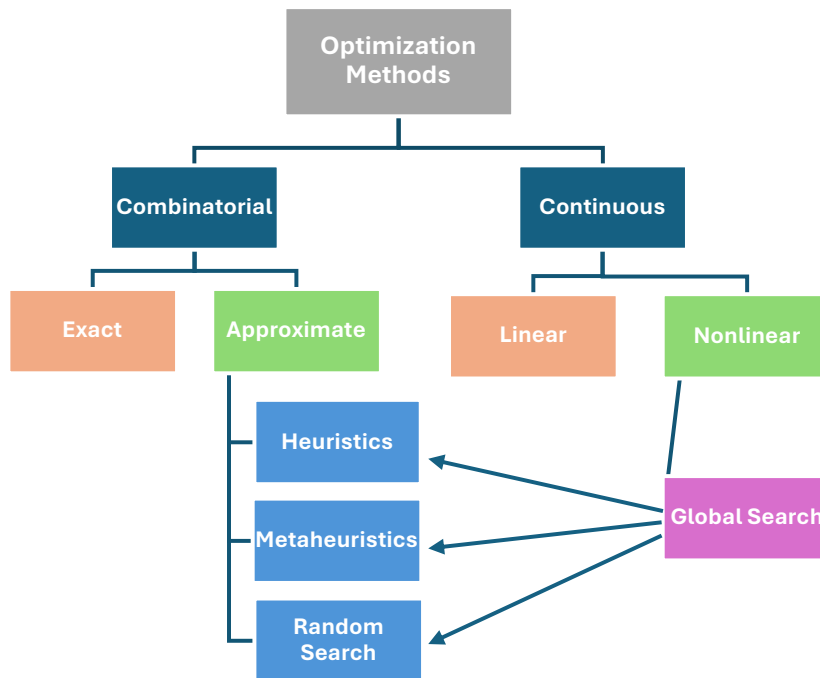


Figura 1.1: Taxonomía de los métodos de Optimización (adaptada de [50]).

Heurísticas, Metaheurísticas y Búsqueda Aleatoria, los cuales se utilizan cuando los métodos exactos no son viables. Por otro lado, la optimización continua se clasifica en Lineal y No Lineal, aunque en ambos casos también se pueden aplicar métodos heurísticos y metaheurísticos. Además, en la optimización no lineal se incluye la Búsqueda Global, que intenta explorar el espacio de soluciones de manera más exhaustiva. Las flechas en el diagrama indican cómo las Heurísticas, Metaheurísticas y la Búsqueda Aleatoria se utilizan tanto en la optimización combinatoria como en la continua, proporcionando técnicas eficaces para resolver problemas complejos de optimización donde los métodos exactos no son prácticos.

1.3.1. Optimización y las Ciencias Computacionales

Los problemas complejos, a menudo requieren técnicas avanzadas de optimización para manejar la alta dimensionalidad de los datos y las interacciones no lineales entre las variables. Por tanto, la optimización es esencial para el

1. INTRODUCCIÓN GENERAL

avance de la informática, especialmente en áreas como Big Data, Inteligencia Artificial y Sistemas Distribuidos. Por ejemplo, el entrenamiento de redes neuronales profundas depende de técnicas de optimización para ajustar millones de parámetros, mientras que en sistemas distribuidos la asignación eficiente de recursos computacionales es clave para el rendimiento global.

Además, la Computación de Altas Prestaciones (HPC) ha estado asociada principalmente a las ciencias clásicas, como la física, la astronomía y la química, o problemas complejos de ingeniería. Sin embargo, en la actualidad, las ciencias sociales también están comenzando a aprovechar el potencial de la simulación computacional. Los sistemas modelados en estas áreas, que pueden ser desde una población con comportamiento sanitario, una organización con procesos internos, entre otros, son altamente complejos y producen grandes volúmenes de datos. Simular estos sistemas requiere tiempos de cómputo prolongados en computadores convencionales. Además, los modelos y fenómenos que se estudian son inherentemente probabilísticos por lo que las ciencias sociales y los sistemas de decisión se benefician del uso de HPC para simular, analizar y comprender estos sistemas, así como para generar conocimiento basado en los datos obtenidos.

El presente trabajo, de naturaleza interdisciplinaria, se centra en el desarrollo y validación de una heurística avanzada para encontrar soluciones computacionalmente eficientes en problemas de optimización complejos. Aunque la metodología puede tener aplicaciones en distintos ámbitos, en esta tesis se estudian problemas clásicos y relevantes en informática y optimización, como el problema de la mochila y la gestión de personal en Servicios de Urgencias Hospitalarias (SUH). En lo que sigue, referiremos a estas unidades por sus siglas en inglés, *Emergency Department* (ED). La tesis se estructura en dos bloques principales: primero, el diseño, parametrización y análisis del algoritmo heurístico; y segundo, su aplicación práctica a estos problemas, evaluando su desempeño en entornos realistas.

En uno de los casos de estudio, se integra el algoritmo con un modelo de simulación previamente desarrollado por nuestro grupo de investigación *High-Performance Computing for Efficient Applications and Simulation* (HPC4EAS), para mejorar la configuración del staff médico en un servicio de urgencias hospitalarias. Esta

1. INTRODUCCIÓN GENERAL



Figura 1.2: Metodología para resolver un problema de optimización.

combinación interdisciplinaria, que vincula la informática, la ingeniería y la ciencia de la salud, demuestra el potencial de la heurística para optimizar recursos en entornos complejos mediante la colaboración entre optimización matemática y simulación.

Así, en este trabajo se recurre a simulaciones computacionales para modelar un sistema que no sigue una formulación matemática convencional, como se mencionó anteriormente. Para abordar estos retos de optimización, hemos adaptado nuestra metodología heurística específica para obtener soluciones eficientes y viables en contextos altamente complejos.

Nuestra propuesta consiste en una metodología heurística innovadora que integra técnicas de muestreo aleatorio y agrupamiento para abordar problemas de optimización combinatoria en espacios multidimensionales. Se espera que esta propuesta permita contribuir a mitigar limitaciones de los métodos tradicionales, ofreciendo una herramienta eficaz para problemas complejos con aplicaciones prácticas tanto en logística, planificación, salud y otros ámbitos, siguiendo el curso del diagrama 1.2.

1.3.2. Algunos retos en la Optimización Moderna

La optimización moderna enfrenta múltiples desafíos que surgen tanto de la creciente complejidad de los problemas a resolver como de las limitaciones inherentes a los métodos y recursos disponibles. Estos retos condicionan el desarrollo de nuevas técnicas y el mejoramiento de las existentes, y se presentan en diversas formas según el contexto y la naturaleza del problema. A continuación,

1. INTRODUCCIÓN GENERAL

se describen los principales desafíos actuales en el campo de la optimización:

Alta Dimensionalidad y Complejidad del Espacio de Búsqueda

Uno de los problemas más significativos en la optimización es el denominado *mal de la dimensionalidad* [14]. A medida que el número de variables o dimensiones del problema crece, el espacio de búsqueda se expande exponencialmente, dificultando la Exploración completa y eficiente. Esto provoca que los métodos tradicionales basados en búsquedas exhaustivas sean inviables debido a los costos computacionales exorbitantes, haciendo necesaria la implementación de estrategias que puedan manejar esta complejidad de forma inteligente.

Presencia de Múltiples Óptimos Locales

En muchos problemas reales, especialmente aquellos con funciones objetivo no lineales y multimodales, existen numerosos óptimos locales que pueden atrapar a los algoritmos de optimización, impidiendo que alcancen el óptimo global. Este fenómeno dificulta la convergencia hacia soluciones verdaderamente óptimas y obliga a diseñar métodos que puedan escapar de estos puntos de atracción y realizar una Exploración más global del espacio.

Restricciones Complejas y Dinámicas

Los problemas de optimización suelen estar sujetos a múltiples restricciones, que pueden ser lineales o no lineales, estáticas o dinámicas. La incorporación de estas restricciones incrementa la dificultad del problema, pues obliga a los algoritmos a buscar soluciones factibles dentro de regiones específicas del espacio. La gestión eficiente de estas restricciones, sin sacrificar la calidad de la solución ni la eficiencia computacional, representa un reto importante.

Limitaciones en la Evaluación de la Función Objetivo

En muchas aplicaciones prácticas, la evaluación de la función objetivo puede ser costosa en tiempo y recursos, como ocurre en simulaciones complejas, experimentos físicos o modelos computacionales avanzados. Esta limitación impone la necesidad de desarrollar métodos que reduzcan el número de evaluaciones necesarias,

1. INTRODUCCIÓN GENERAL

mediante aproximaciones, muestreos inteligentes o técnicas de *surrogate modeling* (modelos sustitutos).

Escalabilidad y Adaptabilidad de los Algoritmos

La capacidad de los algoritmos para escalar adecuadamente con el tamaño del problema y adaptarse a cambios en las condiciones del entorno es otro reto. Los métodos deben ser robustos frente a variaciones en los parámetros, cambios en las restricciones y nuevos datos, manteniendo un rendimiento eficiente sin requerir ajustes manuales constantes.

Balance entre Exploración y Explotación

Encontrar un equilibrio adecuado entre dos fuerzas opuestas pero complementarias: la Exploración del espacio de soluciones (búsqueda global) y la Explotación de las regiones prometedoras (búsqueda local). Este balance es lo que define el éxito de los algoritmos heurísticos y metaheurísticos. Una Exploración insuficiente puede conducir a soluciones subóptimas, mientras que una Exploración excesiva puede resultar en un desperdicio de recursos computacionales y tiempo.

La Exploración hace referencia a la capacidad del algoritmo para recorrer de manera amplia el espacio de soluciones posibles, con el objetivo de evitar quedar atrapado en óptimos locales y de identificar zonas no visitadas que podrían contener mejores soluciones. Esta fase es esencial en etapas tempranas de la búsqueda o cuando la información sobre la estructura del problema es limitada.

Por otro lado, la Explotación consiste en refinar las soluciones ya conocidas, enfocándose en áreas del espacio de búsqueda que han mostrado buen rendimiento hasta el momento. Esta intensificación permite mejorar la calidad de las soluciones y acelerar la convergencia hacia un óptimo (local o global).

Un desbalance entre ambas estrategias puede comprometer seriamente la efectividad del algoritmo: una Explotación prematura puede llevar al estancamiento en soluciones subóptimas, mientras que una Exploración excesiva puede traducirse en una búsqueda ineficiente y en un mayor consumo de tiempo y recursos computacionales sin una mejora sustancial de las soluciones.

1. INTRODUCCIÓN GENERAL

Por esta razón, muchos algoritmos modernos incorporan mecanismos adaptativos que ajustan dinámicamente el grado de Exploración y Explotación a lo largo del proceso de búsqueda. Ejemplos incluyen el enfriamiento gradual en algoritmos como *Simulated Annealing*, o los esquemas de actualización de población en algoritmos evolutivos y de enjambre, que favorecen la Exploración al inicio y gradualmente se enfocan en la Explotación conforme progresa la búsqueda [103].

Además, el uso de estrategias de diversidad poblacional, procedimientos de reinicio (*restarting*), o estrategias híbridas (como el uso de búsqueda local intensiva combinada con Exploración global) ha demostrado ser una buena estrategia para mantener un equilibrio eficaz. En problemas complejos, donde la superficie del espacio de búsqueda es altamente no lineal o multimodal, este equilibrio no solo mejora la calidad de la solución, sino que también garantiza una mayor robustez del algoritmo frente a diferentes instancias del problema.

Integración con Sistemas Complejos y Multidisciplinarios

Finalmente, en el contexto de aplicaciones reales, los problemas de optimización rara vez se presentan de forma aislada. A menudo, están profundamente integrados en sistemas complejos y multidisciplinarios que involucran datos heterogéneos, incertidumbre y múltiples objetivos conflictivos donde interactúan múltiples subsistemas, cada uno con dinámicas propias, restricciones particulares y objetivos específicos. Estos entornos pueden abarcar desde la planificación de infraestructuras inteligentes y redes de transporte hasta sistemas ciberfísicos, procesos industriales automatizados y plataformas de atención sanitaria avanzada.

Una característica común de estos escenarios es precisamente la presencia de datos heterogéneos, estructurados y no estructurados, provenientes de diversas fuentes, a menudo en tiempo real, lo que complica su integración y análisis. Además, la toma de decisiones en estos sistemas se ve afectada por un alto grado de incertidumbre (por ejemplo, demanda fluctuante, fallos imprevisibles o comportamiento humano no lineal) y por la existencia de múltiples objetivos que, en muchos casos, son conflictivos entre sí (como minimizar costos mientras se maximiza la calidad del servicio).

1. INTRODUCCIÓN GENERAL

Para abordar esta complejidad, las técnicas tradicionales de optimización resultan insuficientes. Es necesario recurrir a enfoques avanzados como la *optimización multiobjetivo*, que permite encontrar soluciones de compromiso (óptimos de Pareto), así como a métodos *estocásticos*, que consideran la variabilidad inherente a los datos y al entorno. Asimismo, el desarrollo de *estrategias híbridas*, que combinan distintos paradigmas, como algoritmos evolutivos con simulación, o aprendizaje automático con optimización matemática, se ha vuelto una parte importante para lograr soluciones robustas, escalables y aplicables en la práctica.

En resumen, la optimización moderna debe evolucionar hacia un enfoque sistémico e integrador, capaz de operar eficazmente en entornos multidisciplinarios complejos y de adaptarse a la incertidumbre inherente a los sistemas reales. Este enfoque implica gestionar simultáneamente múltiples objetivos y restricciones, garantizando que las soluciones propuestas sean no solo óptimas desde un punto de vista matemático, sino también viables y sostenibles en contextos prácticos.

Para lograrlo, es necesario combinar de manera coherente avances en teoría de optimización, algoritmos metaheurísticos y capacidad computacional, explorando nuevas estrategias que permitan abordar problemas de creciente tamaño, complejidad y dinamismo. La integración de técnicas de muestreo estocástico, algoritmos híbridos, búsqueda local intensiva y simulaciones basadas en agentes constituye un ejemplo de cómo la innovación metodológica puede traducirse en mejoras tangibles en la calidad y eficiencia de las soluciones.

Asimismo, la optimización moderna debe incorporar mecanismos que garanticen robustez y resiliencia frente a variaciones del sistema, permitiendo generar conjuntos de soluciones diversas y flexibles que faciliten la toma de decisiones en escenarios reales. Este enfoque multidimensional no solo mejora la calidad de los resultados, sino que también proporciona herramientas prácticas para enfrentar desafíos emergentes en planificación, asignación de recursos y diseño de sistemas complejos.

En definitiva, la investigación en optimización debe avanzar hacia métodos capaces de equilibrar Exploración y Explotación, manejar incertidumbre, integrar múltiples

1. INTRODUCCIÓN GENERAL

fuentes de información y ofrecer soluciones operativamente relevantes. Solo de este modo se podrá garantizar que los desarrollos teóricos y computacionales se traduzcan en beneficios reales para la ciencia, la ingeniería y la gestión de sistemas complejos. Esta motivación constituye el punto de partida para el diseño, implementación y validación del algoritmo heurístico desarrollado a lo largo de los Capítulos de esta Tesis.

1.4. Estructura del Trabajo

El presente documento se organiza en varios capítulos con el objetivo de guiar al lector a través del desarrollo y los resultados de esta investigación. En primer lugar, se presenta la Introducción y el Contexto del estudio, seguida de la exposición de los Objetivos general y específicos. A continuación, se desarrolla el Marco Teórico y los Antecedentes relevantes. Posteriormente, se describe la metodología propuesta, y su análisis. Seguidamente, se presentan los casos de estudio, comenzando con el problema de la mochila binaria (KP01) y continuando con la aplicación de la metodología propuesta en la gestión de personal en un Servicios de Urgencias Hospitalarias. Finalmente, se presentan las Conclusiones, principales Aportaciones y Líneas de Trabajo Futuro. Además, al final de cada capítulo de esta tesis, se incluye un glosario (p. ej. [1.1](#) que resume los términos y conceptos clave utilizados, con el fin de facilitar la lectura y comprensión de los contenidos presentados:

- **Capítulo 1: Introducción y Contexto.** Se introducen los conceptos fundamentales relacionados con la optimización y su relevancia en el ámbito de la informática y otras disciplinas aplicadas. Asimismo, se discuten diversas áreas de aplicación, especialmente en contextos de investigación científica y tecnológica, destacando el papel de la optimización en la resolución de problemas complejos. Finalmente, se abordan los principales desafíos actuales en este campo, sentando las bases que justifican la motivación y pertinencia del trabajo desarrollado.
- **Capítulo 2: Objetivos.** Se presentan de forma clara y estructurada los objetivos que guían el desarrollo del trabajo, distinguiendo entre el objetivo

1. INTRODUCCIÓN GENERAL

general y los objetivos específicos. Estos últimos permiten descomponer el propósito central en metas concretas y alcanzables, que orientan las etapas metodológicas, experimentales y analíticas de la investigación.

- **Capítulo 3: Marco Teórico y Antecedentes.** Este capítulo presenta los fundamentos conceptuales y metodológicos que sustentan el desarrollo de este trabajo. En primer lugar, se abordan los conceptos clave asociados a la optimización, las heurísticas y metaheurísticas, así como las herramientas matemáticas y computacionales que permiten operar en espacios de búsqueda complejos y de alta dimensionalidad. Estos elementos proporcionan el marco teórico necesario para comprender las técnicas empleadas en el diseño de algoritmos avanzados.

Asimismo, se lleva a cabo una revisión de la literatura relevante, que incluye trabajos previos relacionados sobre optimización heurística y sus aplicaciones en problemas complejos. Esta revisión permite identificar las limitaciones presentes en los enfoques existentes, en especial de los métodos exhaustivos, y justificar la necesidad de la metodología propuesta.

Finalmente, se contextualiza este estudio dentro de la línea de investigación previamente desarrollada, sobre la cual se construye la presente propuesta. Aunque la presente propuesta no parte completamente desde cero, supone una evolución significativa dentro de la misma línea de investigación, ampliando y mejorando las técnicas anteriores mediante la incorporación de nuevos mecanismos metodológicos. De este modo, el capítulo articula tanto los fundamentos teóricos como los antecedentes científicos que motivan y sustentan el enfoque adoptado.

- **Capítulo 4: Metodología propuesta.** En este capítulo se presenta en detalle la metodología central desarrollada en este trabajo: el Montecarlo Clustering Search Algorithm (MCSA). En primer lugar, se introduce el propósito del capítulo y se describe el algoritmo propuesto, incluyendo su estructura general, los principios que lo sustentan y los parámetros clave que determinan su comportamiento.

1. INTRODUCCIÓN GENERAL

A continuación, el capítulo se divide en dos grandes fases que conforman el MCSA. La Fase 1, de Exploración, explica cómo se genera el Mapa de Montecarlo, cómo se procesa y reduce para identificar regiones relevantes del espacio de búsqueda. En la Fase 2, de Explotación, se desarrolla la formalización del proceso mediante el cual el algoritmo concentra la búsqueda en las zonas más prometedoras, integrando los resultados obtenidos previamente para guiar la optimización hacia configuraciones de alto rendimiento.

Posteriormente, el capítulo aborda la evaluación empírica del MCSA, donde se valida su desempeño tanto con funciones de benchmark clásicas como con los conjuntos de prueba del CEC2017, permitiendo comparar su comportamiento con estándares reconocidos de la literatura. Finalmente, se presentan las conclusiones del capítulo, consolidando las aportaciones fundamentales de la metodología propuesta.

- **Capítulo 5: Aplicación de la metodología y del modelo propuesto: El problema de la Mochila.** El Capítulo 5 presenta la aplicación de la metodología propuesta al Problema de la Mochila Binaria (KP01), un caso de estudio ampliamente reconocido por su complejidad y su relevancia dentro del ámbito de la optimización combinatoria. El capítulo justifica la elección de este problema como banco de pruebas y define con claridad el objetivo metodológico de esta aplicación. Asimismo, se revisan los principales trabajos relacionados y las aproximaciones clásicas y modernas que se emplean habitualmente para resolver este tipo de problemas.

A continuación, se expone la formulación formal del KP01, destacando sus principales características y la naturaleza combinatoria del espacio de soluciones. Posteriormente, el capítulo describe la implementación del MCSA sobre el KP01, abordando la estructura general del algoritmo y presentando detalladamente sus dos fases fundamentales: la Fase de Exploración, orientada a caracterizar el espacio de búsqueda y la Fase de Explotación, donde se concentran los esfuerzos de optimización en las configuraciones más prometedoras. Una vez adaptado el MCSA al problema,

1. INTRODUCCIÓN GENERAL

se presenta un estudio de la validación de su comportamiento en diferentes instancias y el análisis de los resultados obtenidos, destacando su capacidad para identificar soluciones de calidad competitiva.

Finalmente, el capítulo amplía la metodología hacia el Problema de la Mochila Multiobjetivo (MOKP). Se introduce su formulación formal y se analizan los resultados derivados de la adaptación multiobjetivo del MCSA, mostrando cómo el algoritmo puede extenderse para abordar problemas con múltiples objetivos, manteniendo su estructura general pero incorporando nuevos mecanismos para gestionar la naturaleza multiobjetivo del problema.

■ **Capítulo 6: Aplicación del Modelo en la gestión del personal en un Servicio de Urgencias Hospitalarias usando la simulación.**

El Capítulo 6 presenta la aplicación práctica del modelo propuesto en el contexto real de la gestión del personal en un Servicio de Urgencias Hospitalarias (o ED), incorporando técnicas de simulación como herramienta central de apoyo a la decisión. El capítulo se inicia ofreciendo el contexto y la justificación de este caso de estudio, destacando la relevancia de la asignación eficiente de recursos humanos en entornos de alta incertidumbre y elevada presión asistencial. A continuación, se formulan los objetivos específicos de la propuesta, orientados a evaluar la capacidad del MCSA para abordar problemas complejos de configuración de personal.

Posteriormente, se desarrolla una metodología detallada que combina el algoritmo con un simulador de urgencias, estableciendo un marco integrado para explorar y optimizar configuraciones de personal. El capítulo profundiza en la aplicación del MCSA al problema de planificación de personal, explicando su integración con el simulador, la formulación del problema, las restricciones operativas y la estructura de las fases de Exploración y Explotación. Seguidamente, se presenta un proceso de validación experimental, incluyendo diseño, ejecución y análisis de resultados, seguido de una validación extendida que incorpora variables adicionales de recursos humanos y físicos para enriquecer el modelo.

1. INTRODUCCIÓN GENERAL

- **Capítulo 7: Conclusiones, principales aportaciones y líneas abiertas.** En este capítulo se recogen las conclusiones generales del trabajo, sintetizando los principales hallazgos y aportaciones de la tesis. Asimismo, se plantean distintas líneas de trabajo futuro orientadas a consolidar y ampliar el alcance de la metodología propuesta.

Glosario del Capítulo

A continuación se presenta el glosario del Capítulo 1.1, que incluye los términos y símbolos más relevantes utilizados en esta sección. Este recurso tiene como objetivo facilitar la comprensión de la terminología empleada y servir de referencia rápida durante la lectura.

Tabla 1.1: Glosario del Capítulo

Término	Definición
Optimización	Proceso de encontrar la mejor solución posible dentro de un conjunto de restricciones
HPC	<i>High-Performance Computing</i>
Complejidad computacional	Medida del coste en tiempo o memoria asociado a la ejecución de un algoritmo, en términos de la dimensión del problema y de la cantidad de evaluaciones realizadas.
Heurística	Estrategia de búsqueda aproximada que proporciona soluciones satisfactorias en tiempos razonables, sin garantizar optimalidad pero explorando eficientemente espacios complejos.
Metaheurística	Es un marco algorítmico de alto nivel, independiente del problema específico, que orienta y regula el uso de heurísticas y mecanismos de Exploración y Explotación del espacio de búsqueda, con el objetivo de abordar problemas de optimización complejos o de gran dimensionalidad de manera flexible y eficaz.

Continúa en la siguiente página

1. INTRODUCCIÓN GENERAL

Glosario del Capítulo (continuación)

Término	Definición
Escalabilidad	Se refiere a la propiedad de un algoritmo por la cual, ante un incremento significativo en el tamaño o dimensionalidad de las instancias, la relación entre su complejidad asintótica y la calidad de las soluciones producidas permanece estable, permitiendo abordar problemas de gran escala sin que la complejidad se vuelva intratable.
ED / SUH	<i>Emergency Department</i> (Departamento de Urgencias, en inglés), referido a los Servicios de Urgencias Hospitalarias (SUH) en esta tesis. Las referencias a ED en adelante corresponden a este tipo de unidades hospitalarias.
Mal de la dimensionalidad	Fenómeno que ocurre en problemas de optimización y análisis de datos cuando el número de variables o dimensiones crece, provocando un aumento exponencial del espacio de búsqueda y dificultando la Exploración completa y eficiente, lo que hace inviables los métodos exhaustivos tradicionales.
x^*	Vector de coordenadas que minimiza o maximiza la función objetivo
$\mathcal{D}$	Conjunto factible de soluciones
$f(x)$	Función objetivo que se quiere minimizar o maximizar

Capítulo 2

Objetivos de la Investigación

Los objetivos de este trabajo se plantean en respuesta a los principales retos identificados en la optimización moderna, los cuales incluyen la alta dimensionalidad y complejidad de los espacios de búsqueda, la presencia de múltiples óptimos locales, la gestión de restricciones complejas y dinámicas, y las limitaciones en la evaluación de funciones objetivo costosas. Además, se considera la necesidad de algoritmos capaces de escalar con el tamaño del problema, adaptarse a cambios en el entorno y mantener un balance eficiente entre Exploración y Explotación de soluciones, asegurando robustez y calidad en los resultados.

En este contexto, los objetivos definidos buscan: (i) desarrollar y validar una heurística avanzada que aborde estos retos en optimización, (ii) demostrar su efectividad en problemas de optimización complejos mediante casos de estudios o aplicaciones y benchmarks, e (iii) integrar la metodología propuesta en aplicaciones prácticas, garantizando soluciones computacionalmente eficientes y adaptables a sistemas multidisciplinarios y de alta complejidad.

Esto permite al lector comprender cómo cada objetivo se vincula directamente con los retos de la optimización moderna, justificando la pertinencia y alcance del trabajo presentado.

2. OBJETIVOS DE LA INVESTIGACIÓN

2.1. Objetivo General

En consonancia con el objetivo general de esta investigación doctoral, el siguiente objetivo guía el desarrollo de un marco heurístico destinado a abordar problemas combinatorios de alta dimensión.

El objetivo principal de esta investigación es desarrollar una metodología heurística eficiente para la resolución de problemas complejos de optimización combinatoria en espacios multidimensionales. Este enfoque se basa en un algoritmo de búsqueda que combina técnicas de muestreo aleatorio y agrupamiento o *clustering*, con el fin de explorar de manera más eficaz grandes espacios de soluciones y encontrar configuraciones óptimas o cercanas a las óptimas. La combinación de estas técnicas permite mejorar la efectividad en el proceso de búsqueda al reducir el espacio de soluciones a explorar y facilitar la identificación de soluciones prometedoras. A través de esta metodología, se pretende ofrecer una alternativa más efectiva y flexible frente a los métodos tradicionales, como los algoritmos exactos, al ser capaz de abordar problemas de optimización de alta complejidad en un tiempo computacional razonable. Además, se busca aportar un marco robusto que pueda aplicarse a una amplia gama de problemas reales, desde la asignación de recursos hasta la planificación de tareas en entornos dinámicos y de alto volumen de datos.

2.2. Hipótesis de la investigación

Se plantea que un algoritmo basado en la Exploración Montecarlo combinada con técnicas de *clustering* puede reducir el espacio de soluciones y encontrar configuraciones cercanas al óptimo de manera más eficiente que los métodos tradicionales de búsqueda o los algoritmos heurísticos no combinados.

Esta combinación permitiría equilibrar la Exploración global y la Explotación local en entornos de alta dimensionalidad, adaptándose a distintos tipos de funciones objetivo, incluso bajo condiciones de incertidumbre o espacios de búsqueda irregulares. De este modo, la metodología propuesta aspira a ofrecer un marco

2. OBJETIVOS DE LA INVESTIGACIÓN

flexible, interpretable y de alto rendimiento para la resolución de problemas de optimización combinatoria en entornos reales y simulados.

2.2.1. Objetivos Específicos

Los objetivos específicos de esta investigación se detallan a continuación, centrándose en el desarrollo de nuestro enfoque para tareas complejas de optimización.

- Diseño y desarrollo del Algoritmo de Búsqueda: *Montecarlo Clustering Search Algorithm* (MCSA). Este algoritmo tiene como objetivo combinar técnicas de muestreo aleatorio a través de la simulación de Montecarlo con métodos de *clustering*, con el fin de mejorar la eficiencia en la búsqueda de soluciones óptimas o cercanas a la óptima en problemas de optimización combinatoria. El proceso se enfoca en la Exploración global del espacio de soluciones, seguido de una refinación progresiva mediante el agrupamiento de soluciones similares, lo que permite reducir el espacio de búsqueda y acelerar la convergencia hacia soluciones de alta calidad.
- Análisis, selección y adaptación de técnicas de clustering adecuadas para el modelo propuesto. Este proceso implica evaluar diversas metodologías de agrupamiento que se ajusten a las características específicas del problema de optimización, con el fin de identificar las más efectivas para mejorar la eficiencia y precisión del algoritmo de búsqueda. La adaptación de estas técnicas permitirá optimizar la segmentación del espacio de soluciones, favoreciendo una Exploración más estructurada y dirigida hacia las mejores configuraciones posibles, lo que a su vez facilitará la obtención de soluciones óptimas o cercanas a la óptima en un menor tiempo computacional.
- Definición de los criterios de convergencia apropiados para el algoritmo de búsqueda. Este proceso consiste en establecer las condiciones bajo las cuales el algoritmo considerará que ha alcanzado una solución satisfactoria o que ha explorado suficientemente el espacio de soluciones. Los criterios de convergencia son fundamentales para garantizar que el algoritmo detenga

2. OBJETIVOS DE LA INVESTIGACIÓN

su ejecución de manera eficiente, evitando tanto la sobre-Explotación del espacio de búsqueda como el cálculo innecesario de soluciones que no aportan mejoras significativas. Estos criterios se definen en función de factores como la calidad de las soluciones obtenidas, la robustez de los resultados a lo largo de las iteraciones y los recursos computacionales requeridos para alcanzar un nivel aceptable de solución.

- Validación del método propuesto utilizando funciones matemáticas de referencia, con un enfoque en la robustez y el rendimiento. Este proceso implica evaluar el desempeño del algoritmo frente a un conjunto de funciones estándar utilizadas comúnmente para medir la eficacia de los métodos de optimización. La validación se centra en determinar la capacidad del algoritmo para obtener soluciones de alta calidad de manera consistente, así como en analizar su comportamiento bajo diferentes condiciones, tales como la variabilidad en los parámetros del modelo y la complejidad del espacio de búsqueda. Además, se presta especial atención a la comparación de los resultados obtenidos con los métodos tradicionales, evaluando tanto la precisión de las soluciones como la eficiencia computacional del algoritmo propuesto.
- Aplicación de la metodología a problemas clásicos de optimización combinatoria. Este paso consiste en implementar el algoritmo desarrollado para resolver problemas tradicionales de optimización combinatoria, siendo el *Binary Knapsack Problem* - KP01 uno de los casos representativos. La aplicación sobre problemas de naturaleza similar permite evaluar la efectividad de la metodología en contextos prácticos y comparar su desempeño con aproximaciones o enfoques convencionales, como los métodos exactos o heurísticos ya establecidos. A través de estos casos, se busca no solo validar la capacidad del algoritmo para encontrar soluciones óptimas o cercanas a la óptima, sino también analizar su capacidad para manejar la complejidad y la escala de los problemas combinatorios, mostrando su potencial en términos de eficiencia y adaptabilidad.
- El Problema de la Mochila Binaria (KP01): Este problema es uno de los más

2. OBJETIVOS DE LA INVESTIGACIÓN

conocidos en la optimización combinatoria, donde el objetivo es seleccionar un subconjunto de ítems de tal manera que se maximice el valor total sin exceder una capacidad de peso o volumen. Este caso se resolverá aplicando la metodología propuesta para demostrar su eficacia en un escenario clásico de optimización combinatoria.

- Extensión a un marco multiobjetivo (*Multi-Objective Optimization, MOO*) para manejar instancias de mayor dimensión, aplicamos la metodología sobre el Problema de la Mochila Multiobjetivo (MOKP). Esta extensión consiste en adaptar la metodología propuesta para abordar problemas en los que se deben optimizar múltiples objetivos simultáneamente. En lugar de centrarse únicamente en la maximización de un solo objetivo (como el valor total de los ítems seleccionados), en el MOKP se deben considerar varios factores, como el valor, el peso, el volumen u otras restricciones que puedan aplicarse. Este enfoque más complejo implica la necesidad de generar soluciones que logren un equilibrio entre estos diferentes objetivos, lo que generalmente conduce a un conjunto de soluciones no dominadas (frente a otras alternativas). La extensión al marco multiobjetivo requiere técnicas más sofisticadas de Exploración y selección, ya que no se trata de optimizar un único objetivo, sino de encontrar compromisos adecuados entre varios objetivos en conflicto. Esta metodología permite ponderar distintos objetivos según su relevancia y, en consecuencia, elegir aquellas soluciones que mejor compensen las necesidades del problema. Así, la metodología resulta más robusta y adaptable a una amplia variedad de escenarios reales.
- Aplicación a un problema real de optimización: Este paso implica la implementación de la metodología desarrollada para resolver un problema de optimización en un contexto real. La aplicación práctica permite evaluar la viabilidad y efectividad de la metodología en condiciones no ideales, con datos y restricciones que reflejan situaciones del mundo real. Al abordar un problema real, se busca no solo verificar la capacidad del algoritmo para encontrar soluciones óptimas o cercanas a la óptima, sino también evaluar su rendimiento frente a incertidumbres, variabilidad y complejidades

2. OBJETIVOS DE LA INVESTIGACIÓN

inherentes a los entornos prácticos. La aplicación a un problema concreto puede incluir áreas como la gestión de recursos, la logística, la planificación de la producción o la distribución de tareas, proporcionando una prueba de su aplicabilidad en escenarios reales y demostrando su potencial para optimizar procesos en situaciones complejas.

- Encontrar una configuración del personal sanitario en un Departamento de Emergencias (ED, por sus siglas en inglés) de un hospital: Este objetivo se enfoca en aplicar la metodología de optimización desarrollada para resolver un problema práctico y altamente relevante: la asignación óptima de recursos humanos en un Servicio de Urgencias Hospitalarias (o ED). En un ED, la configuración del personal es crítica para garantizar una respuesta rápida y eficaz ante situaciones de alta demanda, como las que se presentan en emergencias médicas. El reto consiste en encontrar la distribución óptima de médicos, enfermeras y otros profesionales de la salud, asegurando que cada puesto esté cubierto de manera eficiente sin sobrecargar a los miembros del personal. Al abordar este problema con la metodología de optimización, se busca no solo mejorar la eficiencia operativa, sino también reducir los costos asociados a la gestión del personal, mejorar la calidad de la atención, medida a través de una serie de indicadores, y prevenir la saturación del equipo médico, lo cual es una garantía de funcionamiento adecuado en situaciones de alta presión.
- Integración de nuestra propuesta metodológica con un modelo de simulación basado en agentes (ABSM): El *EDSimulator*. Este objetivo se refiere a la incorporación de la metodología de búsqueda que hemos desarrollado, el MCSA, dentro de un entorno de simulación de tipo ABM. Este tipo de modelo de simulación permite modelar y simular el comportamiento de individuos o *agentes* en un sistema complejo, lo que es particularmente útil en escenarios dinámicos como los de un ED, donde el flujo de pacientes, el tiempo de respuesta, la disponibilidad de recursos y la carga de trabajo pueden variar al corto plazo. La integración del MCSA con el ABM permitirá buscar una configuración óptima del personal. Además, hará posible simular

2. OBJETIVOS DE LA INVESTIGACIÓN

su comportamiento en diferentes condiciones y detectar cuellos de botella o ineficiencias. La combinación de ambos modelos permitirá experimentar con distintas configuraciones y escenarios sin necesidad de implementar cambios en un entorno real, lo que proporciona una herramienta valiosa para la toma de decisiones informada y la mejora continua en el diseño y gestión de los recursos humanos en entornos hospitalarios.

- Buscar la configuración del personal sanitario que minimice el Indicador clave de rendimiento o *Key Performance Indicator (KPI)* que mide el tiempo de estancia o *Length of Stay (LoS)* de un paciente en el ED, desde su ingreso hasta su alta. El LoS es una métrica crítica de calidad para los servicios de urgencias. En este sentido, un mayor LoS generalmente refleja ineficiencias, congestión en el sistema y un menor nivel de calidad en la atención. Por tanto, la metodología se aplicará para optimizar la asignación del personal sanitario en función del LoS. Al optimizar la configuración del personal sanitario, se busca reducir la métrica LoS sin comprometer la calidad del servicio. La asignación adecuada de personal sanitario y de apoyo, considerando su ocupación y las especialidades o categorías requeridas, puede agilizar los procesos dentro del ED y conseguir reducir el LoS. El objetivo, entonces, es encontrar una configuración del personal sanitario que minimice el LoS, lo cual contribuye a la eficiencia operativa y a la satisfacción del paciente.

Glosario del Capítulo

A continuación, se presenta el glosario del Capítulo 2.1, que incluye los términos y símbolos más relevantes utilizados en esta sección para facilitar la comprensión de la terminología empleada y servir de referencia rápida.

Tabla 2.1: Glosario del Capítulo

Término	Definición
Benchmark	Conjunto de funciones matemáticas de referencia utilizadas para evaluar y comparar el desempeño de algoritmos de optimización.
<i>Clustering</i>	Técnica de agrupamiento de datos en subconjuntos según similitud.
Optimización combinatoria	Rama de la optimización que busca la mejor solución de un conjunto discreto de posibilidades, típicamente bajo restricciones.
Espacio multidimensional	Conjunto de todas las posibles soluciones de un problema de optimización representadas como vectores en un espacio de n -dimensiones.
Exploración y Explotación	Estrategias complementarias en heurísticas: la Exploración busca nuevas regiones del espacio de búsqueda, y la Explotación refina soluciones en regiones prometedoras.
MCSA	<i>Montecarlo Clustering Search Algorithm</i> : heurística basada en simulaciones Montecarlo utilizada para explorar soluciones en problemas complejos.

Continúa en la siguiente página

2. OBJETIVOS DE LA INVESTIGACIÓN

Glosario del Capítulo (continuación)

Término	Definición
ABM	<i>Agent-Based Modeling</i> (Simulación Basada en Agentes): metodología de modelado que simula interacciones individuales para estudiar el comportamiento colectivo del sistema.
LoS	<i>Length of Stay</i> : tiempo medio de estancia de pacientes en un hospital o unidad, utilizado como indicador de desempeño (KPI).
ED	<i>Emergency Department</i> (Departamento de Emergencias, SUH): unidad hospitalaria encargada de la atención inmediata de pacientes con condiciones críticas o urgentes. En esta tesis, se usará la abreviatura ED para referirse a estas unidades.
KPI	<i>Key Performance Indicator</i> : indicador clave de desempeño utilizado para medir, evaluar y comparar el rendimiento de procesos, configuraciones o sistemas. En esta tesis, se emplean KPI para cuantificar el desempeño de configuraciones en el <i>Emergency Department</i> (ED), como por ejemplo el <i>Length of Stay</i> (LoS) de los pacientes.
KP01	Problema clásico de optimización conocido como <i>Binary Knapsack Problem</i> - <i>KP01</i> (Problema de la Mochila Binaria), donde se seleccionan elementos con valor y peso limitado para maximizar el beneficio total sin exceder la capacidad de la mochila.

Continúa en la siguiente página

2. OBJETIVOS DE LA INVESTIGACIÓN

Glosario del Capítulo (continuación)

Término	Definición
MOKP	<i>Multi-Objective Knapsack Problem</i> : variante multiobjetivo del problema clásico de la mochila, considerando múltiples objetivos de optimización simultáneamente.
SUH	Sistema de Urgencias Hospitalarias: conjunto de recursos humanos, físicos y procesos organizativos destinados a la atención de emergencias en un hospital.

2. OBJETIVOS DE LA INVESTIGACIÓN

Capítulo 3

Marco Teórico

Este capítulo tiene como objetivo proporcionar el marco conceptual y los antecedentes necesarios para comprender los retos de la optimización moderna y las estrategias empleadas para abordarlos. Se analizan los principales problemas que surgen en espacios de alta dimensionalidad, la presencia de múltiples óptimos locales, la gestión de restricciones complejas y los costos asociados a la evaluación de funciones objetivo. Asimismo, se discute la importancia de mantener un equilibrio adecuado entre Exploración y Explotación en los algoritmos heurísticos.

A lo largo del capítulo se presentan los conceptos clave que sustentan la metodología propuesta, incluyendo la integración de técnicas de muestreo aleatorio con métodos de agrupamiento, como base para diseñar algoritmos heurísticos eficientes en espacios de búsqueda combinatorios. Finalmente, se muestra cómo estos principios se aplican en el diseño y validación del algoritmo desarrollado en esta investigación, con aplicaciones en áreas como logística, transporte y gestión de recursos en salud.

3.1. Contexto

El desarrollo de algoritmos de optimización no solo resulta fundamental para resolver problemas específicos en diversas áreas, sino también para mejorar la eficiencia de los procesos científicos y acelerar el análisis de datos. En este sentido, la creación de nuevos algoritmos que se adapten a problemas particulares o que ofrezcan mejores garantías de eficiencia constituye una tarea de gran relevancia. El proceso de desarrollo de algoritmos implica comprender las propiedades del problema, seleccionar técnicas adecuadas y evaluar el rendimiento del algoritmo mediante estrategias como la validación cruzada, la comparación con métodos existentes y el ajuste de parámetros del modelo.

Los problemas de optimización combinatoria en espacios multidimensionales presentan una complejidad inherente debido a la enorme cantidad de soluciones posibles que deben evaluarse para identificar la mejor o más eficiente [85, 58]. Dichos problemas aparecen en aplicaciones de gran impacto, como la planificación de recursos, la asignación de tareas, la logística, la ingeniería de diseño y la inteligencia artificial. La naturaleza multidimensional incrementa de forma exponencial el espacio de búsqueda, lo que hace que los métodos de búsqueda exhaustiva resulten ineficaces y demasiado costosos en términos de tiempo y recursos computacionales.

Dado que la mayoría de estos problemas carecen de una solución exacta eficiente, surge la necesidad de aproximaciones. En este contexto, los algoritmos heurísticos se consolidan como una alternativa viable, pues permiten obtener soluciones de calidad sin explorar de forma exhaustiva todas las configuraciones posibles. El reto principal consiste en diseñar metodologías heurísticas capaces de hallar soluciones cercanas al óptimo en espacios multidimensionales, de manera efectiva y en tiempos razonables.

La aproximación desarrollada en esta investigación consiste en un algoritmo de búsqueda heurístico que combina técnicas de muestreo aleatorio con métodos de agrupamiento. El muestreo aleatorio permite explorar de forma amplia el espacio de búsqueda, mientras que el agrupamiento facilita la identificación de patrones

3. MARCO TEÓRICO

o regiones con soluciones prometedoras. La integración de estas técnicas optimiza la búsqueda en espacios complejos, mejorando la eficiencia frente a los métodos tradicionales.

En este marco, la metodología propuesta se materializa en el algoritmo desarrollado en esta tesis, orientado a resolver problemas de optimización complejos con aplicaciones reales. Su contribución se proyecta en ámbitos como la logística y el transporte, la planificación de rutas, la gestión de recursos en salud, la asignación de tareas y la planificación de horarios. En todos estos escenarios, el algoritmo busca no solo generar soluciones eficientes y rentables, sino también apoyar la toma de decisiones en contextos donde el número de configuraciones posibles es extremadamente elevado.

3.2. Motivación de la propuesta de Optimización

El desarrollo de este trabajo de tesis parte de la necesidad de abordar problemas de optimización caracterizados por su complejidad, alta dimensionalidad y limitada disponibilidad de información estructurada. En estos contextos, los métodos tradicionales, determinísticos, exactos o basados en gradientes, resultan insuficientes para capturar la naturaleza dinámica y no lineal de los sistemas estudiados [91].

Ante esta limitación, se plantea el diseño de una metodología de tipo heurística que combine flexibilidad y capacidad de adaptación, permitiendo representar el proceso de búsqueda como un sistema parametrizable. Esta parametrización no sólo actúa como un mecanismo de control del comportamiento del algoritmo, sino que también posibilita reinterpretar el problema de optimización desde una perspectiva más abierta, donde la Exploración y la Explotación del espacio de soluciones se equilibran manteniendo un esquema que se basa en la integración de principios estocásticos (Montecarlo), con mecanismos de ajuste basados en parámetros de búsqueda.

En conjunto, la propuesta busca ofrecer un marco metodológico que pueda

3. MARCO TEÓRICO

adaptarse a distintos tipos de problemas de optimización, proporcionando soluciones robustas incluso en escenarios donde la información es incierta o incompleta.

3.2.1. Definición Formal y Componentes Clave de la Optimización

En las últimas décadas, los problemas de optimización han evolucionado hacia estructuras cada vez más complejas y extensas, lo que dificulta considerablemente su resolución mediante métodos exactos como [72, 29]. Esta creciente dificultad se debe, en gran medida, a la multidimensionalidad y no linealidad inherentes a muchos problemas, especialmente en contextos científicos o de ingeniería. Como consecuencia, los métodos exactos, si bien útiles en escenarios controlados y de baja complejidad, presentan limitaciones significativas al abordar problemas de alta dimensión o con múltiples restricciones. Esta situación ha impulsado el desarrollo y adopción de algoritmos aproximados, en particular los basados en heurísticas y metaheurísticas, como alternativas eficaces y escalables. Las metaheurísticas pueden definirse como algoritmos de optimización de propósito general, independientes del problema específico, aplicables a una amplia variedad de problemas e instancias. El término *meta* hace referencia a la metodología de alto nivel utilizada para guiar la estrategia heurística subyacente. Estas comparten características comunes que permiten un equilibrio entre Exploración y Explotación, adaptabilidad y robustez frente a problemas complejos [17].

La optimización puede entenderse entonces como un proceso impulsado por el sentido común: elegir la mejor alternativa posible dentro de un conjunto de opciones, considerando ciertas restricciones. Aunque esta idea es intuitiva y forma parte de la toma de decisiones cotidiana, como minimizar costos, maximizar ganancias o equilibrar recursos, su formulación matemática no siempre es sencilla y requiere una descripción más estructurada.

Para establecer una base conceptual, se define el conjunto factible de un problema de optimización como $\mathcal{D} \subseteq \mathbb{R}^n$, que contiene todas las soluciones que satisfacen

3. MARCO TEÓRICO

las restricciones del problema. Un problema de optimización puede formularse clásicamente como:

$$\min_{x \in \mathcal{D}} f(x), \quad (3.1)$$

donde $f(x)$ es la función objetivo a minimizar, y

$$\mathcal{D} = \left\{ x \in \mathbb{R}^n : \begin{array}{ll} g_i(x) \leq 0, & i = 1, \dots, m, \\ h_j(x) = 0, & j = 1, \dots, p \end{array} \right\}. \quad (3.2)$$

Para problemas de maximización, la formulación es análoga: $\max_{x \in \mathcal{D}} f(x)$. Aquí, $g_i(x)$ y $h_j(x)$ representan las restricciones de desigualdad y de igualdad, respectivamente.

Esta representación proporciona un marco formal claro para estudiar problemas de optimización, ya que captura de manera explícita las limitaciones inherentes a la situación analizada. No obstante, uno de los principales retos consiste en traducir problemas del mundo real a esta formulación, especialmente cuando involucran un gran número de variables, múltiples restricciones o relaciones no lineales. Es precisamente en estos escenarios donde las técnicas de optimización aproximada adquieren relevancia.

En estos casos, las técnicas de optimización aproximada, como heurísticas y metaheurísticas, resultan particularmente útiles para obtener soluciones de buena calidad con un coste computacional razonable.

3.2.2. Elementos de la Optimización

El estudio de la optimización matemática comienza con una adecuada formulación del problema, lo cual implica traducir una situación real en términos bien definidos. Esta formulación es especialmente importante en problemas complejos de naturaleza combinatoria y multidimensional, donde el espacio de búsqueda crece exponencialmente y se vuelve inviable aplicar métodos exactos convencionales.

3. MARCO TEÓRICO

Un problema de optimización se compone, en términos generales, según [18], de los siguientes elementos clave:

- **Descripción del modelo:** Representación abstracta del sistema o situación real, incluyendo las variables relevantes, que en este trabajo corresponden principalmente a variables discretas en espacios de alta dimensionalidad.
- **Función objetivo:** Expresión matemática que cuantifica la magnitud que se busca optimizar (minimizar o maximizar), como un costo, un tiempo de operación o un nivel de eficiencia. En términos generales, se representa como

$$f(x) : \mathcal{D} \rightarrow \mathbb{R},$$

donde $x \in \mathcal{D}$ es el vector de variables de decisión y $\mathcal{D}$ el dominio factible. En aplicaciones prácticas, los valores de $f(x)$ se calculan a partir de resultados obtenidos de simulaciones u otras evaluaciones, y se expresan mediante métricas de rendimiento o indicadores finales que reflejan la calidad de la solución. Es importante aclarar que en el caso de los datos intermedios de la simulación, éstos son utilizados para computar $f(x)$, pero no forman parte de la función objetivo directamente.

- **Restricciones:** Condiciones que deben satisfacer las variables del modelo, derivadas de limitaciones prácticas, físicas, técnicas o de negocio. Estas restricciones determinan la viabilidad de las soluciones candidatas.
- **Dominio factible:** Conjunto de todas las posibles soluciones que satisfacen las restricciones. En este trabajo, este dominio se caracteriza por ser discreto y de alta dimensionalidad, lo que impone dificultades adicionales en la búsqueda de soluciones eficientes.
- **Método de resolución:** Técnica computacional empleada para encontrar una o más soluciones cuasi-óptimas dentro del dominio factible. Dado que los problemas abordados en este trabajo presentan alta complejidad combinatoria, se propone un enfoque heurístico basado en técnicas de muestreo aleatorio, agrupamiento y evaluación iterativa, diseñado para

3. MARCO TEÓRICO

explorar eficazmente espacios grandes y no estructurados.

Una correcta formulación de estos elementos no solo permite representar adecuadamente el problema, sino también diseñar estrategias de resolución que se adapten a sus características. En esta tesis, se parte de esta base para construir una metodología heurística orientada a la resolución de problemas de optimización discretos y multidimensionales para obtener soluciones eficientes. En las siguientes secciones se detallará la forma general de los problemas abordados y se ilustrarán sus componentes con ejemplos concretos.

3.3. Métodos de Resolución en Problemas de Optimización

Una vez formulado un problema de optimización, el siguiente paso consiste en seleccionar un método adecuado para resolverlo. De manera general, los métodos de resolución se pueden clasificar en dos grandes grupos: métodos exactos y **métodos aproximados o heurísticos/metaheurísticos** [65]. Esta clasificación refleja la diferencia entre técnicas que garantizan óptimos globales y aquellas que ofrecen soluciones aproximadas en tiempos razonables.

Los métodos exactos buscan encontrar la mejor solución posible (óptimo global) con garantías teóricas de optimalidad. Entre los más conocidos se encuentran la programación lineal, programación entera, programación no lineal, programación cuadrática, programación dinámica, así como algoritmos de ramificación y acotación (*branch and bound*) y planos de corte. Son especialmente útiles para problemas bien definidos y de tamaño moderado, pero su aplicabilidad se ve limitada en problemas de gran escala o con espacios de búsqueda combinatorios muy grandes, debido a la complejidad computacional. En estos casos, el coste computacional asociado al proceso de resolución crece de forma exponencial con el número de variables o restricciones, una manifestación clara de lo que se conoce como complejidad combinatoria o incluso la *maldición de la dimensionalidad* [14].

En problemas reales de alta complejidad, especialmente aquellos *NP-hard*, los

3. MARCO TEÓRICO

métodos exactos resultan imprácticos. En tales casos, se emplean métodos aproximados como **heurísticas** y **metaheurísticas**. Las heurísticas aplican reglas específicas o conocimiento experto del dominio para generar soluciones viables en tiempos razonables. Las metaheurísticas son marcos más generales que combinan Exploración global del espacio de búsqueda con Explotación local para intensificar la búsqueda en regiones prometedoras. Algunos ejemplos son [33]:

- *Algoritmos Genéticos (GA)*: Inspirados en los principios de la evolución natural.
- *Recocido Simulado (Simulated Annealing)*: Basado en procesos físicos de enfriamiento gradual.
- *Optimización por Enjambre de Partículas (PSO)*: Modela el comportamiento colectivo de sistemas biológicos.
- *Algoritmos de Colonia de Hormigas (ACO)*: Se inspiran en el comportamiento de búsqueda de alimento de hormigas.
- *Búsqueda Tabú, Búsqueda por Entornos Variables*, entre otros.

Estas técnicas permiten obtener soluciones eficientes en entornos de alta dimensionalidad, incertidumbre y restricciones complejas, constituyendo la base metodológica adoptada en esta tesis.

3.4. Heurísticas: métodos iterativos y métodos exploratorios

Los métodos de optimización pueden clasificarse de diversas formas según aspectos como la naturaleza del problema, la disponibilidad de derivadas, la convexidad del espacio de búsqueda o la estrategia empleada para explorar las posibles soluciones. Una distinción especialmente relevante para este trabajo distingue los métodos según el tipo de búsqueda realizado: los **métodos iterativos** y los **métodos exploratorios o de búsqueda global** [89].

3. MARCO TEÓRICO

Los *métodos iterativos* realizan un proceso secuencial de mejora a partir de una solución inicial, que se va refinando en cada paso (o iteración) mediante información determinista, como el gradiente, la dirección de búsqueda o una alguna regla de actualización predefinida. Son especialmente útiles para resolver problemas bien estructurados, diferenciables y con buena información inicial. Entre los ejemplos más representativos se encuentran el método del gradiente, el algoritmo de Newton-Raphson y el método de gradiente conjugado [73].

En contraste, los *métodos exploratorios o de búsqueda global* buscan recorrer de forma más amplia el espacio de soluciones, sin depender de condiciones iniciales favorables ni de la continuidad del problema. Estos métodos son especialmente adecuados para problemas complejos, no lineales o multimodales, donde los enfoques deterministas tienden a quedar atrapados en óptimos locales. Dentro de esta categoría se incluyen los métodos heurísticos y metaheurísticos, que se apoyan en estrategias flexibles inspiradas en fenómenos naturales o procesos probabilísticos, como la evolución biológica, el comportamiento colectivo o la dinámica de sistemas físicos [35].

El enfoque adoptado en esta tesis se enmarca dentro de esta última familia de métodos, explorando técnicas heurísticas basadas en búsqueda estocástica, como el método de Montecarlo, que permite escapar de óptimos locales y favorece una Exploración más robusta del espacio de soluciones.

3.4.1. Heurísticas y Metaheurísticas en la Resolución de Problemas Complejos

En problemas de optimización reales, especialmente aquellos que presentan alta complejidad estructural o dimensional, los métodos exactos suelen volverse ineficientes o directamente inviables desde el punto de vista computacional. En este contexto, las heurísticas y metaheurísticas se consolidan como herramientas fundamentales para obtener soluciones viables y de calidad en tiempos razonables.

Estas técnicas permiten explorar espacios de solución multidimensionales y, en muchos casos, imposibles de recorrer completamente mediante métodos

3. MARCO TEÓRICO

deterministas. Su flexibilidad radica en la capacidad de incorporar conocimiento experto, aproximaciones basadas en estructuras del problema y estrategias adaptativas que equilibran la Exploración global y la Explotación local.

La evolución reciente en la investigación en optimización ha impulsado un cambio hacia enfoques que no se limitan al rendimiento cuantitativo, sino que promueven métodos más interpretables, adaptativos y centrados en el diseño algorítmico. Esta perspectiva reconoce el valor de las heurísticas como herramientas no solo para encontrar buenas soluciones, sino también para comprender mejor la estructura del problema y guiar procesos de búsqueda más informados.

Con esta base teórica establecida, en la siguiente sección se presentan los trabajos relacionados y antecedentes que fundamentan la propuesta metodológica de este estudio, destacando las principales influencias, aportes previos y vacíos que motivan el desarrollo del método planteado.

3.5. Fundamentación Teórica y Diseño del Método Propuesto

Los distintos tipos de problemas de optimización presentan desafíos específicos que pueden incrementar significativamente su complejidad. Incluso la optimización monoobjetivo (*Single-Objective Optimization, SOO*), en apariencia sencilla, puede derivar en escenarios con múltiples óptimos locales o restricciones implícitas que dificultan la búsqueda de soluciones robustas. Estas características obligan a emplear algoritmos capaces de mantener un equilibrio adecuado entre Exploración y Explotación para evitar el estancamiento prematuro.

Uno de los factores que más condiciona la dificultad de estos problemas es la dimensionalidad del espacio de búsqueda. A medida que aumenta el número de variables, también lo hace el volumen del dominio definido por sus cotas, lo que genera espacios continuos o discretos de elevada complejidad. Este fenómeno limita la eficacia de los métodos deterministas y afecta directamente al coste computacional. Desde el punto de vista estadístico, esta situación se relaciona con

3. MARCO TEÓRICO

el *efecto Hughes*, que describe cómo el aumento de dimensiones puede mejorar inicialmente el rendimiento, pero termina deteriorándolo si no se dispone de suficientes datos [46]. En consecuencia, tanto en optimización como en modelado, la alta dimensionalidad compromete simultáneamente la precisión, el tiempo de cómputo y la viabilidad de los técnicas tradicionales.

Ante estas limitaciones, las metaheurísticas se posicionan como alternativas especialmente adecuadas para abordar problemas complejos en espacios amplios y no estructurados. Su capacidad para combinar procesos de Exploración global con mecanismos de Explotación local les permite adaptarse a las características del problema sin requerir supuestos estrictos sobre la estructura del espacio de soluciones.

Sobre este marco teórico, el presente trabajo propone un método heurístico que incorpora mecanismos de control parametrizable y componentes estocásticos inspirados en Montecarlo, con el objetivo de regular dinámicamente la intensidad de Exploración y Explotación. Asimismo, se integra el uso de técnicas de *clustering* para identificar regiones relevantes del espacio de búsqueda, facilitando una orientación más eficiente del proceso iterativo. En conjunto, esta fundamentación teórica respalda el diseño de un método flexible, escalable y adaptativo, orientado a resolver problemas complejos en entornos caracterizados por alta dimensionalidad e incertidumbre.

3.6. Trabajo Relacionado

Esta sección presenta los antecedentes y fundamentos del algoritmo de búsqueda Montecarlo Clustering Search Algorithm (MCSA), una propuesta desarrollada y refinada en el marco de este trabajo de tesis. El punto de partida se encuentra en la tesis doctoral de Eduardo Cabrera [22] (2013), quien, desde el grupo de investigación HPC4EAS, propuso una estrategia heurística para abordar un problema de asignación de personal en un Servicio de Urgencias Hospitalarias (*Emergency Department* ó ED), caracterizado por su alta complejidad combinatoria y su naturaleza dinámica.

3. MARCO TEÓRICO

A partir de esa base, se plantea aquí un proceso de reformulación y extensión metodológica del algoritmo, con el objetivo de ampliar su aplicabilidad a problemas más complejos y multidimensionales, incluyendo contextos de optimización continua y multiobjetivo. En este sentido, se describe la evolución conceptual y técnica del método, destacando tanto las mejoras introducidas como las limitaciones detectadas que motivaron el rediseño.

El propósito es ofrecer un marco de comprensión sobre las decisiones metodológicas adoptadas en el desarrollo del MCSA, contextualizando su origen, sus componentes principales y la progresiva integración de nuevas estrategias de Exploración estocástica, reducción del dominio de búsqueda y análisis de convergencia. Finalmente, se describen los primeros pasos hacia la formulación de una nueva metodología, apoyada en la validación y el análisis diagnóstico del modelo base a partir de funciones *benchmark*, mediante las cuales ha sido posible evaluar su comportamiento, identificar sus limitaciones y señalar posibles líneas de mejora ante entornos de mayor complejidad.

3.7. Fundamentos del Algoritmo base

El algoritmo de búsqueda desarrollado en esta tesis toma como punto de partida la propuesta presentada por E. Cabrera, originalmente desarrollada para resolver un problema combinatorio complejo. En aquel trabajo, el objetivo era optimizar la distribución del personal sanitario, técnico y de enfermería de un ED bajo múltiples restricciones y condiciones de operación. A partir de esta base metodológica, la presente investigación extiende y adapta la perspectiva para abordar problemas de mayor diversidad y complejidad.

La propuesta original combinaba una Fase de Exploración global estocástica mediante la heurística de Montecarlo [69] con una fase de agrupamiento y refinamiento basada en el algoritmo K-Means [49, 63]. La primera fase generaba un conjunto de soluciones aleatorias distribuidas en el espacio de búsqueda, garantizando diversidad y cobertura; mientras que la segunda permitía identificar las regiones más prometedoras agrupando las soluciones en función de su

3. MARCO TEÓRICO

proximidad o similitud.

Una vez identificada la región con mayor densidad de soluciones viables, el algoritmo ejecutaba una búsqueda localizada dentro de ese cluster, con el objetivo de refinar los resultados y aproximarse a un óptimo de alta calidad. Este diseño híbrido permitió equilibrar la Exploración global y la Explotación local, reduciendo el coste computacional respecto a una búsqueda exhaustiva.

Estructura y Funcionamiento del Modelo de referencia

El modelo de referencia, en adelante algoritmo base y que hace referencia al modelo original propuesto por Cabrera, puede entenderse como un proceso iterativo que combina tres etapas principales:

1. **Exploración inicial (Montecarlo):** generación aleatoria de soluciones factibles en el espacio de búsqueda, garantizando una Exploración amplia y no sesgada.
2. **Agrupamiento (K-Means):** clasificación de las soluciones obtenidas en k grupos o regiones factibles, según su similitud en el espacio de búsqueda o en el valor de la función objetivo.
3. **Refinamiento local:** selección del clúster más prometedor y ejecución de una búsqueda intensiva en su entorno, con el fin de mejorar la calidad de las soluciones identificadas.

Esta estructura fue validada exhaustivamente por Cabrera en el contexto discreto y tridimensional del ED. En dicho entorno, el método demostró ser eficiente para encontrar configuraciones robustas de asignación, al combinar el muestreo aleatorio con un proceso de reducción progresiva del dominio de búsqueda.

La combinación de Montecarlo con K-Means en este algoritmo sigue una estrategia de Exploración y refinamiento. En primer lugar, la Exploración aleatoria permite cubrir un amplio rango del espacio de soluciones discretizado. Posteriormente, se usa K-Means para agrupar las soluciones generadas y concentrarse en regiones de

3. MARCO TEÓRICO

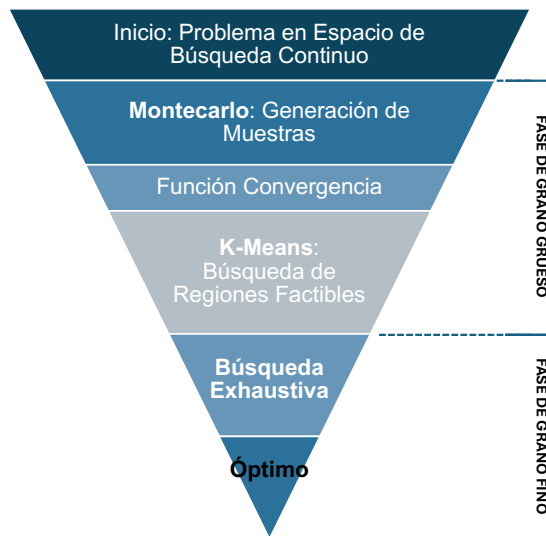


Figura 3.1: Modelo conceptual del algoritmo base. Un problema inicial, ubicado en un espacio continuo es discretizado. Aquí inicia el proceso de muestreo aleatorio y reducción de espacio, buscando Regiones Factibles donde finalmente se llevará a cabo una búsqueda exhaustiva localizada.

mayor concentración de soluciones prometedoras. Este enfoque híbrido permite al algoritmo adaptarse a la naturaleza estocástica del problema de optimización enfocado en el ED.

La Figura 3.1 muestra esquemáticamente esta interacción entre la Exploración estocástica y el refinamiento estructurado, donde la heurística Montecarlo proporciona diversidad global y K-Means concentra el análisis en zonas potencialmente óptimas.

3.7.1. Limitaciones del modelo de referencia

A pesar de su eficacia en el problema original, el algoritmo base presentaba ciertas limitaciones que restringían su aplicabilidad a contextos más generales de optimización:

- **Alcance dimensional limitado:** la validación original se realizó en un espacio de tres dimensiones. La cuarta variable, el valor medio del

3. MARCO TEÓRICO

índice LoS, se mostró únicamente mediante el color en el gráfico de dispersión 3D, empleándose como indicador visual para verificar que el ordenamiento obtenido era coherente (por ejemplo, mostrando gradientes de color ordenados). Sin embargo, esta codificación cromática no constituía una dimensión procesada algorítmicamente, por lo que el método no gestionaba explícitamente un espacio de cuatro dimensiones, lo que limitaba su generalización a problemas de mayor dimensionalidad.

- **Dependencia de un único mínimo global:** el problema del ED presentaba una única región factible dominante, por lo que el modelo no fue diseñado para gestionar escenarios con múltiples óptimos locales de similar valor, una característica común en funciones multimodales o no convexas.
- **Número fijo de clusters (k):** el método dependía de una elección fija de k , lo que resultaba adecuado en el contexto controlado del ED pero ineficiente para distribuciones más irregulares o heterogéneas.
- **Ausencia de mecanismos adaptativos:** la heurística carecía de un componente dinámico que ajustara la Exploración en función del comportamiento de las soluciones o de la dispersión entre clusters.

Estas restricciones se traducían en una falta de versatilidad del modelo ante problemas continuos, multidimensionales o con múltiples regiones factibles. En otras palabras, el algoritmo funcionaba correctamente en el contexto original debido a que la estructura del espacio de búsqueda era relativamente regular y bien delimitada; sin embargo, no había sido diseñado para manejar distribuciones con regiones dispersas o con varios óptimos de similar calidad.

Proceso de adaptación y reformulación

Con el fin de superar estas limitaciones y explorar el potencial del modelo en escenarios más complejos, el presente trabajo realizó una adaptación integral del algoritmo. En primer lugar, se reimplementó completamente el método en el entorno `Python` [79], lo que permitió integrar herramientas de análisis y

3. MARCO TEÓRICO

visualización avanzadas.

A continuación, se diseñó un conjunto de experimentos sobre funciones benchmark clásicas de la optimización continua, como por ejemplo *Ackley*, *Rastrigin*, *Sphere*, *Hölder Table*, entre otras [70]. Estas funciones presentan múltiples regiones factibles y distribuciones no convexas, lo que permitió analizar el comportamiento del modelo ante espacios de búsqueda con más de una región óptima o con mínimos locales de valor similar.

De este modo, la evaluación realizada no buscaba verificar la validez del algoritmo original, ya demostrada en su aplicación inicial, sino explorar su capacidad de generalización y adaptación ante topologías más complejas del espacio de soluciones. Los resultados obtenidos permitieron identificar las fortalezas y debilidades del modelo y sentaron las bases para el desarrollo de una nueva metodología, orientada a mejorar la capacidad de análisis y convergencia del MCSA frente a problemas multidimensionales.

El procedimiento propuesto se desarrolla en varias etapas. En primer lugar, se generan N soluciones iniciales mediante muestreo aleatorio. A continuación, cada subconjunto se somete a una función de aptitud definida por la media de los valores obtenidos en las iteraciones de Montecarlo en esta fase de muestreo. Esta función cumple el propósito de activar el criterio de parada, determinando hasta dónde continuar. Posteriormente, las soluciones se agrupan mediante K-Means para identificar subconjuntos de configuraciones similares. De cada grupo se selecciona la mejor solución, es decir, aquella con el valor más bajo según la función objetivo.

Para efectos de validación de nuestra propuesta, esto significa que se comprueba la calidad de la solución seleccionada mediante la fórmula del *benchmark*, verificando su grado de aproximación al óptimo, en línea con la estructura general de los algoritmos metaheurísticos [13], y se incorpora una iteración adicional de refinamiento para comprobar si la solución mejora. El resultado final corresponde a una única solución que, tras este proceso de generación, evaluación, agrupamiento y validación, representa la mejor aproximación al óptimo alcanzada por el procedimiento.

3.7.2. Evaluación del algoritmo base a través de *benchmarks*

La validación de algoritmos de optimización constituye un paso obligatorio para garantizar la solidez de los resultados y la fiabilidad de las conclusiones. En este contexto, el uso de funciones matemáticas de referencia (*benchmark functions*) se ha consolidado como una práctica estándar en la literatura científica [75]. Dichas funciones ofrecen un entorno controlado y bien definido que permite medir y comparar el desempeño de diferentes técnicas de optimización, identificando fortalezas y limitaciones en escenarios de diversa complejidad, como problemas multimodales, paisajes irregulares o de alta dimensionalidad.

El empleo de funciones *benchmark* facilita el análisis empírico de algoritmos heurísticos y metaheurísticos, ya que permite comparar su efectividad en términos de precisión, velocidad de convergencia y capacidad de escape de óptimos locales. Las funciones tienen características particulares que ponen a prueba distintas capacidades de los algoritmos y son capaces de modelar dificultades representativas de problemas reales de optimización global.

Además, desempeñan un papel clave en la evaluación de algoritmos de optimización. En la investigación, resulta imprescindible disponer de un conjunto estandarizado de problemas que permitan probar la eficacia de nuevas metodologías antes de su aplicación en casos reales. Estas funciones ofrecen un entorno controlado que posibilita medir el rendimiento de los algoritmos bajo condiciones equivalentes, facilitando así la comparación entre distintos enfoques y proporcionando información valiosa acerca de su efectividad y robustez.

El uso de funciones *benchmark* para la validación resulta importante por varias razones. En primer lugar, aseguran la reproducibilidad de los experimentos, ya que al tratarse de problemas ampliamente documentados, cualquier investigador puede replicar los resultados y validar las conclusiones. En segundo lugar, permiten establecer una comparación justa entre algoritmos, evitando sesgos que podrían surgir al seleccionar problemas ad hoc. Finalmente, constituyen un marco de fiabilidad, pues un algoritmo que logra un desempeño sólido en un amplio conjunto

3. MARCO TEÓRICO

de funciones con diferentes características puede considerarse más generalizable y aplicable a problemas reales. Además, es necesario evaluar la efectividad de los procedimientos de optimización estocástica, identificando sus fortalezas y limitaciones en distintos escenarios. En la literatura sobre funciones objetivo se aborda una amplia variedad de problemas de prueba. Para este trabajo, centramos nuestra evaluación especialmente en aquellos problemas cuyos parámetros son reales y cuyas funciones objetivo están matemáticamente bien definidas.

Entre las principales dificultades que presentan las funciones *benchmark* se incluyen:

- **Paisajes multimodales:** muchas funciones de prueba contienen múltiples óptimos locales que dificultan la convergencia hacia la solución global.
- **Alta dimensionalidad:** los algoritmos deben demostrar su capacidad de escalar de manera eficiente a medida que aumenta el número de variables o el tamaño del espacio de búsqueda.
- **Superficies con gradientes variables:** algunas funciones presentan regiones planas o altamente oscilatorias que afectan de forma significativa a los métodos basados en gradientes.

En consecuencia, los algoritmos de optimización deben ser evaluados sobre un conjunto diverso y representativo de funciones, con el fin de garantizar su generalización y aplicabilidad en escenarios complejos y heterogéneos. Esta práctica no solo incrementa la credibilidad de los resultados, sino que también permite identificar los contextos en los que un algoritmo puede ofrecer ventajas competitivas frente a otros existentes.

La comunidad científica ha consolidado marcos de referencia estandarizados, como las competencias organizadas por el *Congress on Evolutionary Computation* (CEC) [59] o las plataformas COCO/BBOB (*Black-Box Optimization Benchmarking*) [41], que ofrecen colecciones amplias y desafiantes de problemas. Estos entornos no solo permiten evaluar de manera sistemática el rendimiento de diferentes métodos, sino que también facilitan la reproducibilidad de los

3. MARCO TEÓRICO

experimentos. En el presente trabajo de tesis, se introducirá un conjunto representativo de estas funciones en el capítulo dedicado a la Evaluación Experimental, donde se detallará su implementación, métricas de rendimiento y el análisis comparativo de los resultados obtenidos.

El objetivo de esta sección es justificar la selección e ilustrar el papel de estas funciones en la evaluación del algoritmo base como un prototipo, el cual se apoya en técnicas de muestreo aleatorio y K-Means. La aplicación de este algoritmo en benchmarks seleccionadas permitió establecer un punto de partida y, lograr identificar las limitaciones topológicas de su componente de clustering. Se analizarán en detalle algunas funciones seleccionadas, resultando pertinente exponer sus formulaciones matemáticas y las dificultades que plantean para los algoritmos de búsqueda de tipo heurístico o metaheurístico.

Estas funciones seleccionadas se caracterizan por ofrecer paisajes de distinta complejidad: desde superficies convexas y unimodales, hasta escenarios altamente multimodales con numerosos mínimos locales. Su análisis permite evaluar la capacidad del algoritmo prototipo tanto para realizar una Exploración global efectiva como para garantizar una Explotación precisa en la proximidad de un óptimo. En la Tabla 3.1 se resumen las principales funciones utilizadas en esta primera parte del trabajo, indicando su formulación matemática, la ubicación del óptimo global, el valor alcanzado en dicho punto y una breve descripción de las características que las convierten en pruebas relevantes para la validación experimental.

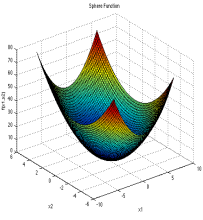
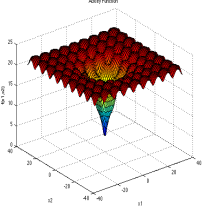
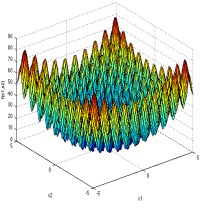
Para complementar la descripción matemática de las funciones *benchmark*, se presentan a continuación representaciones gráficas esquemáticas de cada función en la Tabla 3.2. Cada gráfico ilustra de manera cualitativa la topología del paisaje de la función, destacando características como la presencia de óptimos locales, globales y la forma general del espacio de búsqueda. Estas representaciones permiten al lector visualizar el comportamiento esperado de la función, facilitando la comprensión de los desafíos que plantean para los algoritmos de optimización.

3. MARCO TEÓRICO

Tabla 3.1: Funciones *benchmark* utilizadas en la literatura de optimización.

Función / Fórmula	Óptimo global	Valor
Sphere $f(x) = \sum_{i=1}^d x_i^2$	$x^* = (0, \dots, 0)$	$f(x^*) = 0$
Ackley $f(x) = -20 \exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^d x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^d \cos(2\pi x_i)\right) + 20 + e$	$x^* = (0, \dots, 0)$	$f(x^*) = 0$
Rastrigin $f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$	$x^* = (0, \dots, 0)$	$f(x^*) = 0$
Griewank $f(x) = 1 + \frac{1}{4000} \sum_{i=1}^d x_i^2 - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right)$	$x^* = (0, \dots, 0)$	$f(x^*) = 0$
Hölder Table $f(x, y) = - \sin(x) \cos(y) \exp(1 - \frac{\sqrt{x^2+y^2}}{\pi}) $	$(\pm 8,05502, \pm 9,66459)$	$f(x^*) = -19,2085$
Xin-She Yang (n=2) $f(x) = \sum_{i=1}^d \epsilon_i x_i ^i, \epsilon_i \sim U(0, 1)$	$x^* = (0, \dots, 0)$	$f(x^*) = 0$
Levy N.13 $f(x, y) = \sin^2(3\pi x) + (x - 1)^2[1 + \sin^2(3\pi y)] + (y - 1)^2[1 + \sin^2(2\pi y)]$	$(1, 1)$	$f(x^*) = 0$

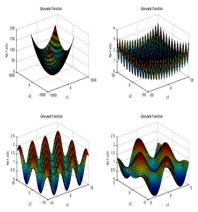
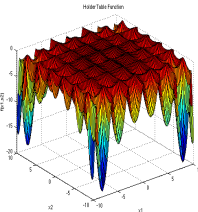
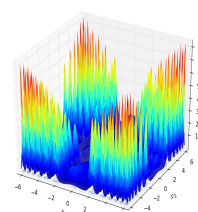
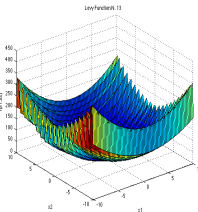
Tabla 3.2: Funciones *benchmark* con sus representaciones gráficas y descripción a la derecha.

Función	Gráfico	Descripción
Sphere Function		Función unimodal, convexa y separable. Evalúa la capacidad de convergencia en paisajes simples.
Ackley Function		Altamente multimodal, con muchos mínimos locales. Evalúa la capacidad de escapar de óptimos locales.
Rastrigin Function		Función multimodal periódica, con muchas trampas locales. Evalúa exploración global.

Continúa en la siguiente página

3. MARCO TEÓRICO

Tabla 3.2 – continúa de la página anterior

Función	Gráfico	Descripción
Griewank Function		Numerosos óptimos locales distribuidos de forma regular. Evalúa robustez y precisión del método.
Hölder Table Function		Extremadamente multimodal, con varios mínimos globales simétricos. Evalúa exploración en espacios complejos.
Xin-She Yang Function (n=2)		Función estocástica con coeficientes aleatorios. Prueba estabilidad y generalización.
Levy N.13 Function		Superficie con múltiples oscilaciones y un único óptimo global. Evalúa precisión de convergencia y algoritmos locales.

3.7.3. Validación Experimental

El procedimiento inicial consistió en aplicar el muestreo Montecarlo sobre un conjunto representativo de funciones benchmark. Las muestras generadas conformaron un mapa de soluciones que refleja las regiones del espacio exploradas por el algoritmo heurístico. A diferencia del caso hospitalario, donde cada

3. MARCO TEÓRICO

dimensión tenía un significado operacional concreto, en las funciones matemáticas los ejes representan variables abstractas sin correlación directa, lo que complica la interpretación geométrica del espacio.

Un espacio o región se considera convexo si, al tomar dos puntos cualesquiera dentro de esa región, el segmento de línea recta que los une está contenido completamente dentro de la región. K-Means está diseñado para trabajar con clusters que son inherentemente convexos o esféricos. El historial de muestreo de Rastrigin cerca de su óptimo se comporta como una nube esférica (convexa), lo que explica por qué K-Means logró una partición visualmente lógica en tu código. Sin embargo, el historial de funciones complejas a menudo generan clusters no-convexos (curvos o en forma de media luna), donde una línea recta entre dos puntos en el cluster podría cruzar una zona de vacío. K-Means es incapaz de separar estas formas no-convexas. Además, K-Means requiere especificar de antemano el número de clusters k , un parámetro cuya elección no es trivial: fijar un valor demasiado pequeño reduce la granularidad del análisis, mientras que valores excesivos generan sobreajuste y divisiones irrelevantes.

En este contexto, el mapa de Montecarlo exhibe una estructura dispersa, con zonas densas y vacíos intermedios, producto de la propia naturaleza probabilística del muestreo. K-means, al aplicarse sobre este tipo de configuraciones, no falla como método, sino que actúa sobre un conjunto carente de geometría convexa o uniformidad en la densidad, produciendo particiones que, aunque correctas desde el punto de vista formal, pueden no tener significado analítico útil. Este comportamiento pone de manifiesto la necesidad de incorporar etapas de análisis complementarias que permitan evaluar la validez y representatividad de las agrupaciones obtenidas.

En los experimentos preliminares se aplicó el algoritmo heurístico como un prototipo sobre distintas funciones tipo benchmark (Ackley, Sphere, Hölder Table, entre otras) con el objetivo de analizar el comportamiento del proceso de clustering empleado en la etapa de reducción del espacio de búsqueda. Para ello se utilizó el algoritmo K-Means, que asigna cada muestra al cluster cuyo centroide minimiza la distancia euclidiana en el espacio de coordenadas, es decir, agrupa las muestras

3. MARCO TEÓRICO

según su proximidad geométrica en $\mathbb{R}^n$, sin considerar el valor de la función objetivo.

Como consecuencia de esta formulación, K-Means tiende a generar particiones esencialmente ortogonales al espacio: en baja dimensión se observan divisiones verticales u horizontales, similares a cortar una esfera en secciones rectilíneas, en lugar de producir particiones radiales o concéntricas acordes a la estructura del paisaje de la función. Esto se evidencia, por ejemplo, en la función Sphere, ver Figura 3.2 donde el valor óptimo se concentra cerca del origen; sin embargo, K-Means la divide en regiones que recuerdan a una cruz cartesiana, en vez de formar anillos concéntricos con niveles similares de la función. De manera análoga, en la función Ackley, como se ve en la Figura 3.3, la región plana central queda fragmentada en subregiones arbitrarias sin relación con la forma real del paisaje.

A pesar de ello, este comportamiento puede resultar útil puntualmente para la reducción del espacio de búsqueda, ya que en ocasiones un cluster puede contener la región donde se localiza el óptimo global. No obstante, este efecto no es sistemático, pues depende de la inicialización aleatoria del algoritmo y, en particular, de la semilla utilizada para seleccionar los centros iniciales. Las figuras adjuntas ilustran claramente esta discrepancia entre la estructura geométrica inducida por K-Means y la estructura real del paisaje de las funciones benchmark.

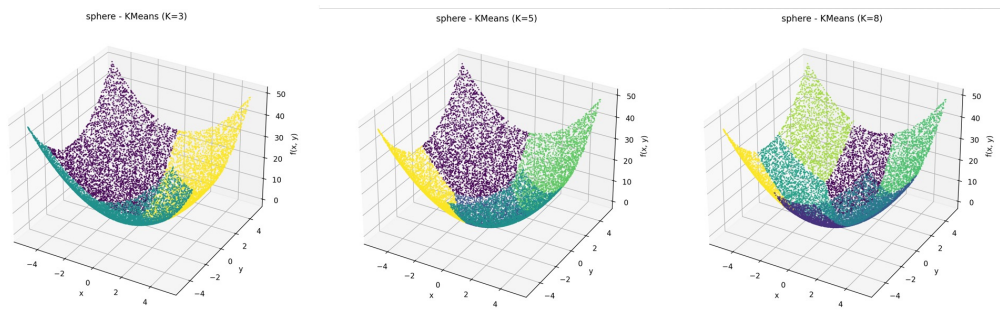


Figura 3.2: Función benchmark Sphere agrupada mediante K-Means utilizando diferentes valores de k .

3. MARCO TEÓRICO

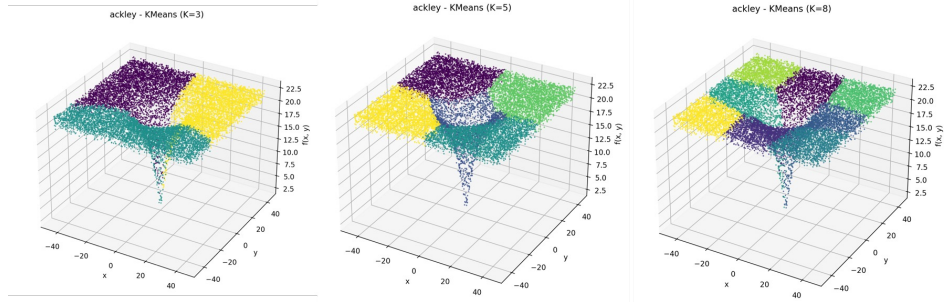


Figura 3.3: Agrupamiento de la función benchmark Ackley mediante K-Means utilizando distintos valores de k , mostrando la variación en las particiones del dominio

3.7.4. Análisis y Discusión

Los experimentos realizados permiten identificar tanto las fortalezas como las limitaciones del proceso de partición mediante K-Means aplicado a las muestras generadas por Montecarlo. Tal como se ha observado en funciones tipo benchmark como Ackley o Sphere, el algoritmo de clustering agrupa las muestras atendiendo a su proximidad geométrica en el espacio de coordenadas, es decir, minimiza la distancia euclidiana entre puntos en $\mathbb{R}^n$ sin considerar el valor asociado de la función objetivo. Esta regla de asignación induce particiones ortogonales respecto al sistema de ejes, generando regiones que pueden describirse como cortes rectilíneos del dominio. En funciones unimodales, como Sphere, este efecto se manifiesta en particiones en forma de cruz, en lugar de producir anillos concéntricos que representarían de forma natural la estructura del paisaje de optimización.

Sin embargo, este comportamiento no solo afecta a funciones unimodales. En funciones multimodales como Hölder Table o Michalewicz, la situación se vuelve aún más compleja. Estas funciones contienen múltiples mínimos globales o locales que pueden situarse a distancias similares en el espacio de coordenadas, pero separados por barreras significativas en términos de función objetivo. En estos casos, K-Means tiende a agrupar varios mínimos dentro de un mismo cluster siempre que las coordenadas sean próximas según su métrica interna, independientemente de la complejidad topológica o de la profundidad de dichos mínimos.

3. MARCO TEÓRICO

Este fenómeno introduce un problema adicional para la reducción del espacio de búsqueda. Si el objetivo es delimitar una región prometedora a partir del clustering, la presencia de varios mínimos en un mismo grupo implica que el área resultante puede abarcar una porción amplia del dominio, obligando a analizar prácticamente todo el mapa original para no descartar posibles soluciones óptimas. Esto reduce de manera considerable la utilidad de K-Means como mecanismo de selección espacial en contextos donde se pretende acelerar la convergencia mediante la restricción del dominio. Este comportamiento se puede ver en las Figuras 3.4 y 3.5

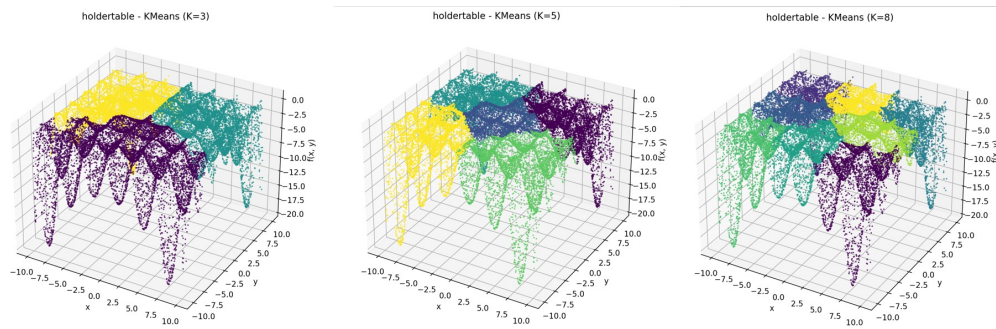


Figura 3.4: Representación de la función Hölder Table y su partición en clusters mediante K-Means para diferentes valores de k . Se observan los cambios en la segmentación del paisaje según la granularidad del clustering, destacando cómo valores mayores de k permiten identificar regiones más locales y abruptas de la función.

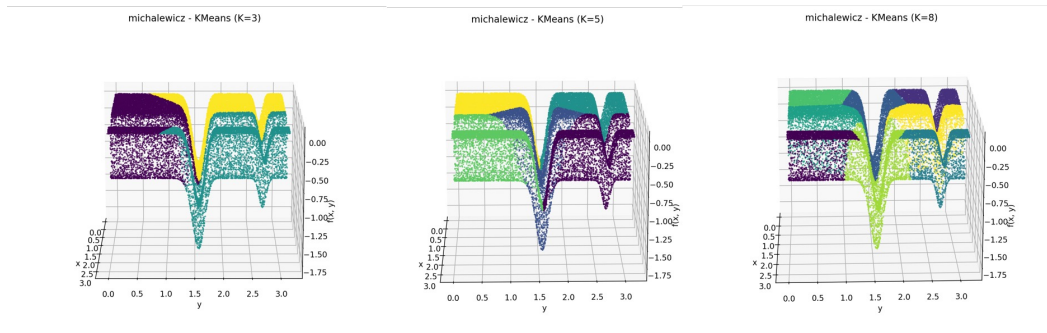


Figura 3.5: Aplicación de K-Means a la función Michalewicz utilizando diferentes valores de k . A medida que aumenta el número de clusters, el algoritmo captura con mayor detalle las regiones estrechas que contienen los mínimos locales profundos, facilitando su identificación en un paisaje altamente multimodal.

Incrementar el número de *clusters* k no resuelve completamente este inconveniente.

3. MARCO TEÓRICO

Aunque un mayor número de particiones conduce a regiones más pequeñas, también incrementa la complejidad computacional y no garantiza una separación significativa entre estructuras relevantes del paisaje, ya que la regla de asignación basada en la distancia euclidiana permanece inalterada. Además, la sensibilidad a la inicialización, determinada por la semilla empleada para seleccionar los centros iniciales, introduce variabilidad adicional en la forma de las particiones.

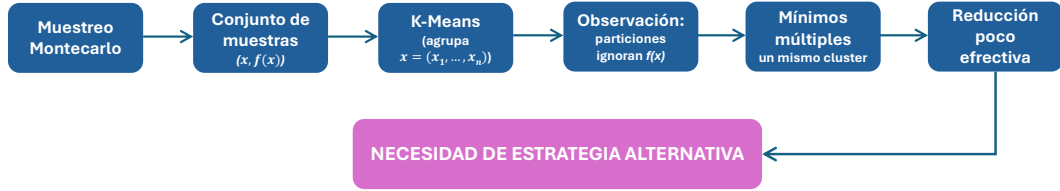


Figura 3.6: Función benchmark Sphere agrupada mediante K-Means y diferentes k .

El diagrama 3.6 resume el flujo conceptual seguido para estudiar el comportamiento del algoritmo base aplicado sobre funciones tipo benchmark. El proceso se inicia con el muestreo Montecarlo, a partir del cual se obtiene un conjunto de puntos $x \in \mathbb{R}^n$ junto con sus valores calculados $f(x)$. Posteriormente, se aplica el algoritmo de agrupamiento K-Means con el objetivo de estructurar el espacio de decisión en regiones que pudieran facilitar una eventual reducción del dominio de búsqueda.

Es importante señalar que el proceso de agrupamiento se realiza sobre las coordenadas x , y no sobre el par $(x, f(x))$. Esta decisión responde a consideraciones metodológicas: incorporar el valor de la función objetivo como una dimensión adicional implicaría operar en un espacio $\mathbb{R}^{n+1}$ cuyas escalas no son homogéneas, distorsionando las distancias euclídeas que definen el comportamiento de K-Means. Como consecuencia, se obtendrían clusters que reflejarían más las magnitudes de $f(x)$ que la geometría intrínseca del espacio de decisión, generando particiones poco coherentes con la topografía real del problema. De hecho, al operar únicamente sobre las coordenadas, K-Means produce particiones ortogonales características del algoritmo, segmentando el dominio en regiones rectilíneas que no necesariamente se corresponden con los mínimos de las funciones analizadas.

3. MARCO TEÓRICO

Sin embargo, al modificar la estrategia de agrupamiento para que K-Means utilice únicamente los valores de la función objetivo $f(\mathbf{x})$, ignorando las coordenadas espaciales, se observa un comportamiento notablemente diferente y más conveniente desde la perspectiva de reducción del espacio. Por ejemplo, en la función Sphere los clusters resultantes forman estructuras concéntricas, alineadas con los niveles de la función, mientras que en Ackley el óptimo global queda aislado en un cluster, facilitando su identificación. Esta estrategia permite concentrarse en regiones de interés de manera más eficiente y reduce significativamente el espacio de búsqueda en funciones unimodales o suavemente multimodales.

Aun así, en funciones altamente multimodales como Hölder Table o Michalewicz, la agrupación basada en $f(x)$ no garantiza una reducción significativa del dominio. En estos casos, varios mínimos relevantes pueden quedar agrupados dentro del mismo cluster aunque se encuentren en extremos opuestos del dominio, o quedan dispersos de forma que la delimitación de regiones prometedoras no simplifica la Exploración; es decir, la estrategia exige volver a analizar prácticamente todo el mapa de la función. Por ello, aunque clusterizar por $f(x)$ mejora notablemente la identificación de zonas de interés en funciones como Sphere o Ackley, sigue siendo necesario evaluar alternativas más sofisticadas para funciones con múltiples mínimos estrechos y paisajes complejos.

En las Figuras 3.7, 3.8, 3.9 y 3.10 se muestran los resultados de aplicar K-Means sobre los valores de la función objetivo para los mismos benchmarks mencionados. Se incluyen cuatro imágenes correspondientes a Ackley, Sphere, Hölder Table y Michalewicz, evidenciando cómo la clusterización por $f(x)$ produce una representación más intuitiva de las regiones relevantes, aunque con limitaciones en funciones altamente multimodales.

Los resultados obtenidos invitan a reflexionar sobre la idoneidad de utilizar K-Means como mecanismo primario para estructurar el espacio de decisión en el contexto de algoritmos heurísticos. Aunque el método proporciona una primera estratificación del dominio y es útil para detectar regiones en las muestras generadas por Montecarlo, su comportamiento está fuertemente condicionado por la métrica euclídea y por la suposición implícita de clusters aproximadamente

3. MARCO TEÓRICO

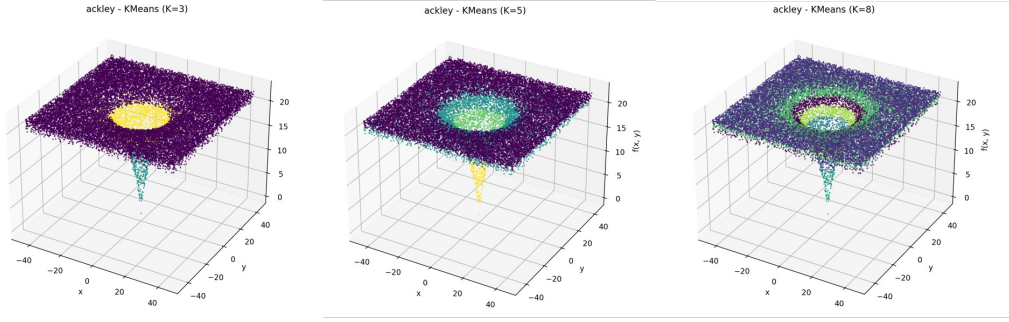


Figura 3.7: Clusterización de la función Ackley mediante K-Means basada en los valores de $f(x)$. El pico global queda aislado en un cluster, mostrando cómo esta estrategia permite concentrarse en regiones de interés y reducir efectivamente el espacio de búsqueda. Se muestran variaciones al utilizar diferentes valores de k

esféricos y separables por distancia. Estas hipótesis rara vez se cumplen en paisajes altamente no lineales o multimodales.

El análisis comparativo sugiere que para que la reducción del espacio de búsqueda sea realmente efectiva es necesario recurrir a estrategias que incorporen explícitamente la información aportada por la función objetivo o que adopten métricas y procedimientos de agrupamiento más flexibles.

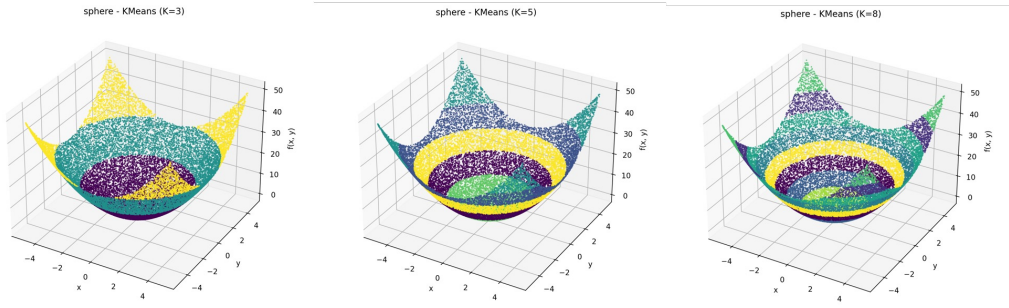


Figura 3.8: Clusterización de la función Sphere mediante K-Means utilizando los valores de la función objetivo $f(x)$. Se observan estructuras concéntricas que reflejan los niveles de la función y facilitan la identificación del mínimo global. Se muestran variaciones al utilizar diferentes valores de k .

Además, la experimentación se amplió para incluir la función Eggholder [71], un benchmark altamente multimodal usado en la literatura para evaluar algoritmos de optimización, los resultados completos de estas pruebas pueden consultarse en

3. MARCO TEÓRICO

el Apéndice A.1.

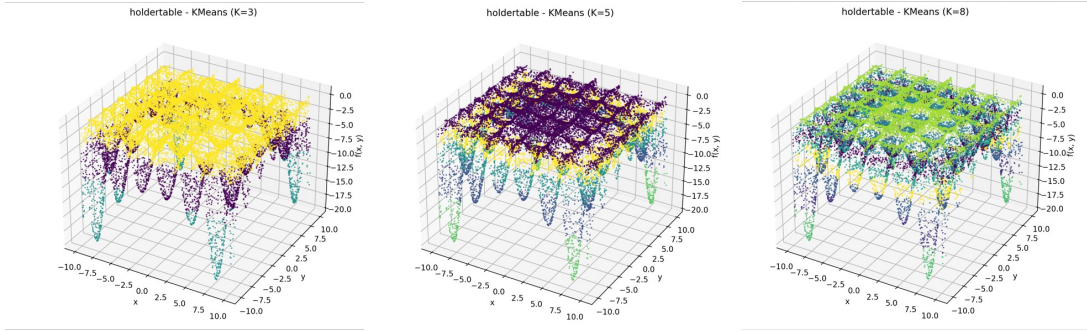


Figura 3.9: Clusterización de la función Hölder Table mediante K-Means usando los valores de $f(x)$. Aunque se identifican algunas regiones con valores deseables, la alta multimodalidad y la presencia de múltiples mínimos locales dificultan la reducción efectiva del dominio. Se pueden ver variaciones al utilizar diferentes valores de k .

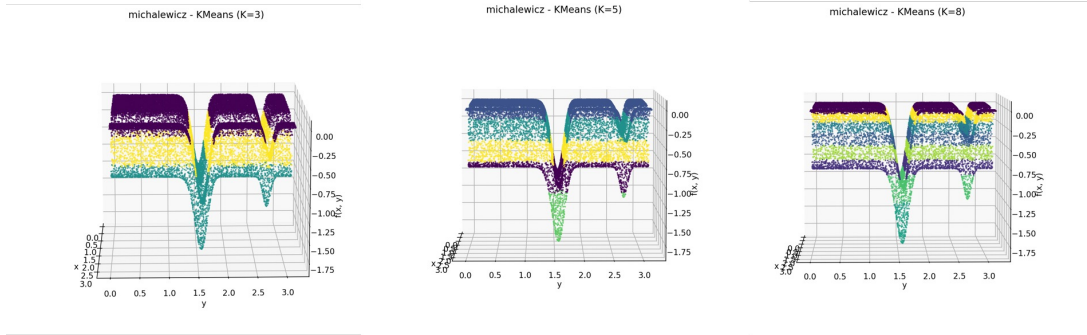


Figura 3.10: Aplicación de K-Means a la función Michalewicz utilizando los valores de $f(x)$. La función multimodal extrema genera clusters que no permiten acotar fácilmente el espacio de búsqueda, evidenciando limitaciones cuando se trabaja con mínimos estrechos y múltiples regiones relevantes. Se muestran variaciones al utilizar diferentes valores de k .

3.8. Conclusiones del Capítulo

En este capítulo se establece la base conceptual y metodológica que sustenta el enfoque propuesto. A partir de la revisión de los fundamentos teóricos de la optimización, se identifican las características esenciales típicas de los problemas

3. MARCO TEÓRICO

de optimización, así como los elementos estructurales que los definen: espacio de búsqueda, función objetivo y restricciones. Asimismo, se formaliza la definición general de un problema de optimización y se describen sus componentes clave, lo que permite enmarcar con rigor la propuesta metodológica dentro de un cuerpo teórico sólido.

El análisis experimental, por otro lado, mostró que el algoritmo base, aunque prometedor, presenta limitaciones cuando se aplica a funciones benchmark de optimización global. En particular, el muestreo estocástico generado por Montecarlo no puede ser tratado directamente con técnicas clásicas de clusterización como K-Means, debido a la compleja topología de las funciones multimodales. Esta limitación motivó el diseño de un proceso de experimentación con diversas funciones que constituye un paso previo esencial hacia la formulación de la propuesta metodológica presentada en el capítulo siguiente.

El problema fundamental con el uso de K-Means no reside únicamente en la selección del número de clusters (k), sino en que el algoritmo asume geometrías convexas y esféricas para cada cluster. Esta suposición choca directamente con la distribución de los puntos en las regiones de interés de las funciones. Al aplicar K-Means sobre las coordenadas x del espacio de decisión, se generan particiones rectilíneas que no reflejan los niveles de la función objetivo, produciendo divisiones que respetan poco la topografía real del paisaje de $f(x)$.

Al analizar la clusterización utilizando únicamente los valores de la función objetivo $f(x)$, se observa un comportamiento distinto y más alineado con la estructura de los paisajes de las funciones. Por ejemplo, en funciones unimodales como Sphere, los clusters resultantes forman estructuras concéntricas que reflejan los niveles de la función, mientras que en Ackley el pico global queda aislado en un cluster, facilitando su identificación y permitiendo una reducción más efectiva del espacio de búsqueda. Esta estrategia demuestra que incorporar $f(x)$ directamente en la clusterización puede mejorar la selección de regiones prometedoras frente al enfoque basado únicamente en coordenadas.

Aun así, en funciones multimodales con múltiples mínimos locales, como Hölder

3. MARCO TEÓRICO

Table o Michalewicz, esta estrategia presenta limitaciones. Incluso clusterizando por $f(x)$, varios mínimos relevantes ubicados en extremos opuestos del dominio pueden quedar incluidos en un solo cluster, impidiendo una reducción significativa del dominio. Incrementar el número de clusters k aumenta la complejidad computacional sin garantizar que los clusters coincidan con los mínimos individuales o con las regiones más relevantes de la función.

En consecuencia, el análisis sugiere que se requiere una estrategia alternativa que permita procesar el mapa generado por el muestreo Montecarlo de manera más alineada con la topología real de la función objetivo. Esto podría implicar otros métodos de clustering, estrategias híbridas o enfoques de análisis espacial que identifiquen de forma efectiva las regiones prometedoras, superando las limitaciones observadas con K-Means tanto sobre coordenadas como sobre valores de $f(x)$.

Glosario del Capítulo

A continuación se presenta el glosario 3.3 de el presente Capítulo, que incluye los términos y símbolos más relevantes utilizados en esta sección. Este recurso tiene como objetivo facilitar la comprensión de la terminología empleada y servir de referencia rápida durante la lectura.

Tabla 3.3: Glosario del Capítulo

Término	Definición
Benchmark	Conjunto de funciones matemáticas utilizadas como referencia para evaluar el comportamiento o el desempeño de algoritmos de optimización.
BBOB	<i>Black-box Optimization Benchmarking</i> , un entorno estandarizado para comparar algoritmos de optimización sin derivadas.
CEC	<i>Congress on Evolutionary Computation</i> , una competición ampliamente utilizada como referencia para evaluar técnicas evolutivas y metaheurísticas.
COCO	<i>COmparing COntinuous Optimization</i> , plataforma diseñada para la experimentación y comparación sistemática de algoritmos de optimización continua.
Algoritmo base	Método desarrollado por Cabrera et al. que combina simulación Montecarlo con técnicas de clustering mediante K-Means.
Métodos exactos	Procedimientos que garantizan encontrar la solución óptima del problema mediante Exploración total o parcial del espacio de búsqueda basada en modelos matemáticos formales. Su complejidad suele crecer exponencialmente con la dimensión.

Continúa en la siguiente página

3. MARCO TEÓRICO

Glosario del Capítulo (continuación)

Término	Definición
Métodos iterativos	Algoritmos que generan una secuencia progresiva de soluciones aproximadas mediante un proceso repetitivo. No aseguran optimalidad global pero permiten obtener soluciones de buena calidad con menor coste computacional.
Métodos exploratorios	Técnicas centradas en recorrer amplias regiones del espacio de búsqueda para evitar el estancamiento en óptimos locales. Son comunes en metaheurísticas y métodos estocásticos.
Búsqueda global	Estrategia orientada a identificar regiones prometedoras en el espacio de búsqueda a gran escala, con el objetivo de aproximarse al óptimo global.
Función objetivo $f(x)$	Función matemática que se desea minimizar o maximizar. El vector x representa una solución candidata dentro del dominio del problema.
Valor óptimo $f(x^*)$	Valor mínimo (o máximo) alcanzable por la función objetivo. El punto x^* es la solución en la que se consigue dicho valor.
x	Vector de variables de decisión del problema de optimización, $x \in \mathbb{R}^n$.
$\mathcal{D}$	Dominio factible del problema, que contiene todas las soluciones que satisfacen las restricciones.
$g_i(x) \leq 0$	Restricciones de desigualdad, $i = 1, \dots, m$, que limitan el conjunto de soluciones admisibles.
$h_j(x) = 0$	Restricciones de igualdad, $j = 1, \dots, p$, que deben cumplirse exactamente.

Continúa en la siguiente página

3. MARCO TEÓRICO

Glosario del Capítulo (continuación)

Término	Definición
$\min_{x \in \mathcal{D}} f(x)$	Formulación de un problema de minimización: búsqueda del vector x dentro del dominio factible que minimiza $f(x)$.
$\max_{x \in \mathcal{D}} f(x)$	Formulación de un problema de maximización: búsqueda del vector x dentro del dominio factible que maximiza $f(x)$.
Algoritmos Genéticos (GA)	Metaheurística inspirada en los principios de la evolución biológica, basada en operadores como selección, cruce y mutación.
Recocido Simulado (SA)	Algoritmo inspirado en el proceso físico de enfriamiento termodinámico. Permite aceptar soluciones peores con cierta probabilidad para escapar de óptimos locales.
Optimización por Enjambre de Partículas (PSO)	Técnica basada en el comportamiento colectivo. Las partículas se desplazan según su mejor experiencia individual y la del grupo.
Algoritmos de Colonia de Hormigas (ACO)	Metaheurística inspirada en la búsqueda de alimento por hormigas, donde la comunicación se modela mediante feromonas digitales.
Búsqueda Tabú (TS)	Método que usa memoria adaptativa para evitar volver a explorar soluciones previamente visitadas o consideradas no prometedoras.
Búsqueda por Entornos Variables (VNS)	Metaheurística que modifica de forma sistemática la estructura del vecindario para mejorar la Exploración y evitar óptimos locales.

Capítulo 4

Metodología propuesta

En esta sección se presenta el método desarrollado en el presente trabajo, concebido como una herramienta heurística diseñada para abordar problemas complejos de optimización desde una perspectiva centrada en el diseño y la interpretación estructural del problema.

El propósito de utilizar heurísticas en este trabajo no es competir directamente con algoritmos clásicos o mejorar su rendimiento en términos comparativos. Más bien, se plantea un enfoque alternativo en el que la heurística es parte de una estrategia de diseño orientada a reinterpretar el problema de optimización desde una perspectiva diferente. Esta visión no se limita a la búsqueda de soluciones óptimas, sino que incorpora la parametrización como un elemento estructural del modelo, lo que permite adaptar la búsqueda según la naturaleza y escala del problema.

Desde este punto de vista, la heurística propuesta no debe entenderse únicamente como un mecanismo de búsqueda, sino como una forma de codificar intuiciones sobre el problema, organizar y modular el espacio de soluciones de manera flexible y facilitar la Exploración de regiones prometedoras. La parametrización no necesariamente opera como una fase adicional del algoritmo, sino como un componente transversal que condiciona tanto la Exploración como la Explotación. Esto permite controlar ciertos aspectos del comportamiento del algoritmo, lo que

4. METODOLOGÍA PROPUESTA

a su vez facilita una adaptación más precisa al contexto específico de aplicación.

La metodología se fundamenta principalmente en la combinación de muestreo estocástico (Montecarlo) con técnicas de segmentación y agrupamiento, lo que sitúa al *Montecarlo Clustering Search Algorithm* (MCSA) dentro de la familia de las heurísticas. A diferencia de los enfoques clásicos centrados principalmente en la intensificación local y orientados a maximizar el rendimiento cuantitativo frente a algoritmos existentes, el MCSA propone un marco alternativo basado en un proceso de construcción progresiva de conocimiento sobre el espacio de soluciones. En lugar de limitarse a aplicar operadores de mejora local, el método genera, analiza y depura regiones del dominio a medida que avanza la búsqueda, integrando información obtenida en etapas previas para orientar etapas posteriores.

Para ello, introduce un mecanismo dual que equilibra explícitamente la Exploración global y la Explotación localizada. Este balance se controla mediante la parametrización del algoritmo, cuyos componentes funcionan como reguladores del comportamiento dinámico del proceso. De este modo, el método no solo persigue la obtención de soluciones eficientes, sino que también proporciona una estructura flexible que permite interpretar el problema, adaptar la estrategia de búsqueda y guiar la construcción iterativa de regiones prometedoras dentro del espacio de soluciones.

El algoritmo se apoya en una estructura modular, compuesta por tres etapas principales, una previa y dos operativas, que articulan la preparación de datos, la Exploración y la Explotación:

- **(1) Interpretación inicial del problema:** Se establece una representación estructurada del espacio de búsqueda, lo que incluye la preparación de datos y la codificación inicial del problema. Esta etapa, que no constituye una fase operativa del algoritmo, es una tarea previa que define cómo se organiza y delimita el espacio donde posteriormente actuará la heurística.
- **(2) Parametrización del Modelo:** Se introducen y ajustan los parámetros clave que modulan el comportamiento dinámico del algoritmo. Estos parámetros son un componente transversal que influye en las fases de

4. METODOLOGÍA PROPUESTA

Exploración y Explotación. Controlan aspectos como la amplitud de la búsqueda, la intensidad del refinamiento local, y las reglas de aceptación de cambios en las soluciones candidatas. Estos cambios se refieren a las modificaciones generadas sobre las soluciones actuales, que pueden ser retenidas si resultan beneficiosas según la función objetivo o rechazadas en caso contrario.

- **(3) Búsqueda y refinamiento:** Con la estructura inicial y los parámetros definidos, se ejecuta el proceso iterativo que combina Exploración y Explotación. La Exploración permite generar nuevas soluciones en regiones aún no evaluadas, mientras que la Explotación mejora las soluciones ya prometedoras. De este modo, el algoritmo identifica regiones de alto potencial y descarta áreas menos relevantes, optimizando progresivamente la búsqueda.

La interacción entre estas etapas permite construir una dinámica adaptativa, en la que el algoritmo responde a la evolución del problema mediante el ajuste de sus propias reglas internas. Esta capacidad de adaptación, combinada con una representación flexible del espacio de búsqueda, constituye uno de los aportes centrales del método propuesto.

En la siguiente sección se describe con mayor detalle cada componente del algoritmo, incluyendo la motivación, fundamentos teóricos, mecanismos operativos y el papel que juega la parametrización en la calidad y comportamiento de las soluciones obtenidas.

4.1. Presentación del Algoritmo

El MCSA es una heurística diseñada para abordar problemas de optimización combinatoria de alta dimensionalidad mediante un ciclo iterativo de *Exploración* y *Explotación*. Su originalidad radica en combinar el muestreo Montecarlo con técnicas de análisis de densidad y de agrupamiento jerárquico, lo que permite equilibrar la búsqueda global y el refinamiento local. La Figura 4.1 sintetiza este flujo y facilita la interpretación de las etapas que lo componen.

4. METODOLOGÍA PROPUESTA

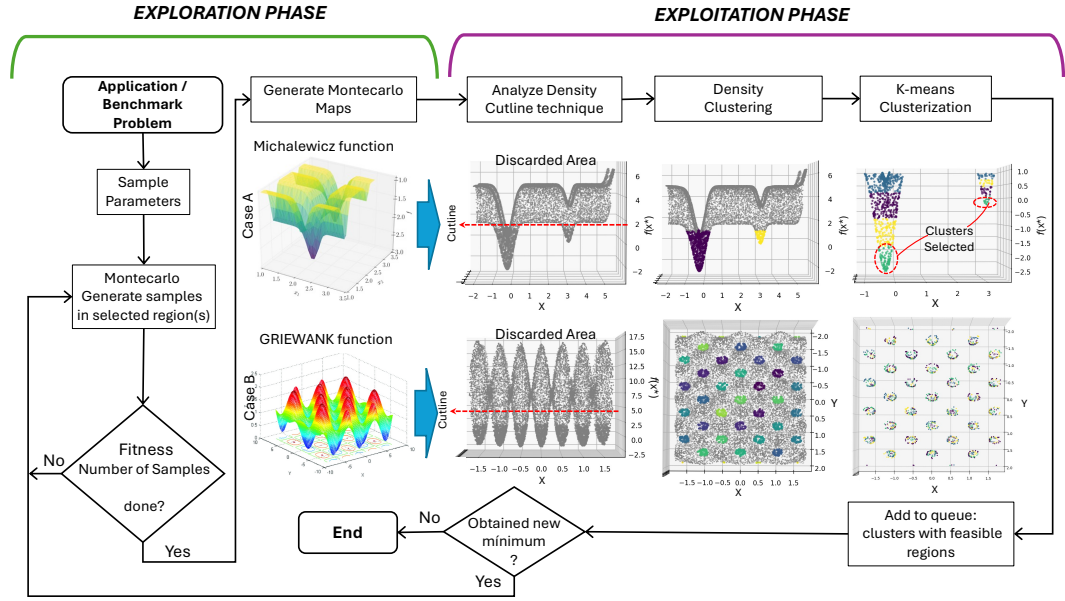


Figura 4.1: Diagrama de flujo de la heurística MCSA, ilustrado mediante dos casos de aplicación (*benchmark*) [44]. Este esquema proporciona una visión general de los principales componentes y del flujo del MCSA del proceso de optimización.

Fase de Exploración: Búsqueda

1. **Muestreo inicial.** A partir de la formulación del problema y sus restricciones, se define el espacio de búsqueda Ω . Se generan N soluciones candidatas $\{x_1, \dots, x_N\} \subset \Omega$ siguiendo un muestreo aleatorio uniforme (Montecarlo). Cada solución se calcula mediante la función objetivo $f(x_i)$ y se etiqueta su factibilidad según las restricciones.

En la Figura 4.2 se muestra la aplicación del criterio de parada durante la Fase de Exploración del MCSA sobre una función benchmark (Rastrigin) en dos dimensiones. Cada punto de la nube representa una muestra $(x_1, x_2, f(x))$, reflejando la cobertura inicial del espacio de búsqueda mediante muestreo Montecarlo. Los ejes horizontales corresponden a las coordenadas del dominio (x_1, x_2) y el eje vertical indica el valor de la función objetivo $f(x)$. Cuando se cumple el criterio de parada, la Exploración finaliza, evitando cálculos adicionales.

4. METODOLOGÍA PROPUESTA

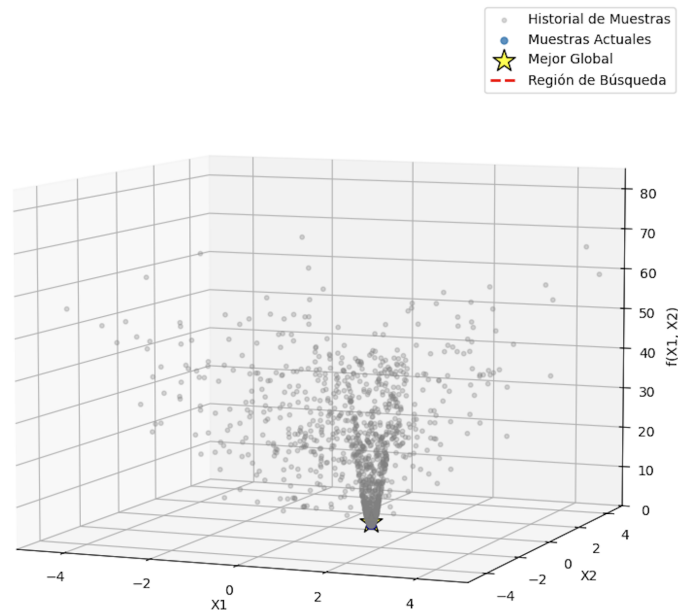


Figura 4.2: La nube de puntos es una representación en la que cada punto del gráfico corresponde a una muestra $(x_1, x_2, f(x))$. La dispersión inicial refleja la cobertura del espacio; a medida que avanza la Exploración y se activa el criterio de parada, definido como la condición que detiene la Exploración cuando se alcanza la precisión deseada en la media funcional o la cobertura máxima del espacio, el proceso finaliza evitando cálculos innecesarios y optimizando la eficiencia del muestreo.

2. **Mapa de Montecarlo.** El conjunto de pares $(x_i, f(x_i))$ constituye el mapa de Montecarlo. En la Figura 4.2 este mapa se visualiza como una nube de puntos que refleja la distribución de las muestras sobre el dominio, abarcando áreas con distintos valores de la función objetivo. La Fase de Exploración busca promover diversidad en la cobertura del espacio, recorriendo distintas regiones del dominio y registrando los valores correspondientes a cada muestra.

Fase de Explotación: Refinamiento

1. **Corte - clustering por densidad** Para cada nivel candidato de corte sobre la función objetivo $f(x)$, se seleccionan todas las muestras Montecarlo cuyo

4. METODOLOGÍA PROPUESTA

valor sea inferior a un umbral. Sobre este subconjunto se aplica el algoritmo DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) [28] con el objetivo de determinar cuántos clusters aparecen bajo ese nivel. El umbral óptimo se define como aquel que maximiza el número de clusters separados.

2. **Clustering por K-Means** Una vez seleccionado el umbral óptimo y determinadas las regiones del dominio asociadas a los clusters detectados, se aplica K-Means sobre cada una de estas regiones para obtener un número predeterminado k de centroides. Estos centroides permiten focalizar la Explotación en zonas delimitadas del espacio de búsqueda. En la Figura 4.1, este paso se muestra en la zona *Exploitation Phase*.
3. **Generación de nuevas muestras.** A partir de los centroides se generan nuevas soluciones en torno a estas regiones de interés. Cada nueva solución se calcula y, si mejora el mejor valor conocido, se actualiza la mejor solución global.
4. **Ciclo Exploración - Explotación.** El proceso de Exploración, análisis de densidad, corte y Explotación se repite iterativamente hasta cumplir los criterios de parada. Dichos criterios pueden basarse en un número máximo de iteraciones, en la ausencia de mejoras significativas o en la convergencia del valor de la función objetivo.

4.2. Parametrización de MCSA

Tal como se describió en la estructura general del algoritmo, la parametrización constituye un componente transversal que define el comportamiento de las fases operativas del ciclo, Exploración, Generación y procesamiento del mapa de Montecarlo, Reducción y Explotación. No se trata de una fase independiente, sino de un conjunto de decisiones que regulan la dinámica del algoritmo en todas las etapas de búsqueda. Además, estos parámetros no poseen valores universales, sino que pueden ajustarse en función del problema específico. A continuación se

4. METODOLOGÍA PROPUESTA

describen los principales componentes de dicha parametrización.

- **Parámetros de Exploración.** Definen cuántas muestras se generan en la fase inicial (N) y el dominio Ω . Un N grande aporta mayor diversidad pero incrementa el coste computacional.
- **Parámetros de densidad y corte.** Comprenden el número de intervalos M usados para discretizar el eje de valores de la función objetivo y el umbral de densidad τ que decide qué intervalos se retienen. Estos parámetros determinan cuán selectiva es la fase de *pre-clustering*.
- **Parámetros de clustering.** Incluyen el número de centroides k usados en K-means. Un valor alto de k produce más clusters y, por tanto, una Explotación más intensiva en diversas regiones; un valor bajo concentra el esfuerzo en menos zonas.
- **Criterios de parada y número de repeticiones.** Establecen cuándo finalizar el ciclo (por ejemplo, número máximo de iteraciones, convergencia de la media de $f(x)$ o ausencia de mejoras). Repetir el proceso con diferentes inicializaciones aleatorias permite evaluar la consistencia de los resultados y aumenta la probabilidad de encontrar soluciones de calidad representativa en distintos escenarios de partida.

Estos parámetros proporcionan flexibilidad para adaptar el algoritmo a distintos problemas de optimización. Ajustando la muestra inicial, la discretización de la función objetivo, el umbral de densidad y el número de centroides, se modula el equilibrio entre Exploración global y Explotación local que refleja la Figura 4.1. En las secciones siguientes se detallan cada una de las fases operativas: Exploración, Generación y procesamiento del mapa de Montecarlo, Reducción del mapa y Explotación.

4.3. MCSA Fase 1: Exploración

La primera fase del algoritmo MCSA se centra en la Exploración del espacio de soluciones, especialmente en problemas de alta dimensionalidad y sin conocimiento previo sobre la distribución de los óptimos. Para ello, se emplea un enfoque de muestreo aleatorio uniforme (Montecarlo [84]), que permite recorrer el espacio de manera flexible, simple y no sesgada, sin imponer estructuras que podrían limitar la diversidad de las soluciones.

Aunque técnicas como *Latin Hypercube Sampling* [51, 62] o secuencias de Sobol [12] ofrecen una cobertura más uniforme por dimensión, su efectividad disminuye en espacios muy grandes, donde se requieren miles de muestras para lograr una cobertura comparable. Además, dichas técnicas pueden introducir sesgos si se desconoce la topología de la función objetivo, mientras que Montecarlo mantiene la aleatoriedad necesaria para una Exploración inicial amplia.

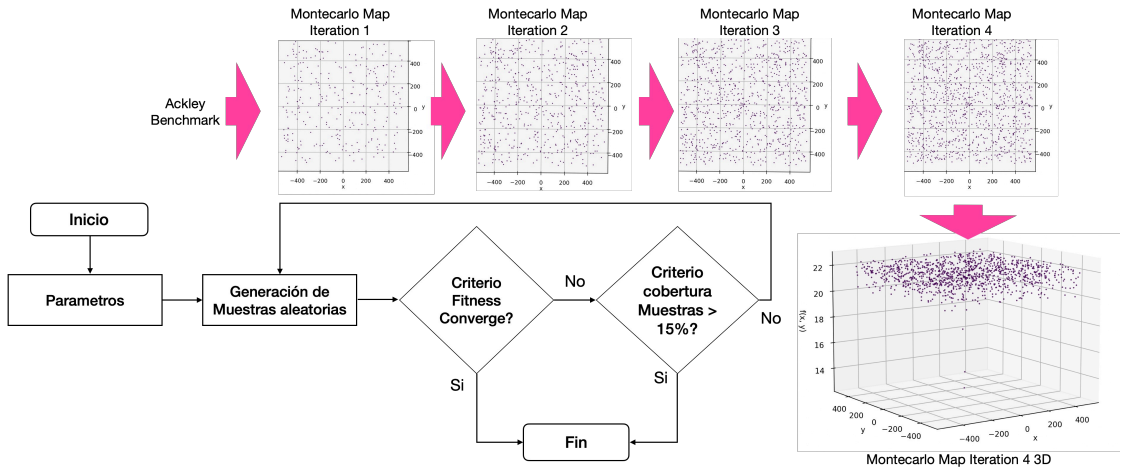


Figura 4.3: Esquema general de la Fase de Exploración del MCSA.

Durante esta fase, se generan soluciones candidatas mediante bloques de muestreo. Cada bloque consiste en un número configurable de muestras,

$$\mathbf{x}^{(i)} \in \Omega \subseteq \mathbb{R}^n, \quad (4.1)$$

4. METODOLOGÍA PROPUESTA

evaluadas mediante la función objetivo $f(\mathbf{x})$. El objetivo es identificar muestras aptas (*fit samples*), es decir, aquellas cuyo valor de $f(\mathbf{x})$ resulte competitivo según un umbral de aptitud, y construir un *mapa Montecarlo* que represente de manera empírica las regiones con mayor concentración de soluciones de interés dentro del dominio.

En la Figura 4.3 se muestra este proceso: el algoritmo genera sucesivos bloques de muestras independientes distribuidas uniformemente sobre Ω , las evalúa y conserva únicamente aquellas que cumplen el umbral de aptitud. La acumulación progresiva de estas muestras permite construir un mapa inicial del espacio, que servirá como punto de partida para las siguientes etapas del MCSA.

Para garantizar la consistencia y convergencia del muestreo se utilizan dos criterios de parada, ambos ilustrados en la misma Figura 4.3:

1. **Criterio de precisión.** Se compara la media de los valores de la función objetivo entre bloques consecutivos de muestras. Se establece una tolerancia de 10^{-2} como parámetro predeterminado, y se requieren al menos dos iteraciones para calcular la media entre los dos conjuntos de datos.
2. **Criterio de cobertura.** Se define un porcentaje máximo del espacio de búsqueda que puede ser muestreado, típicamente 15 % del total. Este límite es configurable para adaptarse a problemas de distinta dimensionalidad, equilibrando exhaustividad y eficiencia computacional.

El tamaño del bloque de muestreo y el porcentaje máximo de cobertura permiten al usuario ajustar la granularidad de la Exploración y la eficiencia del proceso, asegurando que el mapa Montecarlo represente adecuadamente la diversidad de soluciones posibles sin generar tiempos de cómputo excesivos.

Finalmente, las muestras obtenidas en esta fase constituyen la base para la siguiente etapa del MCSA: la Explotación mediante clustering. En ella, se analizarán densidades locales de soluciones y se seleccionarán regiones con mayor potencial, enfocando la búsqueda hacia óptimos prometedores. El siguiente pseudocódigo 1 muestra en resumen lo que ocurre en esta etapa:

4. METODOLOGÍA PROPUESTA

Algorithm 1 MCSA - Fase 1: Exploración

Require: Número de dimensiones n , espacio de búsqueda Ω , tamaño de bloque B , porcentaje máximo de cobertura $P_{\text{máx}}$, tolerancia ϵ

Ensure: Mapa Montecarlo $\mathcal{M}$ con muestras aptas

```
1: Inicializar  $\mathcal{M} \leftarrow \emptyset$ 
2: Inicializar contador de muestras totales  $N_{\text{total}} \leftarrow 0$ 
3: Inicializar media anterior  $\bar{f}_{\text{prev}} \leftarrow \infty$ 
4: Inicializar indicador de convergencia  $\text{converge} \leftarrow \text{false}$ 
5: while  $N_{\text{total}} < P_{\text{máx}} \cdot |\Omega|$  and not  $\text{converge}$  do
6:   Generar bloque de  $B$  muestras aleatorias:  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(B)}\} \subset \Omega$ 
7:   for cada muestra  $\mathbf{x}^{(i)}$  do
8:     Calcular  $f_i \leftarrow f(\mathbf{x}^{(i)})$ 
9:     if  $f_i$  supera el umbral de aptitud then
10:      Agregar  $(\mathbf{x}^{(i)}, f_i)$  a  $\mathcal{M}$ 
11:    end if
12:  end for
13:   $N_{\text{total}} \leftarrow N_{\text{total}} + B$ 
14:  Calcular media actual del bloque:  $\bar{f}_{\text{curr}} \leftarrow \frac{1}{B} \sum_{i=1}^B f(\mathbf{x}^{(i)})$ 
15:  if  $|\bar{f}_{\text{curr}} - \bar{f}_{\text{prev}}| < \epsilon$  y  $N_{\text{total}} \geq 2B$  then
16:     $\text{converge} \leftarrow \text{true}$ 
17:  end if
18:   $\bar{f}_{\text{prev}} \leftarrow \bar{f}_{\text{curr}}$ 
19: end while
```

4.3.1. Definición del espacio de búsqueda y generación de muestras aleatorias

La Fase de Exploración del MCSA tiene como objetivo generar un conjunto suficientemente amplio y diverso de soluciones iniciales, que permita cubrir el espacio de búsqueda Ω . Este muestreo se realiza de forma aleatoria, siguiendo un esquema de Montecarlo uniforme.

Ahora bien, tras la Fase de Exploración, el algoritmo MCSA produce un conjunto amplio de muestras aleatorias que representan de manera empírica el espacio factible del problema. Este conjunto, denominado *Mapa de Montecarlo*, contiene tanto soluciones de alta calidad como combinaciones menos favorables, reflejando la diversidad del espacio de búsqueda. Sin embargo, este mapa todavía no posee

4. METODOLOGÍA PROPUESTA

una estructura clara que pueda ser utilizada directamente para la identificación de regiones prometedoras.

1. **Definición del espacio de búsqueda.** Sea el problema definido en n dimensiones, con variables que cumplen $\mathbf{x} = (x_1, \dots, x_n) \in \Omega \subseteq \mathbb{R}^n$, donde Ω representa las restricciones del problema.
2. **Generación de la muestra inicial.** Se generan N soluciones independientes:

$$\mathcal{S} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}, \quad \mathbf{x}^{(i)} \sim U(\Omega), \quad (4.2)$$

donde $U(\Omega)$ denota la distribución uniforme sobre el espacio de búsqueda. Cada solución candidata $\mathbf{x}^{(i)}$ se calcula con la función objetivo f .

3. **Evaluación de factibilidad y desempeño.** Cada muestra se etiqueta como factible o no factible en función de las restricciones del problema:

$$\text{factible}(\mathbf{x}^{(i)}) = \begin{cases} 1 & \text{si } \mathbf{x}^{(i)} \in \Omega_{\text{factible}}, \\ 0 & \text{en caso contrario.} \end{cases} \quad (4.3)$$

4. **Base para etapas posteriores.** El subconjunto $\mathcal{S}_F = \{\mathbf{x}^{(i)} \in \mathcal{S} \mid \text{factible}(\mathbf{x}^{(i)}) = 1\}$ constituye un subconjunto factible de las muestras que cumplen todas las restricciones del problema y sirve como entrada para la Fase de Explotación, donde se aplican procedimientos de *clustering* y corte para identificar regiones prometedoras del espacio de búsqueda.

Este muestreo inicial es un paso importante, ya que puede promover que la Explotación posterior se base en regiones prometedoras y suficientemente representadas en la muestra.

En el contexto de este trabajo, las muestras generadas mediante el algoritmo de Exploración Montecarlo presentan una distribución espacial característica: no se organizan en estructuras lineales ni en anillos definidos, sino que tienden a formar

4. METODOLOGÍA PROPUESTA

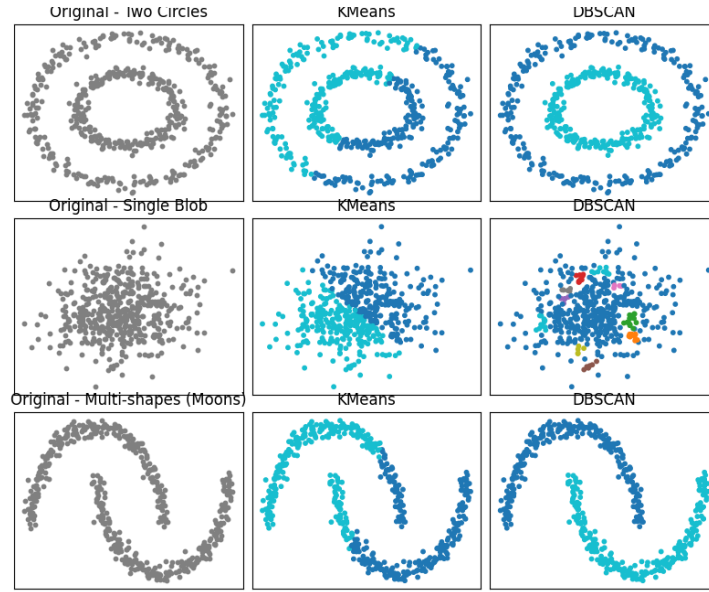


Figura 4.4: Comparativa de técnicas de agrupamiento.

conglomerados densos e irregulares, similares a las masas observadas en el ejemplo central del análisis comparativo de la figura 4.4.

Antes de aplicar técnicas de clustering, es necesario realizar un proceso previo que permita extraer estructuras relevantes del conjunto completo de muestras Montecarlo. Aunque la Exploración Montecarlo garantiza una buena cobertura del espacio de búsqueda, esta distribución uniforme provoca que, si se aplica clustering directamente, el algoritmo tienda a considerar todo el dominio como un único cluster. Para evitarlo, se introduce un procedimiento de filtrado basado en niveles de la función objetivo, que permite revelar regiones antes de la Fase de Explotación. Este tratamiento incluye, por ejemplo:

- Segmentación por intervalos de la función objetivo: el rango de valores de la función se divide en M intervalos, lo que facilita el análisis local de densidades y el posterior corte de regiones menos relevantes.
- Identificación de clusters por densidad: dentro de cada intervalo, se evalúa la concentración de regiones.

Este paso intermedio asegura la transición de la Exploración a la Explotación.

4. METODOLOGÍA PROPUESTA

Al preparar el mapa de esta manera, los algoritmos de clustering posteriores (DBSCAN y K-Means) pueden identificar clusters permitiendo que la Fase de Explotación se enfoque en regiones de relevancia para la optimización.

4.4. MCSA Fase 2: Explotación

Tras completar la Fase de Exploración y disponer del Mapa de Montecarlo que cubre de manera amplia el dominio del problema, la heurística entra en la Fase de Explotación. El objetivo de esta etapa es identificar y estructurar las regiones más prometedoras del espacio de búsqueda, concentrando el análisis en zonas donde se observa un comportamiento potencialmente óptimo.

El primer mecanismo de esta fase es el procedimiento de barrido por niveles de la función objetivo, que permite revelar estructuras internas en el mapa aplicando DBSCAN sobre subconjuntos filtrados. Este proceso detecta automáticamente los niveles de $f(x)$ en los que emergen clusters diferenciados, lo que constituye la base para el análisis posterior.

Una vez identificado el nivel de corte óptimo y obtenidas las regiones iniciales mediante DBSCAN, la Explotación continúa con un *esquema de clustering jerárquico en dos niveles*, diseñado para organizar de forma más precisa las zonas de interés:

1. **Clustering por densidad (DBSCAN):** se detectan regiones densas de soluciones sin necesidad de fijar a priori el número de clusters. Esta etapa elimina ruido, aísla configuraciones dispersas y delimita las zonas donde la densidad de soluciones es significativa.
2. **Clustering por centroides (K-Means):** dentro de las regiones densas identificadas por DBSCAN, se aplica K-Means para refinar la partición en subconjuntos más compactos y homogéneos. Los centroides resultantes representan soluciones candidatas concentradas en zonas de alta promesa.

Este doble enfoque permite que la Fase de Explotación avance de manera guiada,

4. METODOLOGÍA PROPUESTA

combinando la capacidad de DBSCAN para descubrir estructuras de forma arbitraria con la precisión de K-Means para modelar regiones coherentes dentro de los clusters detectados.

4.4.1. Reducción del Mapa de Montecarlo: Procedimiento de Barrido/Corte

Una vez generado el mapa de Montecarlo, el siguiente paso consiste en aplicar un procedimiento sistemático para identificar estructuras relevantes en el espacio de búsqueda. Si se aplicara clustering directamente sobre todas las muestras, las cuales cubren el dominio de manera amplia debido al muestreo Montecarlo, los algoritmos densidad-basados como DBSCAN tenderían a producir un único cluster, impidiendo detectar regiones diferenciadas.

Para evitarlo, se realiza un barrido por niveles de la función objetivo $f(x)$. Para cada intervalo de valores de $f(x)$, se seleccionan las muestras que quedan por debajo del nivel de corte y se aplica DBSCAN sobre ese subconjunto. Este enfoque permite revelar, a distintos niveles de la función objetivo, la aparición o desaparición de clusters, facilitando identificar los umbrales en los que el espacio se organiza en regiones separadas con comportamiento significativo para la optimización.

El procedimiento realiza un barrido sobre el eje de la función objetivo $f(x)$. El rango completo de valores de $f(x)$ se discretiza en un número fijo de niveles, determinado por el parámetro heurístico **MaxSteps** (en esta implementación, 10). Cada nivel define un umbral o *cutline* sobre $f(x)$ que selecciona únicamente las muestras cuyo valor es inferior al nivel actual del barrido.

Este proceso puede interpretarse como un desplazamiento vertical progresivo desde los valores más altos de $f(x)$ hacia niveles más bajos, examinando en cada paso la estructura que emerge al aplicar DBSCAN sobre el subconjunto filtrado. Si se intentara aplicar clustering directamente a todas las muestras Montecarlo, las cuales cubren uniformemente el dominio debido a la naturaleza del muestreo aleatorio, DBSCAN tendería a identificar un único cluster. Por ello, el barrido por

4. METODOLOGÍA PROPUESTA

niveles es necesario para revelar estructuras internas que de otro modo quedarían ocultas.

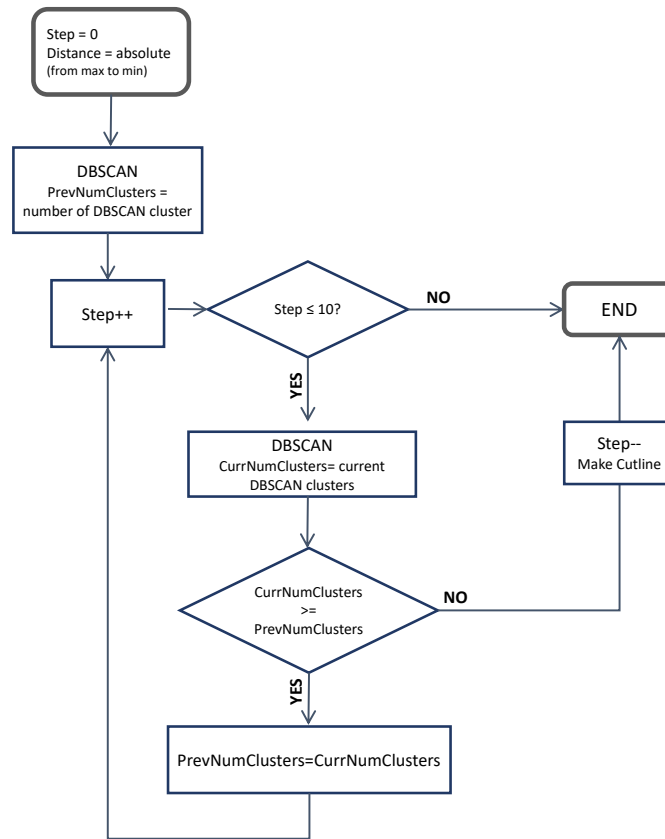


Figura 4.5: Diagrama de flujo para la determinación de la línea de corte en la Fase de Explotación del MCSA, combinando segmentación por intervalos y evaluación iterativa por DBSCAN.

La lógica detallada del procedimiento se muestra en el diagrama de flujo de la Figura 4.5. El algoritmo inicia calculando el número de clusters detectado en el primer nivel y avanza paso a paso, comparando en cada iteración el número actual de clusters con el obtenido en el nivel anterior. Si el número de clusters aumenta o se mantiene estable, el procedimiento continúa descendiendo en el eje de $f(x)$. En cambio, cuando el número de clusters deja de mejorar, o se alcanza el número máximo de niveles definido, el proceso se detiene y el nivel de corte seleccionado corresponde al último punto en el que se observó el número máximo de clusters diferenciados.

4. METODOLOGÍA PROPUESTA

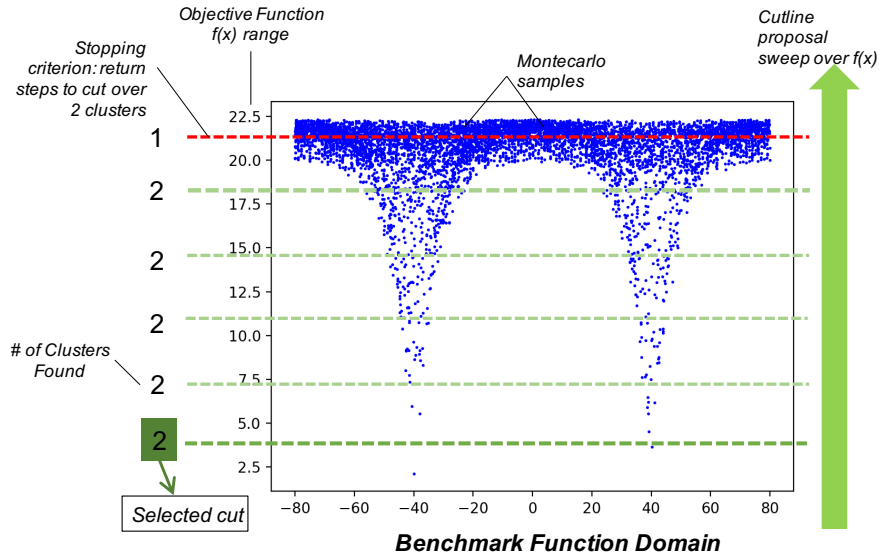


Figura 4.6: Ejemplo de corte y agrupamiento sobre una función benchmark compuesta.

Este mecanismo permite identificar automáticamente el umbral de $f(x)$ que separa el mapa en regiones bien definidas, revelando valles, depresiones o estructuras relevantes para la optimización. La Figura 4.6 ilustra un ejemplo del procedimiento: para cada nivel de corte, el subconjunto filtrado produce una segmentación distinta del dominio; el nivel óptimo corresponde al punto donde DBSCAN detecta el mayor número de clusters.

4.4.2. Algoritmo de clustering DBSCAN y justificación del análisis basado en densidad

La elección de DBSCAN como algoritmo de clustering para el análisis del Mapa de Montecarlo no fue arbitraria. Tras analizar diferentes métodos disponibles en librerías científicas como `scikit-learn` [76], se observó que los algoritmos basados en particiones, como K-Means, presentan limitaciones importantes en este contexto: requieren especificar de antemano el número de clusters y asumen que estos son convexos y aproximadamente esféricos. Estas suposiciones no se ajustan a la estructura no convexa y altamente irregular que pueden presentar las regiones

4. METODOLOGÍA PROPUESTA

interesantes de un espacio de optimización.

Por el contrario, DBSCAN identifica clusters como regiones de alta densidad separadas por áreas de baja densidad. Esta propiedad resulta especialmente adecuada para los datos generados mediante Montecarlo, en los que las geometrías relevantes pueden tener formas arbitrarias y no estar separadas por fronteras lineales. Además, DBSCAN presenta una ventaja fundamental frente a métodos como K-Means: no requiere especificar el número de clusters, sino que lo determina automáticamente a partir de la estructura del conjunto de datos. Esta característica es determinante en el contexto del heurístico, ya que el número de regiones de interés no es conocido a priori.

El algoritmo utiliza dos parámetros: ϵ (o **eps**), que define el radio del vecindario de cada punto, y **min_samples**, que establece el número mínimo de muestras necesario para considerar un punto como *core point*. Los puntos núcleo forman el esqueleto de las regiones densas, mientras que los puntos alcanzables se agregan como periféricos, y los puntos que no cumplen estos requisitos se etiquetan como ruido. La Figura 4.7 ilustra esta clasificación.

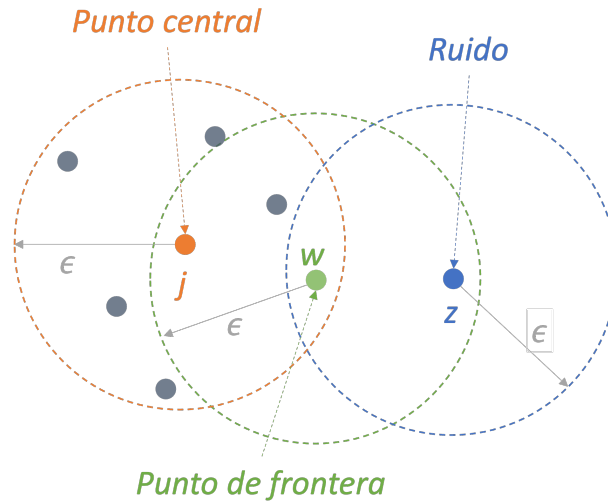


Figura 4.7: Clasificación de muestras en el algoritmo DBSCAN para **minPts** = 6. Los puntos núcleo (core), de borde (border) y de ruido (noise) se identifican en función de su densidad local.

4. METODOLOGÍA PROPUESTA

A pesar de estas ventajas, la aplicación directa de DBSCAN sobre todo el conjunto de muestras Montecarlo no resulta efectiva. Esta limitación motivó la introducción del *barrido por niveles de $f(x)$* descrito en la Sección anterior, donde DBSCAN se aplica únicamente a subconjuntos filtrados mediante umbrales crecientes sobre el valor de la función objetivo. Es precisamente en estos subconjuntos en los que emergen las diferencias de densidad necesarias para que DBSCAN revele agrupamientos naturales en el espacio de búsqueda.

Una vez identificadas las zonas densas mediante DBSCAN, se aplica posteriormente el algoritmo K-Means sobre los subconjuntos de puntos válidos. El propósito de esta segunda etapa es refinar los agrupamientos, dividiendo las regiones densas en subconjuntos más compactos y homogéneos. Aunque K-Means es sensible a la inicialización y tiende a generar particiones convexas, su aplicación en este contexto resulta adecuada, ya que opera únicamente sobre zonas previamente validadas por densidad y, por tanto, con una estructura interna más estable.

La combinación secuencial de un método basado en densidad seguido de un método particional aporta varias ventajas metodológicas:

- Permite excluir el ruido detectado en la primera etapa y enfocar el análisis en las regiones realmente relevantes.
- Facilita una identificación jerárquica de regiones: primero mediante densidad (estructuras arbitrarias), y posteriormente mediante particionamiento refinado.
- Proporciona centroides representativos dentro de cada región densa, lo que simplifica la caracterización geométrica del espacio y mejora la interpretabilidad del proceso de Explotación.

4.4.3. Definición de regiones potenciales

Tras obtener los clusters resultantes del proceso de Explotación, el siguiente paso consiste en transformar dichos agrupamientos en regiones potenciales donde se

4. METODOLOGÍA PROPUESTA

focalizará la generación de nuevas muestras. Este mecanismo permite acotar el dominio en torno a las zonas más prometedoras identificadas en las etapas previas.

El procedimiento se estructura de manera metodológica en tres pasos:

1. **Selección de la mejor muestra en cada cluster.** Para cada cluster identificado, se selecciona la muestra que presenta el menor valor de la función objetivo $f(x)$. Esta muestra actúa como representante del comportamiento más prometedor dentro de la región.
2. **Determinación de las coordenadas centrales.** Las coordenadas de la mejor muestra se utilizan como centro geométrico de la región potencial. Este punto captura el óptimo local observado y sirve como referencia para definir la nueva zona de búsqueda.
3. **Construcción de la región potencial en cada dimensión.** A partir de la dispersión de las muestras del cluster, se calcula su extensión en cada dimensión del espacio de decisión. Esta extensión refleja la variabilidad interna del cluster. La región potencial se define como un hiperrectángulo centrado en las coordenadas de la muestra óptima, empleando como semilongitud la mitad de la extensión del cluster en cada dimensión. De este modo:
 - se mantiene la escala real del cluster,
 - se respetan sus proporciones geométricas,
 - y se focaliza la búsqueda en torno a los puntos más prometedores sin perder variabilidad exploratoria.

El resultado final es un conjunto de regiones potenciales, una por cada cluster significativo, que delimita dominios locales donde se generarán nuevas muestras en la siguiente iteración del heurístico. Esta estrategia dirige la búsqueda hacia las zonas más relevantes del espacio sin comprometer la diversidad necesaria para evitar óptimos locales prematuros.

Para ilustrar de manera visual el procedimiento descrito, se incluye a continuación

4. METODOLOGÍA PROPUESTA

un ejemplo aplicado a la función de Michalewicz ???. La Figura 4.8 muestra el conjunto de muestras generadas mediante Montecarlo junto con la línea de corte seleccionada durante la Fase de Explotación. Este corte filtra las muestras permitiendo revelar la estructura interna del espacio únicamente en el intervalo donde emergen los clusters relevantes.

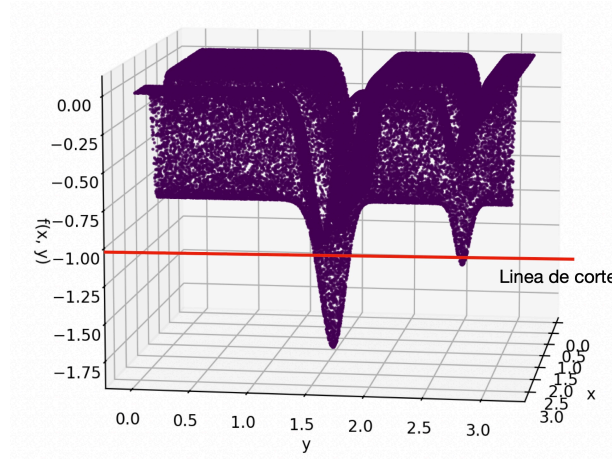


Figura 4.8: Ejemplo del corte aplicado sobre el mapa de Montecarlo para la función de Michalewicz, mostrando la línea de corte seleccionada tras el barrido por niveles. Se pueden observar dos regiones en la parte inferior de la línea de corte, que serán seleccionadas como regiones potenciales.

Una vez identificados los clusters a partir del conjunto filtrado, el heurístico procede a la definición de regiones potenciales. La Figura 4.9 representa este proceso mediante dos iteraciones consecutivas del MCSA. En la primera iteración (izquierda), se observa cómo cada cluster se analiza de forma independiente: (1) se selecciona la muestra con el menor valor de $f(x)$, (2) se calcula la extensión del cluster en cada dimensión, y (3) se construye una región acotada centrada en las coordenadas de la mejor muestra.

En la iteración siguiente (derecha), el proceso se repite de manera idéntica, pero ahora restringido únicamente a las regiones generadas en la iteración anterior. Se observa cómo las nuevas muestras se concentran en dominios más específicos, reflejando la progresiva focalización del heurístico hacia las zonas más prometedoras del espacio de búsqueda.

4. METODOLOGÍA PROPUESTA

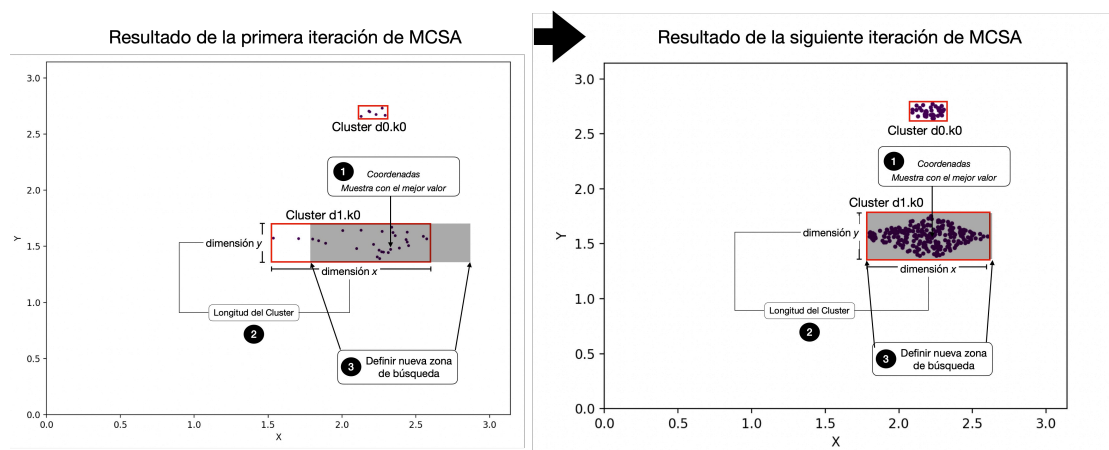


Figura 4.9: Proceso de construcción de regiones potenciales en dos iteraciones consecutivas del MCSA. Las regiones mostradas corresponden a las dos zonas destacadas en la figura anterior, vistas en el plano $[X,Y]$ (vista superior), identificando los picos de la función de Michalewicz como regiones potenciales seleccionadas para el refinamiento. (1) Selección de la mejor muestra en cada cluster, (2) cálculo de la longitud del cluster en cada dimensión, (3) definición de la nueva zona de búsqueda centrada en el óptimo local.

Por lo tanto, una vez definidas las regiones potenciales, el heurístico reinicia el ciclo de manera controlada. En lugar de volver a explorar el dominio completo, la nueva fase de muestreo Montecarlo se restringe exclusivamente a las regiones generadas en el proceso anterior. De este modo, el algoritmo concentra el esfuerzo computacional en los subespacios que han demostrado un mayor potencial, manteniendo al mismo tiempo la capacidad de explorar variabilidad interna dentro de cada región.

Este mecanismo de realimentación convierte el MCSA en un procedimiento iterativo: las regiones potenciales obtenidas en una iteración sirven como dominios de búsqueda para la siguiente. Tras generar las nuevas muestras, se repite el mismo flujo metodológico, evaluación de $f(x)$, identificación de estructuras internas, refinamiento de clusters y definición de nuevas regiones, permitiendo que el proceso avance de forma guiada hacia zonas cada vez más prometedoras del espacio de decisión. Esta dinámica iterativa continúa hasta que se satisface alguno de los criterios de parada definidos: precisión de la media de bloques consecutivos, cobertura máxima del espacio de búsqueda o número máximo de iteraciones.

4. METODOLOGÍA PROPUESTA

4.4.4. Formalización

El procedimiento formal de Explotación puede describirse como sigue:

1. **DBSCAN global inicial.** Se aplica DBSCAN sobre toda la muestra $\mathcal{S}$. Dado que en la práctica suele producir un único cluster (o ninguno útil), se recurre a un refinamiento posterior.
2. **Segmentación en intervalos.** El rango de valores de la función objetivo se divide en M intervalos equidistantes:

$$\mathcal{I} = \{I_1, I_2, \dots, I_M\}, \quad I_m = [f_{\min} + (m-1)\Delta, f_{\min} + m\Delta), \quad (4.4)$$

donde

$$\Delta = \frac{f_{\max} - f_{\min}}{M}. \quad (4.5)$$

3. **Clustering iterativo por intervalo.** Se recorre secuencialmente el conjunto de intervalos $\mathcal{I}$. Para cada I_m , se extrae la submuestra $\mathcal{S}^{(m)} = \{\mathbf{x}_i \in \mathcal{S} : f(\mathbf{x}_i) \in I_m\}$ y se aplica DBSCAN, tal y como se expresa en el diagrama 4.5:

$$\mathcal{C}^{(m)} = \{C_1^{(m)}, \dots, C_{q_m}^{(m)}\}. \quad (4.6)$$

Durante este proceso se registra q_m y se evalúa si el número de clusters aumenta respecto al intervalo anterior.

4. **Determinación del punto de corte.** El proceso se detiene al alcanzar el intervalo I_{m^*} que produce el *máximo número de clusters*, esto es,

$$m^* = \arg \max_m q_m. \quad (4.7)$$

A partir de este punto no se analizan más intervalos ($m > m^*$). Los clusters relevantes quedan definidos como

4. METODOLOGÍA PROPUESTA

$$\mathcal{C}^* = \mathcal{C}^{(m^*)}. \quad (4.8)$$

descartando los intervalos $m > m^*$.

5. **Refinamiento final con K-Means.** Sobre los clusters seleccionados $\mathcal{C}^*$, se aplica K-Means para obtener los centroides más representativos:

$$\mathcal{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k\}, \quad \mathbf{z}_i \in \mathbb{R}^n \quad (4.9)$$

que constituyen las soluciones candidatas para la siguiente iteración del MCSA.

En conclusión, el algoritmo MCSA queda definido a partir de dos fases complementarias. La primera, de *Exploración*, permite recorrer de manera amplia el espacio de búsqueda mediante muestreo estocástico y el registro de configuraciones con desempeño aceptable. La segunda, de *Explotación*, concentra el esfuerzo en las regiones más prometedoras mediante técnicas de agrupamiento y selección de centroides, refinando progresivamente las soluciones candidatas. La articulación de ambas fases establece el diseño final del método, que combina amplitud de búsqueda con capacidad de focalización, y que será validado experimentalmente en los apartados siguientes.

4.5. Evaluación del MCSA

Objetivos de la Evaluación

El presente análisis de rendimiento tiene por objeto la validación intrínseca del Algoritmo MCSA desarrollado. La evaluación se centra en tres métricas fundamentales, diseñadas para cuantificar el comportamiento del algoritmo frente a un conocimiento a priori del espacio de optimización:

- **Calidad de la Solución ($\mathbf{f}_{\text{best}}$):** Se evalúa la proximidad de la mejor

4. METODOLOGÍA PROPUESTA

solución encontrada (f_{best}) respecto al óptimo global conocido (f^*) de la función objetivo.

- **Localización del Óptimo:** Se determina la precisión en la convergencia del vector de decisión $\mathbf{x}_{\text{best}}$ hacia la coordenada $\mathbf{x}^*$ del óptimo global, analizando la precisión espacial en el espacio de búsqueda.
- **Eficiencia de Muestreo ($\mathcal{E}_m$):** En este trabajo, se mide la eficiencia del algoritmo como la relación entre el número total de muestras generadas (N_{MCA}) y el tamaño potencial del espacio de búsqueda ($N_{\text{Exhaustivo}}$), sirviendo como métrica directa de la mejora del MCSA frente a una búsqueda aleatoria simple o exhaustiva.

Justificación: Validación Interna

Es imprescindible señalar que el propósito de esta evaluación no es el **benchmarking** competitivo frente a otros algoritmos de optimización global, como los Algoritmos Genéticos (GA), la Optimización por Enjambre de Partículas (PSO), o métodos basados en Gradiente Estocástico.

La exclusión de dichas comparativas se fundamenta en dos razones metodológicas y prácticas:

1. **Énfasis en el Mecanismo del MCSA:** El objetivo primario de este trabajo es validar la efectividad del mecanismo y del criterios de parada propuesto para el MCSA. La comparación fundamental se establece contra el comportamiento exhaustivo/simple ($N_{\text{Exhaustivo}}$), que representa el peor escenario de muestreo. Con este contraste, se demuestra que la adaptabilidad del MCSA resulta en una mejora sustancial en la efectividad computacional ($\mathcal{E}_m$), ya que el método alcanza soluciones de calidad elevada utilizando un número significativamente menor de muestras.
2. **Requerimientos Específicos de Implementación:** La implementación rigurosa y justa de algoritmos metaheurísticos avanzados, como GA o PSO, requiere una calibración exhaustiva de sus hiperparámetros (por

4. METODOLOGÍA PROPUESTA

ejemplo, tamaño de población, coeficientes de inercia, coeficientes cognitivos y sociales) y un nivel de conocimiento especializado en cada técnica. Realizar únicamente una comparativa superficial, sin ajustar adecuadamente estos parámetros ni asegurar la correcta implementación, podría introducir sesgos no deseados y dar lugar a conclusiones incorrectas sobre el desempeño relativo de los algoritmos. Por este motivo, en el presente trabajo no se ha realizado una comparación directa con dichas metaheurísticas.

En resumen, la evaluación se concibe como una referencia interna que certifica la funcionalidad y la ventaja computacional del MCSA respecto a la Exploración bruta, cumpliendo el objetivo de tesis de proponer un mecanismo de optimización funcional y auto-regulado. Los resultados de esta validación interna también sirven para justificar y medir el impacto de las **mejoras incrementales** implementadas en el algoritmo tras identificar rendimientos subóptimos en pruebas preliminares.

Validación mediante funciones clásicas de benchmark

Para evaluar la eficacia inicial del MCSA, se seleccionó un conjunto de funciones matemáticas clásicas utilizadas comúnmente en optimización: Ackley, Sphere, Rastrigin, Griewank, Hölder Table y Michalewicz. Estas funciones fueron ejecutadas 30 veces por función, utilizando distintas semillas aleatorias para garantizar consistencia en los resultados. El algoritmo de Montecarlo opera sobre un espacio de decisión $\mathcal{D} \subset \mathbb{R}^n$ para minimizar una función objetivo $f(\mathbf{x}) : \mathcal{D} \rightarrow \mathbb{R}$. La naturaleza adaptativa del método radica en la contracción y reorientación iterativa del espacio de muestreo $\mathcal{S}^{(k)} \subset \mathcal{D}$.

La metodología de Optimización por MCSA se implementa para equilibrar la necesidad de Exploración inicial con la Explotación localizada. A diferencia de la búsqueda aleatoria pura, con esta estrategia se garantiza una convergencia acelerada al actualizar, en cada iteración k , la región de muestreo $\mathcal{S}^{(k)}$ y su punto central, $\mathbf{c}^{(k)}$, al mejor punto encontrado en la iteración previa, $\mathbf{x}_{\text{best}}^{(k-1)}$. La contracción del rango de búsqueda, definida por un factor $\rho \in (0, 1)$, fuerza un acotamiento progresivo en las regiones prometedoras. La detención del muestreo

4. METODOLOGÍA PROPUESTA

se controla mediante los criterios de precisión de la media funcional $\bar{f}$ y cobertura máxima del espacio, asegurando que el proceso se interrumpa únicamente cuando la mejora marginal en la calidad promedio de los puntos no justifica nuevos cálculos. Este enfoque proporciona un compromiso equilibrado entre precisión y eficiencia computacional durante el refinamiento del mínimo.

4.5.1. Resultados Experimentales

A continuación, se presentan los resultados de la validación experimental realizada. El objetivo es evaluar su capacidad para explorar eficazmente el espacio de búsqueda, identificar regiones prometedoras y refinar la solución. Para ello, se empleó un conjunto representativo de funciones benchmark ampliamente utilizadas en la literatura de optimización, ejecutando cada experimento 30 veces con semillas independientes con el fin de analizar la robustez del heurístico. Los resultados obtenidos permiten valorar tanto la calidad de las soluciones alcanzadas, medida como el porcentaje de acercamiento al óptimo global, como la eficiencia del proceso de muestreo con respecto al tamaño real del problema.

Como se observa en la Tabla 4.1, el MCSA alcanza soluciones de alta calidad en todos los casos, con un porcentaje de aproximación al óptimo mayor al 96 % y, en varios casos, logra exactamente el valor óptimo. La relación entre número de muestras y tamaño del problema indica que el algoritmo logra identificar soluciones prometedoras de manera eficiente, sin necesidad de recorrer exhaustivamente todo el espacio de búsqueda.

Tabla 4.1: Resultados de la validación utilizando funciones clásicas de *benchmark*.

Función	$f(x^*)$	Óptimo global	Psize	Muestras	Muestras vs Psize (%)	Mejor resultado	Localización mejor resultado	Calidad (%)
Ackley	0	(0, ..., 0)	1,048,576	87,000	8.3	0.0044	(0.00104, 0.00114)	100.00
Sphere	0	(0, ..., 0)	1,048,576	24,300	2.3	0.0238	(0.11025, -0.10818)	98.97
Hölder Table	-19.85	(±8.05502, ±9.66459)	1,048,576	116,800	11.1	-19.2084	(±8.05521, ±9.66407)	96.72
Levi N.13	0	(1, 1)	1,048,576	11,200	1.1	0.0194	(1.00392, 0.88739)	98.98
Griewank	0	(0, ..., 0)	1,440,000	48,605	3.4	0.0235	(0.0, -9.0)	98.97
XinSheYang	0	(0, ..., 0)	250,000	37,601	15.0	0.0000	(0.0, 0.0)	100.00

La *calidad de la solución* se define en este estudio como el porcentaje de acercamiento al óptimo conocido; valores cercanos a 100 % indican que la mejor

4. METODOLOGÍA PROPUESTA

solución encontrada está muy próxima al óptimo teórico. Además de la calidad, debe considerarse la relación entre el número de evaluaciones ($\#$ Samples) y el tamaño efectivo del problema ($Psize$): un bajo porcentaje de cobertura ($\#$ Samples / $Psize$) reduce la probabilidad de explorar exhaustivamente el espacio y obliga al heurístico a equilibrar Exploración y Explotación.

Para complementar los resultados cuantitativos presentados, en las Figuras 4.10 y 4.11 se ilustra gráficamente cómo el MCSA explora el espacio de soluciones y refina los resultados mediante clusterización hasta aproximarse a una solución de calidad. En las mismas figuras se observa que el MCSA inicia con un muestreo aleatorio global, generando un mapa disperso de soluciones potenciales en el espacio de búsqueda. Posteriormente, se aplica la clusterización en dos pasos para identificar regiones densamente pobladas de soluciones prometedoras, concentrando así la búsqueda en zonas de alto interés y facilitando la obtención de soluciones cercanas al óptimo. Estas visualizaciones confirman la eficacia del algoritmo y respaldan los resultados cuantitativos presentados en la anterior Tabla 4.1.

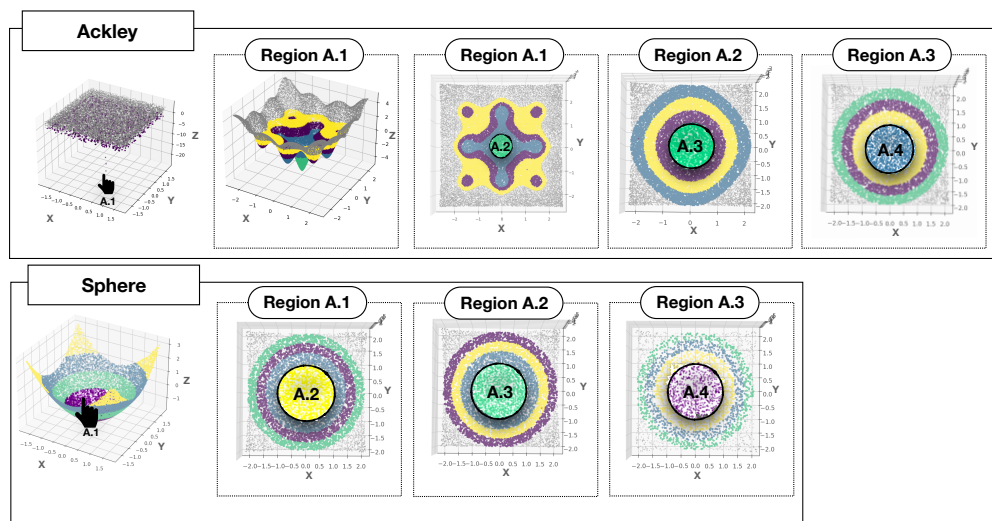


Figura 4.10: Exploración del espacio de soluciones mediante muestreo Montecarlo y refinamiento de soluciones sobre las funciones: Ackley y Sphere.

Cabe aclarar, que respecto a la visualización de la función Ackley se distinguen dos regiones principales, etiquetadas como A.1 y A.2. La etiqueta A.1 indica la región correspondiente a la vista actual, mientras que A.2 corresponde a otra región

4. METODOLOGÍA PROPUESTA

visible desde esa perspectiva. Es decir, aunque el rótulo A.1 señala la ubicación principal, la proyección permite observar simultáneamente otras regiones, como A.2, para ilustrar la relación entre diferentes óptimos locales. De manera análoga, en el segundo conjunto de imágenes que muestra las funciones Xin She Yang y Levy N.13, la vista en 2D mantiene la referencia a la región A.1, pero permite visualizar la distribución de otras regiones, incluyendo A.2, con respecto a la zona principal.

Nuestros resultados muestran que para funciones unimodales y separables (p. ej., Sphere) el algoritmo alcanza altas calidades (99 %) incluso con una cobertura reducida, lo que evidencia una buena capacidad de convergencia local. En funciones altamente multimodales como Hölder Table, pese a una cobertura relativamente alta (11.1 %), la mejor solución observada presenta un residuo apreciable frente al óptimo teórico (calidad 96.7 %), lo que sugiere la necesidad de técnicas de *niching* o intensificación local para localizar todos los mínimos globales con precisión. Para funciones estocásticas (Xin-She Yang) y problemas con óptimo en cero (Ackley, Griewank), las aproximaciones absolutas reportadas (p. ej. residuo 0.0044 en Ackley) indican que el heurístico logra converger muy cerca del óptimo en al menos algunas semillas.

Resultados de la Validación mediante funciones benchmark del CEC2017

Para evaluar la robustez y escalabilidad del MCSA, se seleccionó un conjunto de funciones del benchmark CEC2017. Estas funciones presentan características más desafiantes, incluyendo desplazamiento, rotación, no separabilidad y alta dimensionalidad, lo que permite poner a prueba la capacidad del algoritmo de converger en condiciones complejas y realistas.

En este conjunto de pruebas, cada función fue ejecutada más de 200 veces con semillas aleatorias diferentes. Esto permitió analizar en detalle el algoritmo, la consistencia en la convergencia hacia el óptimo y la capacidad de manejo de diferentes dominios de búsqueda. Las ejecuciones se realizaron en el cluster del

4. METODOLOGÍA PROPUESTA

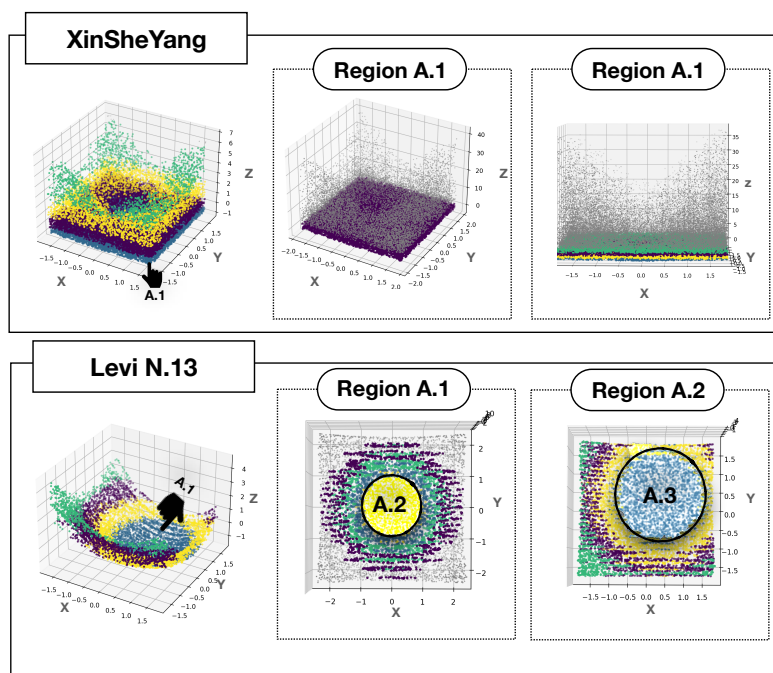


Figura 4.11: Aplicación de la metodología sobre las funciones: XinSheYang y Levi N.13.

departamento DELL AMD Opteron™ 6200 a 1.60 Hz, con 1 nodo (64 cores) y 64 GB de RAM, bajo Linux versión 2.6.

4.5.2. Análisis y Discusión de los Resultados

Como se aprecia en la Tabla 4.2, el MCSA mantiene un desempeño sólido y consistente incluso en escenarios de alta complejidad. Las funciones del CEC2017 suponen un reto debido a la presencia de desplazamientos, rotaciones, no separabilidad y elevada dimensionalidad. Aun así, el algoritmo logra aproximarse al óptimo con valores medios competitivos y una variabilidad controlada entre ejecuciones.

Un aspecto clave evaluado en esta tabla es la robustez, medida como el número de ejecuciones en las que el algoritmo alcanzó el mejor valor registrado para cada función. Este indicador resume el comportamiento del MCSA frente a las variaciones inherentes al muestreo estocástico.

4. METODOLOGÍA PROPUESTA

Tabla 4.2: Resultados de las pruebas con funciones benchmark del CEC2017.

Benchmark	Nº dim.	Dominio	Tamaño del problema	Media de muestras	% respecto al tamaño	Solución óptima	Mejor resultado MCSA
F2: Zakharov desplazada y rotada	2	$[-100,100]^2$	4.00E+08	33188	0.01 %	200	200.00
F3: Rosenbrock desplazada y rotada	2	$[-100,100]^2$	4.00E+08	34790	0.01 %	300	300.00
F3: Rosenbrock desplazada y rotada	10	$[-100,100]^{10}$	1.02E+43	44130	0.00 %	300	300.59
F4: Rastrigin desplazada y rotada	2	$[-100,100]^2$	4.00E+08	25630	0.01 %	400	400.12
F4: Rastrigin desplazada y rotada	10	$[-100,100]^{10}$	1.02E+43	50348	0.00 %	400	427.06
F5: Schaffer F7 desplazada y rotada	2	$[-100,100]^2$	4.00E+08	4000	0.00 %	500	500.00
F5: Schaffer F7 desplazada y rotada	10	$[-50,50]^{10}$	1.00E+40	4510	0.00 %	500	500.00
F6: Lunacek Bi-Rastrigin desplazada y rotada	10	$[-100,100]^{10}$	1.02E+33	32360	0.00 %	600	751.40
F6: Lunacek Bi-Rastrigin desplazada y rotada	10	$[-50,50]^{10}$	1.00E+40	14746	0.00 %	600	791.20
F7: Rastrigin no continua desplazada y rotada	10	$[-50,50]^{10}$	1.00E+40	6629	0.00 %	700	700.00
F8: Levy desplazada y rotada	10	$[-50,50]^{10}$	1.00E+40	15800	0.00 %	800	800.10
F9: Schwefel desplazada y rotada	10	$[-50,50]^{10}$	1.00E+40	9046	0.00 %	900	1124.75

La evaluación del MCSA mediante funciones clásicas y benchmarks del CEC2017 permite extraer conclusiones importantes sobre su desempeño y aplicabilidad en problemas de optimización de distinta naturaleza. En primer lugar, los resultados obtenidos con funciones clásicas muestran que el algoritmo es capaz de aproximarse al óptimo global con un alto grado de precisión (generalmente superior al 96 %) utilizando un número de muestras considerablemente inferior al que requeriría un enfoque exhaustivo. Esto confirma la eficiencia del método en términos de Exploración del espacio de búsqueda y localización de soluciones prometedoras, aun en problemas con múltiples óptimos locales o interacciones complejas entre variables.

Cabe destacar que estas pruebas iniciales no se realizaron para comparar con otras metaheurísticas, sino para establecer una referencia interna que permita validar mejoras implementadas en el algoritmo y su comportamiento frente a distintos tipos de funciones.

Del análisis detallado de los resultados se desprenden tres conclusiones principales:

- **Consistencia en funciones unimodales y multimodales suaves:** el

4. METODOLOGÍA PROPUESTA

MCSA presenta un alto grado de convergencia hacia el óptimo global, con medias muy próximas a los valores óptimos conocidos y con una robustez elevada, lo que indica que el algoritmo tiende a repetir el mejor resultado en un número significativo de ejecuciones.

- **Resiliencia frente a funciones rotadas, desplazadas y altamente no separables:** en aquellos problemas donde la estructura del paisaje es más difícil de explorar, el algoritmo mantiene una buena robustez, mostrando medias cercanas al mejor valor alcanzado. Incluso en funciones particularmente adversas, el MCSA es capaz de identificar regiones prometedoras y concentrar progresivamente el muestreo en torno a ellas.
- **Equilibrio entre Exploración y Explotación:** la estrategia basada en muestreo Montecarlo guiado por clustering permite cubrir de forma eficiente zonas amplias del espacio de búsqueda sin perder la capacidad de intensificación necesaria para afinar soluciones. Esto se refleja en la distribución de los resultados: en la mayoría de las funciones, los valores se encuentran fuertemente concentrados alrededor de la media, demostrando un comportamiento preciso y sin dispersiones extremas.

Los resultados muestran que el algoritmo mantiene un desempeño sólido incluso en condiciones adversas, convergiendo de manera confiable hacia soluciones óptimas o cercanas al óptimo, lo que evidencia su capacidad para adaptarse a espacios de búsqueda extensos y complejos.

La combinación de estos dos conjuntos de pruebas, clásicas y CEC2017, permite concluir que el MCSA ofrece una herramienta heurística robusta, escalable y flexible, capaz de abordar tanto problemas de baja y media dimensionalidad como escenarios altamente complejos de optimización combinatoria y continua. Además, los resultados destacan la ventaja de su estructura modular, que integra de manera efectiva fases de Exploración y Explotación basadas en muestreo Montecarlo y clustering, proporcionando un equilibrio entre cobertura global del espacio de búsqueda y concentración en regiones prometedoras. En síntesis, la evidencia obtenida respalda la aplicabilidad práctica del MCSA en distintos dominios,

4. METODOLOGÍA PROPUESTA

confirmando que el método no solo es eficiente para encontrar soluciones de alta calidad, sino también confiable y estable frente a variaciones en las condiciones de búsqueda.

Finalmente, en el Apéndice A.2 se incluyen los *boxplots* correspondientes a estas funciones, los cuales permiten visualizar gráficamente el grado de concentración de los resultados, así como la presencia de posibles valores atípicos. Estos gráficos refuerzan la conclusión general de que el MCSA no solo obtiene soluciones de calidad, sino que lo hace de forma estable, repetida y predecible a lo largo de múltiples ejecuciones independientes.

4.6. Conclusiones del Capítulo

En este capítulo se presentó de manera integral el MCSA, abarcando desde su conceptualización y parametrización general hasta su funcionamiento detallado y su evaluación mediante benchmarks clásicos y del CEC2017. Se destacó que la estructura modular del algoritmo, que combina las fases operativas de Exploración mediante muestreo Montecarlo, y la Explotación basada en *clustering*, lo que proporciona un equilibrio eficaz entre cobertura global del espacio de búsqueda y concentración en regiones prometedoras. La evaluación a través de múltiples ejecuciones independientes confirmó la robustez y capacidad de convergencia del MCSA frente a funciones con múltiples óptimos locales, interacciones complejas entre variables y alta dimensionalidad.

La Fase de Exploración, basada en muestreo Montecarlo, permite identificar de manera eficiente las regiones del espacio donde es probable localizar óptimos globales o soluciones cercanas al óptimo. El criterio de parada implementado evita un muestreo excesivo, garantizando que la Exploración sea suficiente para cubrir el espacio de manera representativa, pero sin incurrir en un gasto computacional innecesario. Esto se refleja en los mapas de Montecarlo obtenidos, donde se observa la concentración progresiva de muestras en los valles de la función objetivo, confirmando la capacidad del MCSA para detectar zonas prometedoras con un número de evaluaciones significativamente menor al requerido por un enfoque

4. METODOLOGÍA PROPUESTA

exhaustivo.

La Fase de Explotación, mediante clustering y refinamiento local, permite intensificar la búsqueda en las regiones identificadas como prometedoras. Esta estructura modular asegura que el método combine efectivamente la *Exploración global* con la *Explotación local*, evitando quedarse atrapado en óptimos locales y facilitando la convergencia hacia soluciones de alta calidad. La evaluación mediante múltiples ejecuciones independientes (con distintas semillas) evidenció la robustez del MCSA: en la mayoría de las funciones, el algoritmo logra converger de manera consistente hacia el óptimo o soluciones cercanas, mostrando baja dispersión en los resultados y alta repetibilidad de las mejores soluciones.

Los resultados obtenidos evidencian que el método alcanza soluciones de alta calidad utilizando un número de muestras significativamente inferior al requerido por métodos exhaustivos, garantizando una *efectividad computacional* elevada sin comprometer la precisión de la búsqueda. Asimismo, la comparación entre benchmarks clásicos y los del CEC2017 demuestra que el MCSA es escalable, flexible y aplicable a problemas de distinta naturaleza, incluyendo aquellos con elevada complejidad combinatoria y restricciones reales.

La métrica $\mathcal{E}_m$ se define como la relación entre el número total de muestras generadas (N_{MCA}) y el tamaño potencial del espacio de búsqueda ($N_{\text{Exhaustivo}}$), sirviendo como indicador de la utilización efectiva de los recursos computacionales. Un valor alto de $\mathcal{E}_m$ refleja que el algoritmo aprovecha de manera eficiente las evaluaciones de la función objetivo, maximizando la probabilidad de acercamiento al óptimo por unidad de cómputo. Esta propiedad es especialmente relevante en problemas de alta dimensionalidad y complejidad estructural, donde los métodos exhaustivos serían inviables.

En conjunto, estas evidencias consolidan al MCSA como una herramienta heurística para la optimización de problemas complejos y multidimensionales, que no solo es capaz de alcanzar soluciones de gran calidad, sino que lo hace de manera eficiente y estable, ofreciendo una base sólida para su aplicación práctica y futuras extensiones. El MCSA se sustenta en tres pilares metodológicos fundamentales:

4. METODOLOGÍA PROPUESTA

1. **Modularidad de fases:** la separación clara entre Exploración y Explotación asegura que cada fase cumpla su rol sin interferencias, favoreciendo la detección de regiones prometedoras y el refinamiento de soluciones.
2. **Criterio de parada:** el muestreo se detiene de manera adaptativa cuando la convergencia de las mejores muestras indica que las mejoras marginales son insignificantes. Esto permite controlar la duración de la Fase de Exploración, evitando sobrecostos computacionales y favoreciendo la eficiencia del muestreo.
3. **Robustez frente a la aleatoriedad:** la ejecución de múltiples corridas independientes demuestra que el algoritmo mantiene alta consistencia en los resultados, reflejando un comportamiento reproducible y confiable frente a la variabilidad inherente del muestreo estocástico.

Finalmente, la combinación de estas características explica por qué el MCSA logra un equilibrio adecuado entre cobertura del espacio de búsqueda, eficiencia computacional y calidad de las soluciones obtenidas, validando su propuesta metodológica y consolidando su aplicabilidad en el análisis y resolución de problemas de optimización complejos.

Glosario del Capítulo

Se presenta el glosario del presente Capítulo en la Tabla 4.3 a continuación, que incluye los términos y símbolos más relevantes utilizados en esta sección. Este recurso tiene como objetivo facilitar la comprensión de la terminología empleada y servir de referencia rápida durante la lectura.

Tabla 4.3: Glosario del Capítulo

Término	Definición
MCSA	Montecarlo Clustering Search Algorithm. Procedimiento iterativo basado en muestreo, reducción del espacio de búsqueda y selección de regiones prometedoras mediante clustering.
Fase de Exploración	Etapas del MCSA dedicada a muestrear el espacio de búsqueda Ω mediante bloques aleatorios y construir un mapa Montecarlo con muestras aptas.
Fase de Explotación	Etapas donde se aplica clustering (DBSCAN y K-Means) sobre el mapa Montecarlo para identificar regiones densas y extraer candidatos representativos.
Criterio de parada	Condición que determina la detención de la Fase de Exploración del MCSA. Se verifica si se cumplen las condiciones de precisión de la media funcional o de cobertura máxima del espacio, evitando cálculos innecesarios y optimizando la eficiencia del muestreo.

Continúa en la siguiente página

4. METODOLOGÍA PROPUESTA

Glosario del Capítulo (continuación)

Término	Definición
Criterio de precisión	Condición basada en la comparación de la media de los valores de la función objetivo entre bloques consecutivos de muestras. El muestreo se considera convergente si la variación de la media es menor que un umbral ϵ , garantizando que la mejora marginal no justifica nuevas evaluaciones.
Criterio de cobertura	Condición que limita el porcentaje máximo del espacio de búsqueda que puede ser muestreado, típicamente $P_{\text{máx}}$. Permite equilibrar exhaustividad y eficiencia computacional, asegurando que la exploración no exceda un límite predefinido.
$\Omega \subseteq \mathbb{R}^n$	Espacio de búsqueda factible, definido por todas las restricciones del problema.
$U(\Omega)$	Distribución uniforme sobre el espacio de búsqueda Ω , utilizada para generar las muestras iniciales.
$\mathcal{S}$	Conjunto de soluciones candidatas generadas en la Fase de Exploración.
$\mathbf{x}^{(i)}$	Solución individual del conjunto de muestras $\mathcal{S}$.
$\mathcal{S}_F$	Subconjunto factible de muestras que cumplen restricciones del problema.
$f(\mathbf{x})$	Valor de la función objetivo asociado a la muestra $\mathbf{x}$.
Muestra apta	Muestra cuyo valor $f(\mathbf{x})$ supera el umbral de aptitud durante la Fase de Exploración, según su valor objetivo.
$\mathcal{M}$	Mapa Montecarlo compuesto por las muestras aptas.
B	Tamaño del bloque de muestreo en la Fase de Exploración.

Continúa en la siguiente página

4. METODOLOGÍA PROPUESTA

Glosario del Capítulo (continuación)

Término	Definición
$P_{\text{máx}}$	Porcentaje máximo del espacio a cubrir mediante muestreo.
ϵ	Tolerancia empleada para determinar convergencia del muestreo.
τ	Umbral de densidad que se utiliza para determinar qué intervalos se retienen en la fase de pre-clustering.
DBSCAN	Algoritmo de clustering basado en densidad utilizado para identificar grupos en subconjuntos de muestras según su valor objetivo.
K-Means	Algoritmo de clustering basado en centroides aplicado como refinamiento sobre los clusters seleccionados.
ϵ (o eps)	Radio del vecindario en DBSCAN.
min_samples	Número mínimo de puntos requeridos para que un punto sea considerado núcleo (core point).
minPts	Variante de la literatura equivalente al parámetro <code>min_samples</code> , utilizado para definir el número mínimo de vecinos requeridos para que un punto sea considerado núcleo (core point).
core point	Punto con densidad suficiente en su vecindario, es decir, con al menos <code>min_samples</code> vecinos dentro del radio definido.
border point	Punto que no cumple la condición de núcleo pero se encuentra dentro del vecindario de un core point, formando parte de su cluster.

Continúa en la siguiente página

4. METODOLOGÍA PROPUESTA

Glosario del Capítulo (continuación)

Término	Definición
noise point	Punto marcado como ruido por DBSCAN, ya que no pertenece a ningún cluster y no está en el vecindario de un core point.
Cluster	Conjunto de puntos agrupados por DBSCAN o K-Means.
$\mathcal{J}$	Conjunto de intervalos en los que se divide el rango de $f(\mathbf{x})$.
I_m	Intervalo m -ésimo definido por $f_{\min}$, $f_{\max}$ y Δ .
Δ	Amplitud del intervalo en la segmentación: $\Delta = (f_{\max} - f_{\min})/M$.
$\mathcal{S}^{(m)}$	Subconjunto de muestras cuyas evaluaciones pertenecen al intervalo I_m .
$\mathcal{C}^{(m)}$	Conjunto de clusters encontrados mediante DBSCAN en el intervalo I_m .
q_m	Número de clusters detectados en el intervalo I_m .
m^*	Índice del intervalo que produce el máximo número de clusters.
$\mathcal{C}^*$	Clusters seleccionados tras aplicar el corte en el intervalo m^* .
$\mathcal{Z}$	Conjunto de centroides finales extraídos mediante K-Means.
$\mathbf{z}_i$	Centroides representativos de las regiones más prometedoras del espacio.
Factible / No factible	Clasificación de una solución dependiendo de si cumple o no las restricciones del problema.

Continúa en la siguiente página

4. METODOLOGÍA PROPUESTA

Glosario del Capítulo (continuación)

Término	Definición
f_{best}	Mejor valor encontrado por el MCSA.
$\mathbf{x}_{\text{best}}$	Coordenadas del mejor punto encontrado.
f^*	Óptimo global conocido de la función objetivo.
$\mathcal{E}_m$	Eficiencia de muestreo: proporción de muestras generadas respecto al tamaño del espacio de búsqueda exhaustivo.
Ackley	Función multimodal con un óptimo global en $(0, \dots, 0)$. Usada para evaluar Exploración en espacios ondulados.
Sphere	Función convexa unimodal con un único óptimo en $(0, \dots, 0)$.
Hölder Table	Función altamente multimodal con múltiples mínimos locales y dos simetrías en el óptimo global.
Michalewicz	Función con fuertes interacciones entre variables y pozos estrechos de difícil detección para K-Means.
Griewank	Función multimodal suave que combina término cuadrático y cosenos oscilantes.
Zakharov (desplazada/rotada)	Función probada en versiones transformadas para aumentar dificultad.
Rosenbrock	Función con valle estrecho y curvatura suave, probada en versiones rotadas y de alta dimensión.
Rastrigin	Función altamente multimodal, usada también en versiones rotadas y no continuas.
Schaffer F7	Función con topología irregular que requiere precisión en la Exploración.

Continúa en la siguiente página

4. METODOLOGÍA PROPUESTA

Glosario del Capítulo (continuación)

Término	Definición
Lunacek Bi-Rastrigin	Función engañosa con dos pozos principales y comportamientos no convexos.
Levy N.13	Función suave con óptimo en $(1, 1)$.
Schwefel	Función globalmente compleja que contiene regiones donde las soluciones óptimas están alejadas del origen.

Capítulo 5

Aplicación de la Metodología y del Modelo propuesto: El problema de la mochila

En este capítulo se presenta la aplicación práctica de la metodología desarrollada a lo largo del trabajo, con el objetivo de validar su alcance y efectividad en un problema clásico de optimización combinatoria: el Problema de la Mochila Binaria (*Binary Knapsack Problem*, *KP01*). Este problema constituye un banco de pruebas ampliamente reconocido, tanto por su complejidad inherente (NP-completo) como por su relevancia en escenarios donde deben tomarse decisiones bajo restricciones, desde la planificación de recursos en almacenes y la distribución de cargas hasta la asignación eficiente de inversiones o la optimización de cortes en materiales industriales.

La naturaleza discreta y combinatoria del KP01 lo convierte en un caso de estudio ideal para evaluar la aplicabilidad de algoritmos metaheurísticos como MCSA, que buscan soluciones de alta calidad sin necesidad de explorar exhaustivamente todas las combinaciones posibles. Para abordar este problema se propone aplicar el MCSA, diseñado para equilibrar sus dos fases complementarias: Exploración y Explotación. Dicha aplicación permite poner a prueba el MCSA en un entorno

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

exigente y, al mismo tiempo, compararlo con los métodos convencionales utilizados en optimización discreta.

5.1. Justificación de la propuesta

En la optimización de problemas complejos y multidimensionales, los métodos exactos presentan limitaciones importantes, tanto por su costo computacional como por su dificultad para adaptarse a restricciones múltiples o estructuras no lineales. En este contexto, las metaheurísticas han demostrado ser técnicas eficaces, capaces de proporcionar soluciones aproximadas de alta calidad en tiempos razonables. Entre ellas, los algoritmos que equilibran Exploración y Explotación destacan por su capacidad para recorrer amplios espacios de búsqueda y, al mismo tiempo, concentrarse en regiones prometedoras.

MCSA se desarrolla precisamente con este objetivo. El propósito central de esta propuesta es demostrar la aplicabilidad y efectividad del MCSA como herramienta robusta para la resolución de problemas de optimización combinatoria de alta complejidad. Para ello, se seleccionó como caso de estudio el Problema de la Mochila Binaria, un problema clásico NP-completo que representa una instancia paradigmática de optimización discreta con restricciones.

Su Fase de Exploración utiliza muestreo aleatorio uniforme (Montecarlo) para generar un conjunto inicial de soluciones dispersas, garantizando cobertura del espacio sin conocimiento previo sobre la función objetivo. Posteriormente, la Fase de Explotación aplica clustering no supervisado: primero mediante DBSCAN para identificar regiones densas y eliminar ruido, y luego con K-means para refinar los centroides dentro de dichas regiones. Este procedimiento permite reducir el espacio de búsqueda y focalizar los recursos computacionales en subespacios con alto potencial de soluciones óptimas o cuasi-óptimas.

La naturaleza iterativa y adaptativa del MCSA constituye otra fortaleza clave. Tras identificar los K-clusters más prometedores, se realiza una nueva Exploración localizada, afinando progresivamente la búsqueda en base al conocimiento

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

acumulado. Este ciclo continuo de Exploración y Explotación, guiado por clustering en dos pasos, permite que el algoritmo se adapte a la topología del espacio de búsqueda, incrementando la precisión de las soluciones y mejorando la probabilidad de alcanzar el óptimo global.

En las secciones siguientes, se aplicará el MCSA al problema de la mochila (*KP01*), demostrando cómo esta metodología permite abordar de manera eficiente problemas combinatorios de alta dimensión, donde la enumeración exhaustiva de todas las soluciones es inviable y la Exploración inteligente del espacio es esencial.

5.2. Objetivo metodológico de la propuesta

El KP01 consiste en seleccionar un subconjunto de elementos que maximice el valor total sin superar una capacidad límite sobre otro atributo, generalmente el peso. Formalmente, el espacio de búsqueda está constituido por 2^n posibles soluciones, lo que convierte al KP01 en un banco de pruebas ideal para evaluar algoritmos metaheurísticos capaces de explorar espacios de gran escala.

La estrategia propuesta mediante MCSA se basa en una representación vectorial del espacio de búsqueda, donde cada dimensión corresponde a una variable binaria asociada a la inclusión o exclusión de un ítem. Esto permite combinar de manera efectiva la dispersión inicial de soluciones en la Fase de Exploración con la concentración posterior en subespacios relevantes durante la Fase de Explotación. El aspecto distintivo de la metodología radica en el uso de clustering en dos pasos sobre soluciones discretas para guiar la intensificación de la búsqueda, equilibrando diversidad y convergencia de manera adaptativa.

Para validar la eficacia de la metodología, se realizaron experimentos sobre instancias del KP01 de tamaños crecientes, incluyendo hasta 2000 variables binarias. Los resultados muestran que el algoritmo alcanza en promedio más del 90% del valor óptimo conocido, con buena eficiencia en tiempo de cómputo y escalabilidad para problemas grandes. Además, las pruebas con funciones benchmark estándar y/o con algún componente de rotación o traslación, indican

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

que el MCSA logra la solución óptima en el 60 % de las ejecuciones y soluciones cercanas al óptimo (desviaciones menores al 5 %) en el restante 40 %, incluso en casos de alta multimodalidad (existencia de múltiples óptimos locales) y complejidad topológica.

En síntesis, el objetivo de la propuesta no se limita a resolver el KP01 clásico de un solo objetivo, sino que busca establecer una metodología generalizable capaz de manejar tanto problemas monoobjetivo como multiobjetivo, combinando muestreo estocástico y análisis estructural mediante clustering como metodología combinada para problemas de optimización de gran escala.

5.3. Antecedentes y Trabajos Relacionados

En los últimos años, la creciente complejidad y dimensionalidad de los problemas de optimización ha incrementado la necesidad de algoritmos capaces de proporcionar soluciones de alta calidad de manera eficiente. En espacios de búsqueda extensos, los métodos exactos, los algoritmos iterativos convencionales o las heurísticas simples resultan ineficaces debido a su elevado costo computacional y a la imposibilidad de explorar de manera completa todas las posibles configuraciones. En este contexto, los algoritmos metaheurísticos se presentan como una alternativa viable, ya que permiten encontrar soluciones razonablemente buenas sin requerir un recorrido exhaustivo del espacio de búsqueda [15].

Entre los métodos probabilísticos, los métodos basados en distribuciones estocásticas, como Montecarlo, ofrecen ventajas significativas en términos de flexibilidad y costo computacional. Estos métodos utilizan generación de muestras aleatorias para aproximar soluciones, lo que resulta especialmente útil cuando las técnicas determinísticas tradicionales son inviables, como se reconoce en [102, 11]. Dentro de este marco, se incluyen algoritmos de búsqueda aleatoria, métodos de comparación estocástica [37], y particiones anidadas con implementaciones específicas [86], entre otros. La característica común de estos algoritmos es su capacidad de convergencia global bajo ciertas condiciones, dado que cada solución factible puede ser evaluada sistemáticamente para guiar la selección de candidatos

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

prometedores.

Otros métodos aplican algoritmos basados en similitud o metaheurísticos para problemas de optimización de alta dimensionalidad, evaluados mediante funciones de prueba de gran escala [27]. Sin embargo, estos métodos presentan el riesgo de quedar atrapados en óptimos locales. Para abordar problemas de optimización global, los algoritmos metaheurísticos que integran Exploración y Explotación se muestran como las estrategias más efectivas. Entre ellos, los algoritmos evolutivos han sido ampliamente estudiados por su capacidad de equilibrar la cobertura global y la intensificación local de la búsqueda [98, 105, 26, 92]. Este tipo de algoritmos se inspiran en procesos biológicos naturales, como reproducción, mutación, recombinación y selección. Uno de los ejemplos más representativos incluyen el Algoritmo Genético (GA) [78], que ha sido ampliamente aplicado en problemas combinatorios utilizando operadores mutacionales y de cruce, y sus múltiples variantes. Además, en la última década se han propuesto numerosos algoritmos, también de inspiración biológica o física, como estrategias de enjambre, técnicas basados en física, estrategias evolutivas [23] y programación evolutiva [55]. Para una revisión detallada del estado del arte en algoritmos de optimización, se puede consultar [3, 74].

Por otro lado, los algoritmos de análisis de clusters representan un componente central del análisis exploratorio de datos. De manera general, pueden clasificarse en métodos basados en particiones, jerárquicos, basados en densidad y basados en particiones (celdas). Entre los más utilizados se encuentran K-means [36] y DBSCAN [80, 64], tal y como se han descrito anteriormente. En resumen, K-means es un algoritmo de partición que busca minimizar la inercia dentro de los clusters, asumiendo que estos son convexos e isotrópicos. Por su parte, DBSCAN se fundamenta en principios de densidad, identificando regiones de alta concentración de puntos separadas por zonas de baja densidad. De esta forma, permite descubrir clusters de forma arbitraria y es robusto frente a datos atípicos, útil para problemas de agrupamiento espacial o en contextos donde la distribución de las soluciones no es homogénea [31].

En síntesis, la combinación de técnicas probabilísticas con métodos de

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

agrupamiento puede mejorar significativamente la eficiencia de la búsqueda en problemas complejos, proporcionando tanto cobertura global del espacio de búsqueda como capacidad de intensificación en regiones prometedoras. Esta integración constituye la base conceptual sobre la que se desarrolla la metodología propuesta en esta tesis, que combina muestreo Montecarlo con un esquema de clustering en dos pasos para abordar problemas de optimización de alta dimensionalidad y múltiples objetivos.

5.4. Formulación: el Problema de la Mochila Binaria (*KP01*)

El KP01 es uno de los problemas clásicos más representativos dentro de la optimización combinatoria. Consiste en seleccionar un subconjunto de elementos, cada uno con un peso y un valor asociados, de modo tal que el valor total sea maximizado sin que el peso total exceda la capacidad de la mochila.

5.4.1. Definición formal del KP01

Sea un conjunto de n elementos, cada uno con un valor $v_i \in \mathbb{R}^+$ y un peso $w_i \in \mathbb{R}^+$. Se define una variable binaria $x_i \in \{0, 1\}$ para cada elemento, donde $x_i = 1$ indica que el elemento i es incluido en la mochila y $x_i = 0$ indica que no lo es. El objetivo es:

$$\text{Maximizar } Z = \sum_{i=1}^n v_i x_i \quad (5.1)$$

Sujeto a:

$$\sum_{i=1}^n w_i x_i \leq W \quad (5.2)$$

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

$$x_i \in \{0, 1\}, \quad \forall i = 1, \dots, n \quad (5.3)$$

Donde W es la capacidad máxima de la mochila. La dificultad del problema radica en el número exponencial de posibles combinaciones (exactamente 2^n), lo que lo convierte en intratable para grandes valores de n .

KP01 como problema NP-Completo

El KP01 fue identificado por Richard Karp en su seminal trabajo de 1972 [54], en el cual se presentaron 21 problemas NP-Completo fundamentales. Desde entonces, ha servido como problema canónico para probar nuevas técnicas de resolución tanto exactas como aproximadas. Un problema se clasifica como NP-Completo cuando no se conoce un algoritmo que lo resuelva en tiempo polinomial y cuando cualquier problema en NP puede reducirse a él en tiempo polinomial.

5.4.2. Métodos clásicos de resolución y Herramientas modernas de resolución

Para valores pequeños de n , es posible aplicar métodos exactos, que garantizan la obtención de la solución óptima. Entre ellos, destacan:

- **Algoritmos de programación dinámica (DP):** Especialmente eficientes para variantes del problema con capacidad moderada. Su complejidad es pseudo-polinomial ($\mathcal{O}(nW)$), donde W es la capacidad total, pero resulta inviable cuando W es grande.
- **Branch and Bound (B&B):** Este método consiste en explorar el espacio de soluciones a través de un árbol de decisión, donde en cada nodo se calcula un límite superior del valor que podría alcanzarse al seguir por esa rama. Si el límite no supera el mejor valor encontrado hasta el momento, la rama se poda. Aunque reduce el número de soluciones exploradas, su rendimiento se degrada con el crecimiento de n .

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

- **Programación entera lineal (ILP):** Es un método de optimización que aplica restricciones lineales pero requiere que algunas o todas las variables de decisión sean valores enteros. Utiliza formulaciones matemáticas y *solvers* como CPLEX [47] o Gurobi [39] para encontrar soluciones exactas. Sin embargo, su tiempo de resolución crece exponencialmente en instancias grandes.

Por otro lado, actualmente existen numerosas bibliotecas y plataformas que permiten resolver KP01 de forma automatizada, entre ellas:

- **Google OR-Tools:** Una potente suite de optimización que incluye solvers para programación entera, programación dinámica y algoritmos voraces.
- **Python libraries:** Bibliotecas como PuLP, Pyomo, SciPy.optimize y DEAP permiten implementar desde soluciones exactas hasta metaheurísticas evolutivas.
- **Matlab:** Proporciona herramientas como `intlinprog` y funciones personalizadas para algoritmos genéticos, simulated annealing, etc.

Si bien estas herramientas ofrecen soluciones eficientes en muchos casos, presentan limitaciones al abordar instancias de gran escala, debido al crecimiento exponencial del espacio de búsqueda o al ajuste fino que requieren las metaheurísticas tradicionales.

5.4.3. Ventajas de MCSA como estrategia alternativa

El algoritmo MCSA representa una metodología alternativa que aborda el problema desde una perspectiva diferente, sin necesidad de codificar directamente las restricciones en el espacio de búsqueda, y utilizando un método iterativo basado en el heurístico MCSA.

La presente propuesta que considera una representación del espacio de búsqueda como un conjunto discretizado y estructurado de n dimensiones, en el que cada combinación binaria se puede representar dentro de una región multidimensional.

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

A medida que se incrementa el número de dimensiones, se reduce la cantidad de elementos a ubicar por dimensión, lo cual permite gestionar más eficientemente la búsqueda. Esta propiedad es especialmente ventajosa en problemas donde n es elevado, ya que evita el colapso de la búsqueda en regiones de baja densidad y mejora la capacidad de convergencia hacia soluciones de calidad.

Además, la capacidad del MCSA de operar tanto en espacios discretos como continuos lo convierte en un algoritmo versátil, adaptable a una amplia gama de problemas sin necesidad de rediseñar su arquitectura básica. La metodología, que combina la Exploración basada en Montecarlo y Explotación mediante clustering en dos pasos, permite una Exploración eficaz del espacio sin caer fácilmente en óptimos locales, lo cual es un riesgo habitual en algoritmos como, por ejemplo, los algoritmos voraces (*greedy*) o los métodos deterministas.

Respecto a las ventajas competitivas de la propuesta desarrollada en esta tesis:

- **Escalabilidad:** El algoritmo ha demostrado buenos resultados en instancias de hasta 2000 elementos, donde muchos métodos exactos se vuelven prohibitivamente lentos, y por tanto, ineficientes.
- **Estructura adaptable:** Su formulación en múltiples dimensiones favorece la segmentación natural del espacio de búsqueda.
- **Versatilidad:** Puede ser extendido a problemas continuos, multiobjetivo o con restricciones complejas, manteniendo la lógica base.
- **Simplicidad de implementación:** A pesar de su solidez teórica, nuestro algoritmo de búsqueda no requiere herramientas comerciales ni optimizadores sofisticados.

Este conjunto de atributos posiciona a MCSA como una herramienta complementaria y prometedora para abordar problemas de optimización combinatoria, especialmente en instancias en las cuales, los métodos clásicos o exhaustivos resultan ineficientes o demasiado costosos computacionalmente.

5.5. Implementación del algoritmo MCSA en el KP01

En esta sección, describimos la implementación paso a paso del algoritmo MCSA, diseñado para abordar problemas de optimización complejos en espacios discretos de alta dimensión. La implementación se organiza en dos fases operativas principales, Exploración y Explotación, que, mediante la parametrización y mecanismos de adaptación, permiten primero una búsqueda global efectiva y luego un refinamiento local en regiones prometedoras del espacio de búsqueda.

5.5.1. Antecedentes

Una de las aplicaciones más importantes de la aplicación de métodos heurísticos para la resolución de problemas complejos está en [32]. El MCSA, por su parte, ha demostrado su eficacia en la resolución de problemas de optimización multidimensional mediante pruebas exitosas con funciones benchmark. Esta experiencia respalda la solidez de su estructura general y de sus mecanismos de Exploración y Explotación. No obstante, su aplicación al KP01 requiere un proceso de adaptación específico, ya que las características del espacio de búsqueda discreto y combinatorio difieren de los entornos previamente evaluados. En este trabajo se plantea dicha adaptación, aprovechando las bases consolidadas del algoritmo muestreo Montecarlo, clustering en dos pasos y evaluación iterativa de soluciones como punto de partida para desarrollar la metodología propuesta y ajustarse a las particularidades del nuevo problema.

De este modo, el MCSA que se presenta en esta tesis constituye la consolidación de un proceso de desarrollo propio que se adapta y amplía para su aplicación a problemas combinatorios de alta dimensión, como el KP01, donde la enumeración exhaustiva es inviable y la Exploración inteligente del espacio es necesaria.

Representación del espacio y discretización

En lugar de discretizar un espacio continuo, el problema de la mochila ya posee un espacio de búsqueda naturalmente discreto y finito, por lo que la representación se basa en la enumeración implícita de estas combinaciones binarias. Sobre este espacio se aplican las técnicas de muestreo de Montecarlo para explorar y evaluar soluciones, seleccionando combinaciones de elementos siguiendo un muestreo probabilístico. Por tanto, el espacio de búsqueda considerado es un espacio discretizado y binario de dimensión n , donde cada dimensión corresponde a un elemento disponible para incluir en la mochila. Este conjunto representa un hipercubo binario donde cada vértice corresponde a una solución factible (o no) del problema. Cada variable de decisión puede tomar únicamente dos valores posibles: 0 si el elemento no se incluye, o 1 si el elemento sí se incluye. Por lo tanto, el espacio de búsqueda está conformado por todas las posibles combinaciones binarias de los n elementos, formando un espacio de 2^n soluciones potenciales.

Fase de Exploración: principios y propósito

La Fase de Exploración tiene como objetivo generar un conjunto inicial de soluciones representativas que cubran de manera uniforme el espacio de búsqueda, respetando las restricciones del KP01. Para facilitar un análisis coherente, los elementos y sus combinaciones se ordenan previamente según un orden ascendente de peso o valor, lo que asegura continuidad en los ejes del espacio y permite identificar con claridad las regiones relevantes desde el inicio.

Durante cada iteración de Montecarlo se generan 100 muestras, evaluando en cada caso la validez de la combinación propuesta mediante la suma de los pesos de los elementos seleccionados. Solo aquellas combinaciones que cumplen con la restricción de capacidad se consideran válidas y se incorporan al conjunto de soluciones. Una vez obtenidas las muestras válidas, se aplica la función de fitness, que sirve para monitorizar la convergencia y determinar el momento adecuado para detener la Exploración. Este mecanismo garantiza que se genere un muestreo amplio y representativo, proporcionando una base sólida para las etapas posteriores

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

del análisis.

Tras completar la Exploración inicial, se procede a reducir el mapa Montecarlo con el objetivo de enfocar los recursos computacionales en las regiones más prometedoras. Aunque todas las soluciones retenidas son viables, se seleccionan únicamente aquellas que presentan los mejores valores de desempeño, eliminando combinaciones de menor interés. Este proceso de corte no compromete la viabilidad de las soluciones, sino que permite concentrar el análisis en subespacios con alta probabilidad de contener soluciones óptimas o cuasi-óptimas. La reducción del mapa constituye, por tanto, un paso crucial para optimizar la eficiencia de la Fase de Explotación, evitando el procesamiento innecesario de combinaciones poco relevantes.

Fase de Explotación

Tras la reducción del espacio, se activa la Fase de Explotación, cuyo objetivo principal es intensificar la búsqueda en torno a las regiones prometedoras. En esta etapa, la metodología se centra en refinar las combinaciones de elementos que sean válidas, explorando su vecindad para acercarse a óptimos locales o globales.

El proceso de Explotación combina clustering en dos pasos de la siguiente forma:

1. **DBSCAN:** identifica grupos de soluciones válidas dentro del espacio reducido. En el contexto del KP01, estos clusters representan subconjuntos de ítems que cumplen las restricciones y la función objetivo.
2. **K-means:** se aplica sobre cada región detectada por DBSCAN para localizar centroides representativos, que funcionan como puntos centrales alrededor de los cuales se generan nuevas soluciones mediante las iteraciones del heurístico MCSA.

A diferencia de la Fase de Exploración, donde se cubre el espacio de búsqueda de manera amplia, la Fase de Explotación limita la generación de nuevas soluciones a las vecindades de los centroides. Esto permite intensificar la búsqueda, equilibrando diversificación y Explotación local, y asegurando que los recursos computacionales

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

se concentren en las configuraciones de mayor potencial para el KP01. Así, la combinación de DBSCAN y K-means se adapta al carácter discreto y combinatorio del problema de la mochila, proporcionando un mecanismo robusto para refinar soluciones y mejorar progresivamente los resultados obtenidos en etapas anteriores.

Finalmente, el criterio de parada del algoritmo se establece en función de la variación promedio de la función de aptitud entre iteraciones: si dicha variación es mínima o se mantiene estable durante varias iteraciones consecutivas, se considera que el algoritmo ha convergido hacia una solución suficientemente buena. Esta estrategia asegura que la búsqueda se detenga de manera eficiente, evitando muestreo innecesario y concentrando los recursos computacionales en regiones de mayor potencial.

5.6. Aplicación del MCSA al KP01

La aplicación del MCSA al KP01 comienza con la etapa de *preparación inicial del problema*, que consiste en representar estructuradamente el espacio de búsqueda y codificar los parámetros relevantes (peso, valor y capacidad de la mochila). Esta Fase define cómo se particiona el espacio, valida combinaciones factibles y establece la base para la Fase de Exploración posterior, donde se identifican regiones prometedoras mediante muestreo.

Su formulación consiste en seleccionar un subconjunto de elementos de manera que se maximice su valor total sin exceder una capacidad máxima, siendo cada elemento i caracterizado por un peso w_i y un valor p_i . Cada combinación se representa mediante un código binario, asegurando que todas las configuraciones factibles se puedan identificar y ordenar según la función objetivo $Z(x)$, lo que mantiene continuidad y coherencia en el mapa Montecarlo y facilita la aplicación posterior de clustering. Este procedimiento asegura que muestras cercanas en el espacio compartan características similares, lo cual facilita la aplicación de técnicas de clustering en etapas posteriores.

La Figura 5.1 ilustra este proceso: primero, con Montecarlo se generan y calculan

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

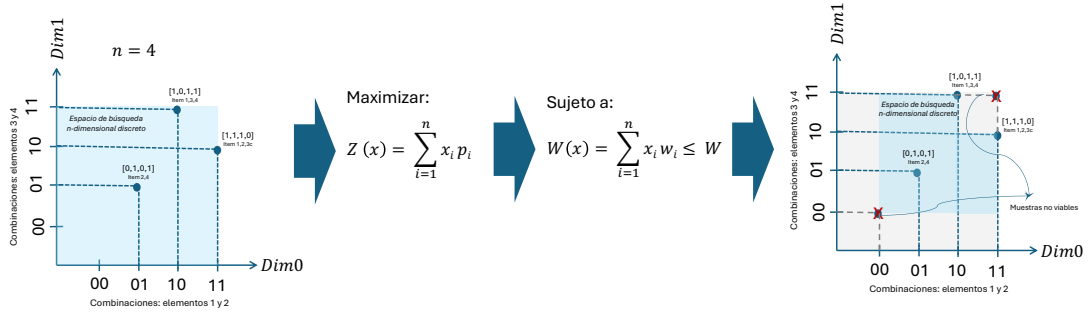


Figura 5.1: Flujo desde el proceso de muestreo hasta la evaluación de cada muestra que promueve la creación de un mapa de Montecarlo

múltiples combinaciones de elementos, asegurando que se cumpla la restricción impuesta, construyéndose así un mapa de Montecarlo multidimensional $\mathcal{S} \subseteq \Omega$ que sirve de base para la posterior identificación de regiones densas y prometedoras. Esta adaptación no solo conserva el núcleo del MCSA, sino que introduce mecanismos de verificación (como las restricciones y la función de aptitud) que permiten evaluar la calidad de las soluciones, asegurando la convergencia hacia configuraciones viables sin depender de exploraciones exhaustivas del espacio total.

Los límites del espacio de búsqueda están determinados por el cálculo del conjunto potencia. Es decir, 2^n . En la Figura 5.2, se muestra un ejemplo con la distribución de 6 elementos en un espacio bidimensional (Ejes X, Y), en el que se colocan 3 elementos por dimensión, de manera que cada dimensión genera $2^3 = 8$ combinaciones posibles. La combinación de ambas dimensiones produce un total de $8 \times 8 = 64$ posibles configuraciones. Este conocimiento hace que se vuelva necesario asegurar que cada combinación factible se pueda identificar. Una de las formas más sencillas de hacerlo es mediante un código binario, lo que facilita la interpretación y permite ordenar los elementos según la función objetivo $Z(x)$ (o el objetivo que se quiera cumplir), garantizando continuidad y coherencia en el mapa de Montecarlo.

Una vez asignados todos los identificadores, y las combinaciones han sido ordenadas según la función objetivo $Z(x)$, se emplea el **Índice Montecarlo**, que es un identificador que utilizaremos para ubicarlas en los ejes del espacio de búsqueda. Este procedimiento se realiza para asegurar que cada muestra quede situada junto

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

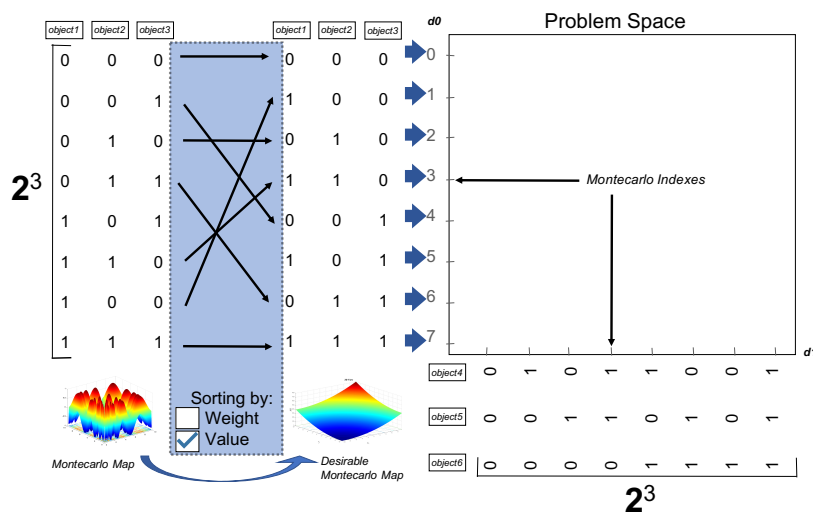


Figura 5.2: Representación del proceso de distribución y ordenación de 6 elementos en dos dimensiones respetando el tamaño del problema.

a otras de valores similares, evitando no solo la Exploración exhaustiva de todo el mapa, sino también una Exploración dispersa o desordenada por regiones inconexas del espacio.

La aplicación del MCSA al KP01 constituye un ejemplo de cómo un algoritmo previamente validado en entornos multiobjetivo también puede adaptarse a problemas discretizados. Se conservan la robustez y la capacidad de generalización de la metodología, al tiempo que se incorporan ajustes específicos que responden a las particularidades del problema, como por ejemplo, la representación de las soluciones y el filtrado inicial de combinaciones no factibles.

Cabe destacar que aplicar el MCSA también permite explorar aproximaciones a soluciones de tipo Pareto en contextos multiobjetivo. En el caso del KP01, esto se realiza mediante un proceso de escalarización, en el cual se asignan ponderaciones a los distintos objetivos, por ejemplo, peso y valor o beneficio, para combinar dichos objetivos en una única función agregada. Esta estrategia, aunque no constituye un enfoque estrictamente multiobjetivo en el sentido clásico, sí permite obtener soluciones que representan compromisos entre los objetivos en conflicto. Al evaluar las combinaciones de elementos bajo diferentes esquemas de

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

ponderación, es posible identificar subconjuntos de soluciones que aproximan un frente de Pareto y que cumplen con las restricciones impuestas. Esto anticipa la extensión natural del algoritmo hacia problemas multiobjetivo más completos y demuestra su versatilidad para abordar distintas formulaciones del problema de la mochila.

5.6.1. Evaluación del MCSA en el KP01

Una vez descrito el procedimiento de ordenación y representación multidimensional mediante ejemplos ilustrativos de pequeña escala, se procedió a validar la metodología sobre instancias de mayor complejidad del problema de la mochila. En particular, se resolvió la instancia EX01, compuesta por 24 elementos con pesos y beneficios heterogéneos. En lugar de presentar la lista completa de elementos, la Tabla 5.1 resume su distribución multidimensional dentro del espacio de búsqueda, mostrando cómo cada ítem queda asignado a las distintas particiones generadas por la estructura del MCSA. Esta representación ofrece una visión condensada pero suficiente para comprender la organización del problema sin necesidad de detallar individualmente los 24 registros.

En otro ejemplo, se podrían distribuir 12 elementos por dimensión, lo cual, daría lugar a 4096 combinaciones posibles por eje. En contraste, al usar una distribución de 4 dimensiones con 6 elementos por cada una, el número de combinaciones se reduciría a solo 64 por dimensión. Esta ordenación disminuiría significativamente la cantidad de elementos que deben colocarse en cada eje, lo que podría mejorar el rendimiento. En el extremo opuesto, una distribución de 24 dimensiones con un solo elemento por dimensión minimiza aún más el esfuerzo de ordenamiento, lo cual podría ser ventajoso cuando se manejan grandes volúmenes de muestras.

La instancia EX01 permitió comprobar la escalabilidad del método, ya que incrementa notablemente la dimensionalidad del espacio de búsqueda respecto a los ejemplos introductorios. A partir de esta definición se aplicó el proceso de muestreo, reducción y Explotación previamente descrito, verificándose que el MCSA mantiene un comportamiento preciso y coherente incluso en problemas de

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

Tabla 5.1: Distribución multidimensional de elementos en el espacio de búsqueda para la instancia *EX01* del problema de la mochila, con 24 elementos.

Arreglo dimensional para la instancia EX01/KP01												
ID element (n)	1	2	3	4	5	6	7	8	9	10	11	12
weight (i)	382745	799601	909247	729069	467902	44328	34610	698150	823460	903959	853665	551830
profit (i)	825594	1677009	1676628	1523970	943972	97426	69666	1296457	1679693	1902996	1844992	1049289
2-Dimensional	1st dim											
3-Dimensional	1st dim								2nd dim			
4-Dimensional	1st dim						2nd dim					
6-Dimensional	1st dim				2nd dim				3rd dim			
8-Dimensional	1st dim			2nd dim			3rd dim			4th dim		
12-Dimensional	1st dim		2nd dim		3rd dim		4th dim		5th dim		6th dim	
24-Dimensional	1st dim	2nd dim	3rd dim	4th dim	5th dim	6th dim	7th dim	8th dim	9th dim	10th dim	11th dim	12th dim
ID element (n)	13	14	15	16	17	18	19	20	21	22	23	24
weight (i)	610856	670702	488960	951111	323046	446298	931161	31385	496951	264724	224916	169684
profit (i)	1252836	1319836	953277	2067538	675367	853655	1826027	65731	901489	577243	466257	369261
2-Dimensional	2nd dim											
3-Dimensional	2nd dim				3rd dim							
4-Dimensional	3rd dim						4th dim					
6-Dimensional	4th dim				5th dim				6th dim			
8-Dimensional	5th dim			6th dim			7th dim			8th dim		
12-Dimensional	7th dim		8th dim		9th dim		10th dim		11th dim		12th dim	
24-Dimensional	13th dim	14th dim	15th dim	16th dim	17th dim	18th dim	19th dim	20th dim	21th dim	22th dim	23th dim	24th dim

mayor tamaño. Esta validación sirvió además como punto de partida para analizar el rendimiento del método tanto en términos de eficiencia computacional como de calidad de las configuraciones de solución, cuyos resultados se presentan en la siguiente sección.

Para instancias de mayor tamaño, o cuando el número de elementos aumenta y debemos trabajar con una entrada extensa que involucra muchos elementos y variables, se recurre a una técnica de distribución en n -dimensiones para resolver el problema. Tal como se muestra en la misma Tabla 5.1, los datos de entrada pueden distribuirse a lo largo de las n dimensiones, donde podemos considerar el uso de arreglos n -dimensionales que contengan subconjuntos de elementos, manteniéndose constante el tamaño del problema.

Una vez finalizado el proceso de ordenamiento y distribución de los elementos a lo largo de los ejes, se da inicio a la Fase de Exploración mediante la generación de muestras aleatorias, con el objetivo de lograr una distribución uniforme de combinaciones válidas sobre el espacio de búsqueda. En resumen, cada combinación

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

específica de elementos (o subconjunto) está representada en una muestra, y su valor se determina a partir de la suma de los pesos de los elementos seleccionados de los subconjuntos correspondientes. Por tanto, para que una muestra sea considerada válida, la suma total de los pesos debe respetar la restricción del problema (es decir, no debe superar la capacidad de la mochila). Cualquier muestra que no cumpla esta condición será descartada. Posteriormente, en la Fase de Explotación, mediante la estrategia de corte por $f(x)$ se eliminan muestras redundantes y permite finalmente la identificación de las regiones factibles.

A continuación, se describe la metodología, aplicada a la instancia *EX01* propuesta por Kreher y Stinson [56] (disponible en línea).

5.6.2. Resultados Experimentales

En el caso de la instancia *EX01*, que representa un problema de la Mochila en 8 dimensiones con 3 elementos por dimensión, se generan 8 combinaciones por cada eje, manteniéndose un tamaño total de $1,676E+07$ combinaciones. Antes de aplicar la Fase de Exploración, se organizan los elementos y sus combinaciones a lo largo de los ejes, permitiendo que el algoritmo MCSA genere muestras aleatorias y construya el mapa de Montecarlo. Solo se conservan aquellas muestras que cumplen con la restricción de peso W , descartando el resto.

La Tabla 5.2 muestra cómo MCSA identifica y calcula cada combinación. El identificador binario permite calcular el peso y beneficio total de cada muestra. En ella se representan todas las combinaciones posibles de la Dimensión uno de la instancia *EX01*, mientras que las otras siete dimensiones se someten al mismo proceso. Finalmente, se selecciona la combinación que maximiza la función $Z(x)$, cumpliendo con la restricción de peso W .

A continuación, en la Figura 5.3 se ilustra una representación visual del proceso de preparación del espacio de búsqueda. En primer lugar, se calculan los valores de las muestras a partir de todas las combinaciones posibles. Luego, estas combinaciones se ordenan en forma ascendente según el valor de la función objetivo, lo que genera un índice de Montecarlo para cada una. Este reordenamiento constituye un paso

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

Tabla 5.2: Solución para la instancia *EX01*. Se representan las posibles combinaciones de los elementos de la dimensión 1 (de un total de 8).

Combinaciones (Cn)		Peso de los Elementos (w_i)					
Id		no. Elem.	1	2	3	$\sum w_i$	$\sum$ Beneficio
C0	000	-	-	-	-	-	0
C1	001	1	382745	-	-	382745	825594
C2	010	2	-	799601	-	799601	1677009
C3	011	1,2	382745	799601	-	1182346	2502603
C4	100	3	-	-	909247	909247	1676628
C5	101	1,3	382745	-	909247	1291992	2502222
C6	110	2,3	-	799601	909247	1708848	3353637
C7	111	1,2,3	382745	799601	909247	2091593	4179231

previo esencial antes de ubicar las muestras a lo largo de los ejes del espacio multidimensional. La figura ilustra conceptualmente este proceso sobre la instancia introductoria mencionada del KP01 y $n = 6$.

En este paso, desde la tabla inicial con las combinaciones sin ordenar, pasando por la versión ordenada según el índice de Montecarlo, hasta la representación espacial en un cubo conceptual, las muestras se distribuyen conforme a dicho orden.

El proceso de muestreo, con que da inicio la Fase de Exploración, permite construir un mapa de soluciones candidatas a partir de la distribución aleatoria de combinaciones válidas y que cumplen la restricción impuesta. A continuación, dicho mapa es depurado mediante un procedimiento de corte sobre $f(x)$, con el objetivo de conservar únicamente las muestras más prometedoras en términos del valor de la función objetivo. En las observaciones, la formulación multidimensional del Problema de la Mochila exhibe un comportamiento análogo al de funciones continuas, lo que facilita la aplicación de la técnica exploratoria del MCSA sobre el espacio de soluciones. A medida que avanza la búsqueda, el algoritmo transita progresivamente hacia la Fase de Explotación, en la cual se emplean mecanismos de agrupamiento para detectar regiones factibles con mayor concentración de soluciones. La Figura 5.4 ilustra esta transición, destacando la reorganización estructural del espacio de búsqueda a partir de los clusters identificados.

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

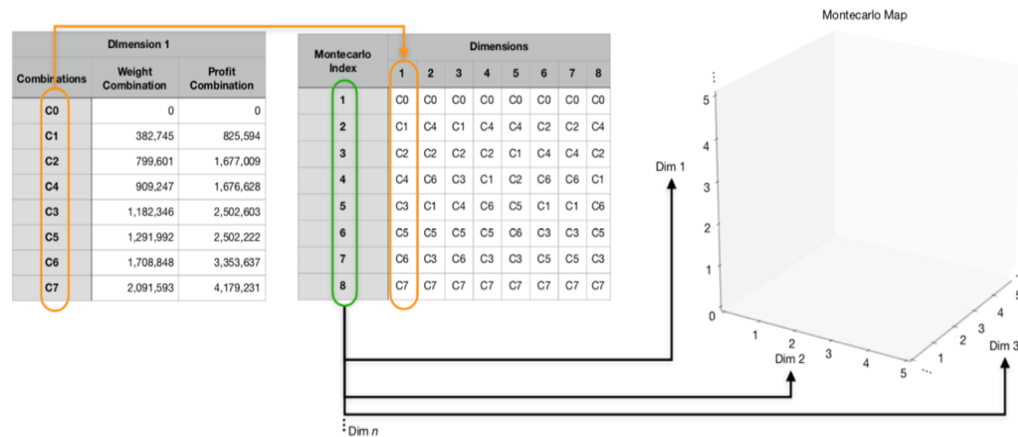


Figura 5.3: Representación del proceso de organización del espacio de búsqueda antes de la Fase de Exploración. Se propone un reordenamiento ascendente según el valor de la función objetivo (índice de Montecarlo). Finalmente, el cubo conceptual ilustra cómo se asignan estas combinaciones a lo largo de los ejes del espacio multidimensional, en preparación para la búsqueda mediante MCSA.

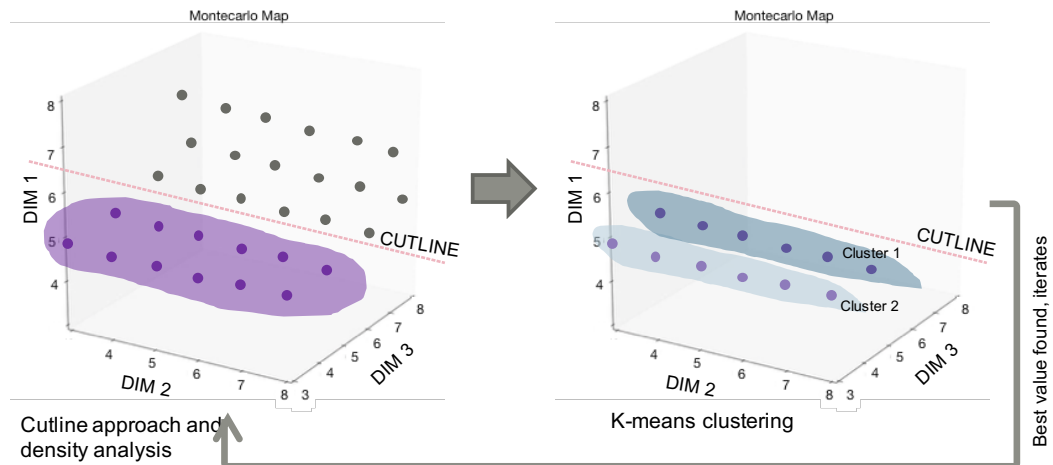


Figura 5.4: Transición a Fase de Explotación. a)Proceso de corte para reducir el espacio de búsqueda y b) Aplicación de DBSCAN para reducir el espacio, seleccionar las muestras más viables y posteriormente aplicar K-means.

Las iteraciones sucesivas contribuyen a refinar progresivamente la calidad de la solución mediante la re-Exploración focalizada en las regiones factibles. La Figura 5.5 ilustra el resultado final obtenido por el algoritmo MCSA, para la instancia

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

del KP01 con $n = 6$, mostrando la configuración óptima identificada. Cada muestra corresponde a una combinación específica seleccionada en cada una de las dimensiones del problema. En la tabla representada, los recuadros destacados señalan las combinaciones elegidas: $C3$ en la Dimensión 1, $C5$ en la Dimensión 2, $C4$ en la Dimensión 3, y así sucesivamente hasta $C2$ en la Dimensión 8. Cabe destacar que en la Dimensión 7 se seleccionó la combinación $C0$, correspondiente al conjunto vacío, lo cual indica que de esa dimensión no se incorporó ningún elemento a la solución final.

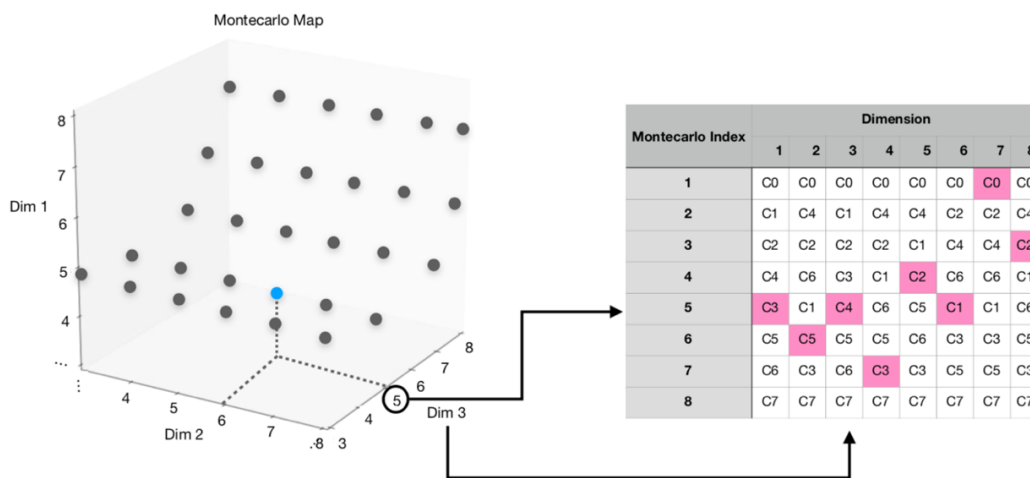


Figura 5.5: Representación de una muestra final generada por MCSA. Cada celda destacada indica la combinación de elementos seleccionada en una dimensión específica. La muestra completa está compuesta por la integración de estas combinaciones, una por dimensión, lo que define una solución candidata dentro del espacio de búsqueda.

Los resultados numéricos se presentan en la Tabla 5.3. En términos generales, la solución obtenida demuestra una alta calidad desde el punto de vista computacional. La proporción entre el número de muestras requeridas para alcanzar una solución factible y el tamaño total, α , del espacio de búsqueda es cercana al 1 %, lo que evidencia una Exploración altamente dirigida. Al comparar estos resultados con la solución previamente calculada, obtenida mediante un algoritmo exacto de ramificación y poda (*branch and bound*), la propuesta aquí desarrollada muestra un desempeño competitivo en calidad de solución obtenida.

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

Además, dado que el algoritmo reduce explícitamente el número de muestras generadas durante el proceso exploratorio, el método implica una disminución del esfuerzo computacional respecto a aproximaciones que requieren un muestreo más intensivo. Este aspecto se analiza conceptualmente en la Sección de Discusión de los Resultados 5.7, aunque no se presentan medidas directas de tiempo de ejecución.

Tabla 5.3: Resultados de MCSA de la instancia *EX01* resuelto en 8 dimensiones. Se muestra la mejor solución obtenida, destacando el peso total asociado a ella que cumple con la restricción del problema.

Dimensión	Combinación	Representación Binaria	Elementos	Peso 1	Peso 2	Peso 3	Peso Total	Ganancia Total
1	C3	011	1,2	382,745	799,601	–	1,182,346	2,502,603
2	C5	101	4,6	729,069	–	44,328	773,397	1,621,396
3	C4	001	9	823,460	–	–	823,460	1,679,693
4	C3	011	10,11	903,959	853,665	–	1,757,624	3,747,988
5	C2	010	14	–	670,702	–	670,702	1,319,836
6	C1	001	16	951,111	–	–	951,111	2,067,538
7	C0	000	–	–	–	–	0	0
8	C2	100	23	–	–	224,916	224,916	466,257
Peso Total							6,383,556	
Ganancia Total								13,405,311

La evaluación del heurístico MCSA mediante funciones de referencia proporcionó un marco preliminar para comprender las características del espacio de búsqueda y los patrones de las regiones factibles. Esta fase exploratoria fue esencial para establecer las bases metodológicas, ya que permitió identificar la similitud estructural entre superficies de referencia y el KP01, lo cual confirmó la idoneidad del MCSA para su aplicación en este tipo de problemas.

En el caso del KP01, la adaptación del modelo clásico a los parámetros del MCSA se llevó a cabo de manera íntegra: los elementos se distribuyeron a lo largo de las dimensiones, se calculó el conjunto potencia para conocer el número de combinaciones posibles y se aplicó un proceso de ordenación que garantiza continuidad en el espacio de búsqueda n -dimensional. Este aspecto resulta ser imprescindible, ya que asegura que el muestreo conserve información estructural sobre la vecindad de soluciones, reduciendo el coste de Exploración global.

La validación experimental comenzó con la resolución de la instancia *EX01*, que con $n = 24$ elementos, proviene de un dataset que se encuentra disponible en

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

repositorios de acceso abierto. Este caso fue tratado con detalle, analizando la correspondencia entre el número de dimensiones, el tamaño del problema y la calidad de las soluciones obtenidas. Posteriormente, se abordaron instancias de mayor complejidad, incluyendo *EX02* ($n = 50$), y una serie de instancias sintéticas diseñadas para incrementar progresivamente el número de elementos hasta alcanzar $n = 2000$. Esta progresión permitió examinar la escalabilidad de la metodología y contrastar la eficiencia de MCSA en escenarios crecientemente demandantes desde el punto de vista computacional.

A continuación se presentan los resultados obtenidos en todas las instancias evaluadas. En ella se detallan, para cada configuración de dimensiones, el tamaño del problema, el número de muestras procesadas, el tiempo requerido para la ordenación de los ejes y la calidad de la solución alcanzada en términos de proximidad al beneficio óptimo conocido. Como se observa en la Tabla 5.4, el método logra en todos los casos una solución cercana al óptimo, incluso cuando el número de elementos crece de forma exponencial y el espacio de búsqueda alcanza magnitudes inabordables por métodos exactos. En particular, se destaca que la calidad se mantiene por encima del 92 % en instancias de hasta $n = 2000$, lo que confirma la robustez del MCSA frente al crecimiento del problema. Asimismo, el bajo tiempo de ordenación de ejes pone de relieve la eficiencia computacional del MCSA, evidenciando que el esfuerzo computacional no escala de forma proporcional al tamaño del espacio de búsqueda.

Tabla 5.4: Detalle de las instancias a resolver: de *EX01* a *EX06*.

Instancia	# Elem.	Tamaño del Problema (Psize)	Capacidad (W)	Beneficio Óptimo (P)
<i>EX01</i>	24	$1,676 \times 10^7$	6,404,163	13,549,094
<i>EX02</i>	50	$1,125 \times 10^{15}$	850	7,534
<i>EX03 (sintético)</i>	240	$1,766 \times 10^{72}$	64,041,630	135,789,553
<i>EX04 (sintético)</i>	500	$3,272 \times 10^{150}$	2,394,376	2,477,959
<i>EX05 (sintético)</i>	1000	2^{1000}	4,864,967	5,052,984
<i>EX06 (sintético)</i>	2000	2^{2000}	9,860,929	10,173,777

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

Tabla 5.5: Resultados para diferentes instancias del KP01: de *EX01* a *EX06*, resueltas por medio del MCSA

EX01 - Capacidad (W) = 6,404,163, n = 24, Beneficio Óptimo (P) = 13,549,094							
# Dimension	Tamaño (Psize)	# de Muestras	# de Muestras vs Psize (%)	Ordenación Tiempo (seg.)	Beneficio Máx (P)	Peso Asociado	Calidad (%)
2	1.68E+07	1,727	0.0102	0.0992	13,463,657	6,403,149	99.36
4	1.68E+07	1,644	0.0097	0.0030	13,461,329	6,402,653	99.35
6	1.68E+07	3,147	0.0187	0.0011	13,437,475	6,375,863	99.17
8	1.68E+07	7,004	0.0417	0.0009	13,436,707	6,399,256	99.17
12	1.68E+07	4,429	0.0263	0.0006	13,385,615	6,389,719	98.79
24	1.68E+07	146	0.0008	0.0000	13,300,418	6,392,566	98.16
EX02 - Capacidad (W) = 850, n = 50, Beneficio Óptimo (P) = 7,534							
# Dimension	Tamaño (Psize)	# de Muestras	# de Muestras vs Psize (%)	Ordenación Tiempo (seg.)	Beneficio Máx (P)	Peso Asociado	Calidad (%)
2	1.13E+15	2,084	0.0000	1276.6851	6,242	805	82.85
5	1.13E+15	1,417	0.0000	0.0489	6,643	818	88.17
10	1.13E+15	2,856	0.0000	0.0023	6,962	810	92.40
25	1.13E+15	5,044	0.0000	0.0016	6,233	849	82.73
50	1.13E+15	524	0.0000	0.0011	6,445	772	85.54
EX03 Sintético - Capacidad (W) = 64,041,630, n = 240, Beneficio Óptimo (P) = 135,789,553							
# Dimension	Tamaño (Psize)	# de Muestras	# de Muestras vs Psize (%)	Ordenación Tiempo (seg.)	Beneficio Máx (P)	Peso Asociado	Calidad (%)
20	1.77E+72	557	0.0000	1.1231	130,279,429	63,791,741	95.94
30	1.77E+72	1,622	0.0000	0.0868	130,589,231	63,981,629	96.17
40	1.77E+72	899	0.0000	0.0226	130,448,449	63,941,709	96.06
48	1.77E+72	1,148	0.0000	0.0162	130,824,332	63,923,428	96.34
60	1.77E+72	3,170	0.0000	0.0089	130,805,991	63,987,106	96.32
80	1.77E+72	172	0.0000	0.0081	130,011,304	63,741,250	95.74
EX04 Sintético - Capacidad (W) = 2,394,376, n = 500, Beneficio Óptimo (P) = 2,477,959							
# Dimension	Tamaño (Psize)	# de Muestras	# de Muestras vs Psize (%)	Ordenación Tiempo (seg.)	Beneficio Máx (P)	Peso Asociado	Calidad (%)
50	3.272E+150	4,000	0.0000	1.7433	2,298,986	1,962,308	92.77
EX05 Sintético - Capacidad (W) = 4,864,967, n = 1000, Beneficio Óptimo (P) = 5,052,984							
# Dimension	Tamaño (Psize)	# de Muestras	# de Muestras vs Psize (%)	Ordenación Tiempo (seg.)	Beneficio Máx (P)	Peso Asociado	Calidad (%)
100	2^1000	7,000	0.0000	3.5074	4,660,746	3,947,911	92.23
EX06 Sintético - Capacidad (W) = 9,860,929, n = 2000, Beneficio Óptimo (P) = 10,173,777							
# Dimension	Tamaño (Psize)	# de Muestras	# de Muestras vs Psize (%)	Ordenación Tiempo (seg.)	Beneficio Máx (P)	Peso Asociado	Calidad (%)
200	2^2000	4,000	0.0000	6.9587	9,581,024	8,455,373	94.17

5.7. Discusión de los Resultados

En términos generales, los resultados muestran que el MCSA alcanza soluciones de alta calidad en todas las instancias analizadas. La Tabla 5.5 presenta los valores óptimos de referencia junto con los resultados alcanzados, lo que permite una comparación directa entre la calidad de las soluciones y los recursos computacionales empleados. Una primera observación relevante es la baja relación entre el número de muestras requeridas y el tamaño del problema, lo cual refleja que el MCSA no precisa un conocimiento proporcional del muestreo para mantener su desempeño. Asimismo, el análisis del tiempo de ordenación por eje (*Ordenación*) confirma que un menor número de elementos por dimensión se traduce en reducciones significativas en el tiempo de ordenación, por lo que la carga asociada a este paso también se reduce. Esto impacta directamente en el tiempo total de procesamiento y, por ende, en la gestión de recursos computacionales.

En cuanto a la calidad de las soluciones obtenidas en 30 ejecuciones para cada configuración y por número de dimensiones, se alcanzaron valores superiores al 98 % para *EX01*, un promedio del 86 % en *EX02* y superiores al 96 % en *EX03*. En el caso de las instancias *EX04*, *EX05* y *EX06*, los valores obtenidos fueron del 92.77 %, 92.23 % y 94.17 % respectivamente.

Como se puede ver, estos resultados son especialmente notables en escenarios de alta dimensionalidad, donde la heurística mantiene niveles de calidad competitivos a pesar de la complejidad inherente al problema. Un caso particular de éxito se observa en la instancia *EX01* con 8 dimensiones, donde se alcanzó un beneficio máximo de **13.436.707**, equivalente al 99.17 % del valor óptimo, respetando además la restricción de peso W y con una relación de muestras menor al 1 % respecto al tamaño del espacio de búsqueda, por lo que, desde una perspectiva computacional, esta instancia mostró una alta eficiencia (relación entre el número de muestras necesarias y tamaño del problema).

La formulación de instancias sintéticas respondió a dos objetivos principales: (i) aumentar progresivamente la complejidad del problema mediante el incremento del número de elementos, lo que conlleva un crecimiento exponencial del espacio

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

de soluciones, y (ii) definir soluciones óptimas de referencia que permitieran comparaciones significativas y robustas. Los resultados obtenidos confirman que el MCSA es capaz de mantener una elevada calidad en entornos de creciente dimensionalidad, otorgando una ventaja clara frente a algoritmos de fuerza bruta que, aunque logran la solución óptima, incurren en un coste computacional prohibitivo.

No obstante, cabe reconocer que la metodología enfrenta las limitaciones propias de los problemas n -dimensionales. A medida que aumenta el número de dimensiones, se intensifica la conocida maldición de la dimensionalidad, fenómeno que implica un crecimiento exponencial del espacio de búsqueda y una disminución drástica de la densidad relativa de las muestras. En términos prácticos, esto se traduce en que, aunque el número absoluto de muestras se mantenga elevado, su distribución resulta cada vez más dispersa en el espacio, dificultando la identificación de regiones de interés y reduciendo la eficiencia de la Exploración.

En el caso particular del MCSA, esta dispersión introduce varios efectos: (i) la convergencia hacia soluciones de alta calidad puede volverse más lenta, ya que es necesario incrementar el número de muestras para mantener niveles de precisión comparables, (ii) se incrementa el riesgo de estancamiento en regiones subóptimas debido a la menor densidad de vecinos con valores significativos, y (iii) se reduce la capacidad de aproximar de manera precisa las fronteras de Pareto en contextos multiobjetivo, puesto que la escasez relativa de muestras afecta la cobertura uniforme de dichas fronteras.

Con el fin de contrarrestar estas limitaciones, se exploraron estrategias complementarias dentro del marco del MCSA. Una de ellas fue la implementación de un esquema de *partición del espacio de búsqueda*, que consiste en dividir el espacio multidimensional en subregiones manejables y aplicar el heurístico de forma independiente en cada una de ellas. Esta aproximación, aunque utilizada de manera preliminar y no publicada en los resultados experimentales, demostró ser viable y representa una alternativa prometedora para mantener una cobertura más uniforme del espacio, facilitar la convergencia y reducir el impacto de la dispersión de muestras en altas dimensiones.

Estas dificultades, compartidas con otros algoritmos heurísticos, señalan la necesidad de profundizar en la línea de investigación mencionada. El perfeccionamiento de la estrategia de partición, junto con la integración de esquemas adaptativos de muestreo, permitiría extender la aplicabilidad del MCSA a problemas de mayor escala, donde las exigencias computacionales se vuelven críticas y las limitaciones aquí descritas se hacen más evidentes.

5.8. Adaptación de la metodología al problema de la mochila multiobjetivo (MOKP)

Para ampliar la aplicabilidad de nuestra metodología a otros problemas de optimización combinatoria de tipo NP, consideramos el Problema de la Mochila Multiobjetivo (MOKP), que constituye una extensión del KP01 original. En esta variante, nuestro objetivo es optimizar simultáneamente dos funciones objetivo distintas: por un lado, maximizar el peso total de los elementos seleccionados para incluir en la mochila y, por otro, maximizar el beneficio generado por estos elementos seleccionados, todo ello respetando la restricción de peso.

Aunque el enunciado fundamental del KP01 se mantiene sin cambios, nuestra propuesta introduce una diferencia notable en cuanto a la estrategia de resolución. En este contexto, tratamos la restricción de peso tanto como una limitación como una función objetivo, transformando así el problema en un Problema de la Mochila Biestructurado. Por supuesto, la incorporación de restricciones adicionales incrementa aún más la complejidad inherente del problema dentro de este marco.

El campo de la optimización multiobjetivo se ocupa de problemas de optimización que implican múltiples objetivos que, en general, son conflictivos entre sí, en el sentido de que no existe una solución factible que sea óptima para todos ellos a la vez. Por lo tanto, deben buscarse soluciones de compromiso. Una solución de compromiso es aquella que no puede mejorarse en uno de los objetivos sin empeorar al menos uno de los valores de las otras funciones objetivo.

Para abordar el Problema de la Mochila Multiobjetivo, es necesario

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

adaptar nuestra metodología a sus parámetros específicos. Como se demostró anteriormente, mediante el cálculo del conjunto potencia, al igual que en el caso binario de la mochila, se puede determinar el tamaño del problema y, posteriormente, definir los límites del espacio de búsqueda de n -dimensiones. La propuesta distintiva aquí consiste en disponer los elementos, cada uno con sus respectivos pesos, valores y otros atributos relevantes, dentro del espacio de búsqueda utilizando para ello una agregación escalar de dichos atributos. Esta agregación es un mecanismo que asigna coeficientes o factores de importancia a cada objetivo con el fin de generar un único valor representativo que facilite la ordenación y comparación de las combinaciones.

Este procedimiento permite asignar pesos y combinar múltiples objetivos en una única métrica operativa, manteniendo la flexibilidad necesaria para manejar coeficientes reales o enteros según la formulación del problema. Además, la técnica admite su aplicación en variantes del problema de la mochila donde los elementos pueden ser fraccionados, lo cual introduce una complejidad adicional. En este contexto, la agregación escalar actúa como un puente entre la representación multidimensional del problema y la necesidad de construir un espacio de búsqueda ordenado y coherente para el funcionamiento del MCSA.

5.8.1. Definición formal del Problema de la Mochila Multiobjetivo

El Problema de la Mochila Multiobjetivo (MOKP, por sus siglas en inglés) constituye una extensión natural de la versión clásica del problema de la mochila, en la que se consideran simultáneamente múltiples objetivos de optimización. Formalmente, se define de la siguiente manera:

$$\begin{aligned} \text{Maximizar: } & \mathbf{Z}(x) = (z_1(x), z_2(x), \dots, z_m(x)) \\ \text{sujeto a: } & \sum_{j=1}^n w_j x_j \leq W, \\ & x_j \in \{0, 1\}, \quad j = 1, \dots, n, \end{aligned} \tag{5.4}$$

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

donde cada función objetivo $z_i(x)$ se calcula como

$$z_i(x) = \sum_{j=1}^n c_j^i x_j, \quad i = 1, \dots, m. \quad (5.5)$$

En esta formulación, n representa el número total de elementos disponibles, m el número de objetivos, c_j^i el beneficio del elemento j según el objetivo i , w_j el peso o coste asociado a dicho elemento, y W la capacidad máxima de la mochila. La variable de decisión x_j toma el valor 1 si el elemento j es incluido en la solución, y 0 en caso contrario. Se asume que W , c_j^i y w_j son valores enteros positivos, que $w_j \leq W$ para todo j , y que $\sum_{j=1}^n w_j > W$, de manera que no sea posible incluir todos los elementos simultáneamente, evitando así formulaciones triviales.

A diferencia del problema monoobjetivo, en el contexto multiobjetivo no existe, en general, una única solución óptima. En su lugar, se trabaja con el concepto de dominancia de Pareto: una solución x^1 domina a otra x^2 si cumple que $z_i(x^1) \geq z_i(x^2)$ para todo $i = 1, \dots, m$, y al menos una desigualdad es estricta. El conjunto de soluciones no dominadas se denomina *frontera de Pareto*, y constituye el principal objeto de interés en este tipo de problemas, dado que refleja el compromiso entre los distintos objetivos.

En la práctica, este tipo de problemas se resuelve aplicando técnicas que permiten aproximar dicha frontera. Una técnica habitual consiste en la combinación lineal ponderada de los objetivos:

$$Z_\lambda(x) = \sum_{i=1}^m \lambda_i z_i(x), \quad \lambda_i \geq 0, \quad \sum_{i=1}^m \lambda_i = 1, \quad (5.6)$$

donde los parámetros λ_i permiten asignar diferentes niveles de importancia relativa a cada objetivo. Este mecanismo constituye una forma de explorar posibles soluciones no dominadas bajo determinados pesos, generando así distintos puntos de compromiso. Sin embargo, este enfoque presenta limitaciones, pues no garantiza la identificación de toda la frontera de Pareto cuando esta es no convexa. Para ampliar la Exploración pueden emplearse otros métodos conocidos,

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

como el método ε -constraint o algoritmos evolutivos multiobjetivo específicos (por ejemplo, NSGA-II o SPEA2), capaces de aproximar un conjunto más diverso y representativo de soluciones. En cualquier caso, el MOKP es reconocido como NP-difícil, lo que motiva la aplicación de heurísticas y metaheurísticas, como el MCSA, para obtener soluciones de calidad con un coste computacionales moderado y acorde a la complejidad del problema.

5.8.2. Resultados de la implementación

La Tabla 5.6 presenta los resultados obtenidos para la instancia *EX01*, previamente descrita, adaptada al contexto multiobjetivo a partir de su formulación clásica. En este caso, se consideraron distintos factores de importancia para los objetivos de peso y beneficio, con el objetivo de ilustrar la capacidad de la metodología propuesta para generar soluciones de compromiso bajo diferentes configuraciones de preferencias. La experimentación abarcó tanto distribuciones bidimensionales como de mayor dimensionalidad, lo que permite evaluar el desempeño de la metodología a medida que se incrementa la complejidad del espacio de búsqueda.

Tabla 5.6: Resultados obtenidos para la instancia *EX01* adaptada como problema multiobjetivo con distintos factores de importancia.

Instancia o Problema: <i>EX01</i> Capacidad de la mochila (W) = 6,404,180 (n) = 24 elementos						
Tamaño Problema (Psize)	# de Dimensiones	# de Muestras	# de Muestras vs Psize (%)	Mayor Beneficio	Peso Asociado	Factor de Importancia {0-1}
$1,677 \times 10^7$	2	12,902	0.08	13,524,340	6,397,001	Weight = 0, Profit = 1
$1,677 \times 10^7$	2	13,476	0.08	12,980,893	6,404,180	Weight = 1, Profit = 0
$1,677 \times 10^7$	12	12,174	0.07	13,518,963	6,398,128	Weight = 0, Profit = 1
$1,677 \times 10^7$	2	12,928	0.08	13,517,399	6,389,617	Weight = 0.5, Profit = 0.5
$1,677 \times 10^7$	12	11,994	0.07	13,509,694	6,390,248	Weight = 0.5, Profit = 0.5

Como puede observarse, la variación de los factores de importancia influye directamente en la selección de la solución, evidenciando que el método de la suma ponderada permite generar distintas alternativas de compromiso. Al priorizar exclusivamente el beneficio, se alcanzan valores de ganancia superiores, aunque a costa de un mayor uso de la capacidad disponible. Por el contrario, al dar

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

mayor relevancia al peso, las soluciones se ajustan estrictamente a la restricción de capacidad, sacrificando parte del beneficio potencial.

La Exploración de distribuciones multidimensionales (hasta 12 dimensiones) confirma que el método mantiene un desempeño consistente, aunque con un incremento del esfuerzo de búsqueda proporcional a la complejidad del espacio de soluciones. Esto demuestra que el MCSA es capaz de generar soluciones de alta calidad con un número significativamente reducido de muestras, destacando su eficiencia en la Exploración del espacio de búsqueda y la robustez frente a la expansión dimensional.

En términos cuantitativos, como se aprecia en la misma Tabla 5.6, incluso para dimensiones mayores, la proporción de muestras frente al tamaño total del problema se mantiene por debajo del 8 %, alcanzando beneficios cercanos al óptimo. Esta característica evidencia la capacidad del método para ofrecer resultados competitivos minimizando el uso de recursos computacionales.

En síntesis, esta validación experimental pone de relieve que el MCSA no solo es capaz de manejar instancias de tamaño medio como *EX01*, sino que también ofrece flexibilidad para representar distintas configuraciones de preferencias multiobjetivo. Estos resultados constituyen un punto de partida sólido para el análisis posterior y sirven como antesala a las conclusiones generales del capítulo.

5.9. Conclusión del capítulo

En este capítulo se presentó la heurística MCSA como una propuesta destinada a resolver problemas de optimización combinatoria, estructurada en sus dos fases: *Exploración* y *Explotación*. El proceso completo incorpora además una etapa previa de preparación y parametrización, necesaria para definir el espacio de búsqueda, las restricciones y los parámetros operativos que harán posible la fase exploratoria. En cuanto a la Fase de Exploración, garantiza el muestreo en un espacio de búsqueda de n -dimensiones, mientras que la Fase de Explotación permite focalizar el análisis en regiones con mayor potencial, refinando las soluciones a través de iteraciones

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

sucesivas. Esta lógica dual asegura un equilibrio entre diversidad de búsqueda y calidad de las soluciones alcanzadas.

Precisamente la etapa de adaptación y parametrización constituyó un paso imprescindible para consolidar la metodología, destacando la relevancia de la ordenación y clasificación de los elementos por medio de un índice, en el espacio de búsqueda. Estos aspectos aportan continuidad y coherencia al procedimiento, asegurando que las muestras generadas sean interpretables y comparables entre sí.

La validación del MCSA frente a funciones de referencia estándar confirmó su capacidad para generar resultados competitivos, alcanzando en algunas funciones un promedio de hasta un 90% de calidad en las soluciones obtenidas, en comparación con los resultados de métodos exhaustivos. Estos hallazgos motivaron la aplicación del MCSA a un caso práctico ampliamente estudiado: el KP01.

Los experimentos realizados confirman que el método MCSA ofrece soluciones de alta calidad con un número significativamente reducido de muestras, incluso en problemas de alta dimensión. Esta adaptabilidad también posibilita una reducción del tiempo de ejecución: observaciones de distintas ejecuciones muestran que, al aumentar el número de dimensiones, disminuye la cantidad de elementos a ordenar en cada iteración, reduciendo el tiempo relativo de ordenación y optimizando el tiempo global de cómputo.

Esta extensibilidad del método permite su aplicación a problemas multiobjetivo, optimizando múltiples objetivos sin requerir una reprogramación completa del algoritmo. En conjunto, estos resultados evidencian que MCSA mantiene la calidad de las soluciones, reduce la carga computacional y ofrece escalabilidad y aplicabilidad en problemas complejos de asignación de recursos, planificación de producción y distribución.

No obstante, el desarrollo también permitió identificar limitaciones, particularmente vinculadas al incremento de dimensionalidad y a la dispersión asociada a la maldición de la dimensionalidad. Como línea de investigación abierta, se contempla la posibilidad de paralelizar ciertos procedimientos del MCSA, por ejemplo la evaluación de múltiples soluciones simultáneamente o la

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

aplicación de clustering en distintos subconjuntos de la muestra. Este enfoque podría mejorar la eficiencia en problemas de gran escala y sensibles al tiempo de respuesta, como la gestión sanitaria, la asignación de recursos en entornos hospitalarios o la toma de decisiones en situaciones de emergencia.

En suma, este capítulo ha puesto de relieve la versatilidad de MCSA, sus fundamentos metodológicos y su capacidad de adaptación a diferentes contextos de optimización combinatoria. Sobre estas bases, los capítulos siguientes profundizarán en la ampliación de sus aplicaciones y en la consolidación de las mejoras propuestas, con el objetivo de convertir a MCSA en una herramienta robusta, generalizable y escalable para problemas de creciente complejidad.

Glosario del Capítulo

A continuación se presenta el glosario de símbolos del Capítulo 5.7, que incluye los términos y símbolos más relevantes utilizados en esta sección para facilitar la comprensión de la terminología empleada y servir de referencia rápida.

Tabla 5.7: Glosario del Capítulo

Término	Descripción
Knapsack Problem (KP01)	
n	Número de elementos disponibles para la mochila (tanto KP01 como MOKP).
v_i	Valor del elemento i .
w_i	Peso del elemento i .
x_i	Variable binaria de decisión: 1 si el elemento i se incluye, 0 en caso contrario.
W	Capacidad máxima de la mochila.
$Z(x)$	Valor total de la solución, función objetivo.
i	Índice de un elemento en el conjunto de elementos del KP01.
Multiobjective Knapsack Problem (MOKP)	
x_j	Variable de decisión binaria: 1 si el elemento j se incluye en la solución, 0 en caso contrario.
w_j	Peso o coste del elemento j .
c_j^i	Beneficio del elemento j respecto al objetivo i .
W	Capacidad máxima de la mochila.
m	Número de objetivos en el problema multiobjetivo.

Continúa en la siguiente página

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

Glosario del Capítulo (continuación)

Término	Descripción
λ_i	Peso relativo no negativo asignado al objetivo i , con $\lambda_i \geq 0$ y $\sum_i \lambda_i = 1$.
$z_i(x)$	Valor de la función objetivo i para la solución x .
$\mathbf{Z}(x)$	Vector de funciones objetivo de la solución x , $\mathbf{Z}(x) = (z_1(x), \dots, z_m(x))$.
x^1, x^2	Soluciones usadas para definir dominancia de Pareto.
$Z_\lambda(x)$	Función objetivo ponderada mediante ponderación, $Z_\lambda(x) = \sum_{i=1}^m \lambda_i z_i(x)$.

5. APLICACIÓN DE LA METODOLOGÍA Y DEL MODELO PROPUESTO: EL PROBLEMA DE LA MOCHILA

Capítulo 6

Aplicación del Modelo en la gestión del personal en un Servicio de Urgencias Hospitalarias utilizando la simulación

La optimización de recursos en entornos complejos y dinámicos constituye un problema relevante en múltiples sectores, desde la industria y la logística hasta la gestión sanitaria. En particular, los Servicios de Urgencias hospitalarias representan sistemas altamente interdependientes y estocásticos, donde la demanda de atención varía continuamente y las decisiones sobre asignación de personal tienen un impacto directo sobre la calidad del servicio y la eficiencia operativa.

Para abordar este tipo de problemas, las metodologías clásicas de optimización analítica suelen resultar insuficientes, debido a la complejidad estructural y al comportamiento emergente del sistema. En este contexto, la simulación basada en agentes (*Agent-Based Modeling*, ABM) [34, 16] se ha consolidado como una

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

herramienta capaz de modelar el comportamiento individual de los distintos actores del sistema (pacientes, médicos(as), enfermería, personal administrativo) y sus interacciones, reproduciendo de manera realista la dinámica de entornos hospitalarios.

La integración de algoritmos metaheurísticos de optimización con simuladores basados en agentes permite explorar de manera sistemática y eficiente el espacio de configuraciones posibles, identificando soluciones que optimizan múltiples indicadores de desempeño. Este enfoque combina la capacidad predictiva de la simulación con la potencia exploratoria de heurísticas avanzadas, proporcionando un marco robusto para la toma de decisiones en sistemas complejos.

En este capítulo se presenta la aplicación de la heurística MCSA para la gestión del personal en un Servicio de Urgencias Hospitalarias (ó un ED, por sus siglas en inglés), utilizando el simulador EDSimulator [61] desarrollado por el grupo de investigación *High-Performance Computing for Efficient Applications and Simulation (HPC4EAS)*. La sección siguiente justifica esta propuesta, describiendo el contexto hospitalario, la problemática a resolver y el valor añadido de la integración entre optimización heurística y simulación basada en agentes.

6.1. Justificación y contexto de la propuesta de aplicación

La gestión eficiente de los recursos humanos en los Servicios de Urgencias hospitalarias representa un problema crítico para los sistemas de salud modernos. La variabilidad impredecible de la demanda asistencial, junto con la escasez de personal y las restricciones operativas propias del entorno hospitalario, dificulta la asignación óptima de profesionales en turnos y funciones específicas. Una planificación inadecuada puede derivar en tiempos de espera prolongados, saturación de los equipos y disminución de la calidad asistencial, lo que subraya la necesidad de metodologías robustas y adaptativas para la toma de decisiones.

El uso de simulación basada en agentes (ABM) permite capturar el

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

comportamiento dinámico y las interacciones de los distintos actores dentro del Servicio de Urgencias, ofreciendo una representación fiel de la complejidad del sistema. Esta aproximación posibilita evaluar el impacto de distintas configuraciones de personal sobre indicadores críticos, como la eficiencia operativa, la ocupación de recursos y la experiencia del paciente.

Por otro lado, los problemas de asignación y planificación en entornos hospitalarios se caracterizan por espacios de soluciones amplios, no lineales y altamente restringidos, lo que hace que los métodos exactos sean computacionalmente costosos o inviables. En este contexto, la heurística MCSA surge como una herramienta adecuada para explorar de manera sistemática y eficiente las posibles configuraciones de personal, identificando soluciones de alta calidad sin necesidad de recorrer exhaustivamente todo el espacio de búsqueda.

La integración de MCSA con un simulador calibrado a partir de datos reales del Hospital Parc Taulí ofrece un valor diferencial significativo: combina la capacidad exploratoria de la optimización metaheurística con la validación empírica que proporciona la simulación, generando soluciones que son tanto óptimas desde el punto de vista computacional como viables y relevantes desde la perspectiva operativa. Esta combinación metodológica permite abordar problemas complejos de manera más segura y confiable, apoyando la toma de decisiones estratégicas basadas en evidencia [45].

En suma, la propuesta justifica su desarrollo al ofrecer un marco sólido y flexible para mejorar la asignación de personal sanitario en Servicios de Urgencias, incrementando la eficiencia, reduciendo riesgos y proporcionando una base metodológica replicable para otros entornos hospitalarios.

El segundo problema de aplicación del algoritmo MCSA se centra en la configuración óptima de los recursos humanos y físicos en un Servicio de Urgencias hospitalario. A diferencia del problema de la mochila, que es de naturaleza puramente combinatoria y discreta, este escenario presenta un espacio de búsqueda multidimensional más complejo, donde deben considerarse tanto la disponibilidad de personal como de equipamiento, y su distribución estratégica entre diferentes

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

áreas del servicio.

El espacio de soluciones está definido por múltiples dimensiones, cada una de las cuales representa la asignación de un recurso específico a una unidad funcional del servicio. Entre los recursos humanos se incluyen médicos(as), enfermeros(as) y personal técnico, mientras que los recursos físicos comprenden salas de rayos X, boxes de atención y equipamiento especializado. Dichos recursos se distribuyen entre dos áreas principales del departamento de urgencias:

- **Área A:** destinada a la atención de pacientes críticos (niveles de triaje 1, 2 y 3).
- **Área B:** orientada a pacientes menos graves (niveles de triaje 4 y 5).

Triage en urgencias - Sistema Manchester (MTS)

El triaje es el proceso de clasificación de pacientes según la urgencia de su atención médica en un Servicio de Urgencias. Su objetivo es priorizar la atención de los casos más graves, garantizando que los recursos limitados se utilicen de manera eficiente y que los pacientes reciban cuidados proporcionales a su estado clínico.

Uno de los sistemas de triaje más utilizados internacionalmente es el Sistema de Triage de Manchester (Manchester Triage System, MTS), que clasifica a los pacientes en cinco niveles de prioridad, enumerados del 1 al 5 [1], como se ve en la imagen 6.1 a continuación:

En este sistema, los colores sirven como referencia visual rápida para el personal sanitario: los pacientes rojos se atienden de inmediato, los naranjas y amarillos de manera prioritaria según disponibilidad, y los verdes y azules pueden esperar sin riesgo crítico.

Por otro lado, para evaluar la eficacia de distintas configuraciones de personal sanitario, se definen indicadores clave de desempeño (*Key Performance Indicators*, o *KPI*), que permiten cuantificar la eficiencia operativa y la carga laboral. Entre los KPI seleccionados destacan: el tiempo promedio de estancia de los

PRIORIZACIÓN DE PACIENTES

Método para la clasificación de los pacientes priorizando la gravedad y la urgencia.

NIVEL	COLOR	CLASIFICACIÓN
1	ROJO	RESUCITACIÓN: Paciente que requieren resucitación, con riesgo vital inmediato y debe recibir asistencia a su llegada.
2	NARANJA	EMERGENCIA: Situaciones de emergencia con riesgo vital y debe ser atendido inmediatamente.
3	AMARILLO	URGENTE: No se considera una emergencia pero existe riesgo potencial. Requieren múltiples exploraciones.
4	VERDE	POCO URGENTE: Considerado menos grave, el paciente no requiere atención inmediata.
5	AZUL	NO URGENTE: Consultas del nivel asistencial de Atención Primaria.

Figura 6.1: Proceso de clasificación de pacientes según la urgencia de su atención médica en un Servicio de Urgencias (elaboración propia adaptada de figura de referencia general).

pacientes (*Length of Stay*, o *LoS*), los tiempos de atención por nivel de triaje, la ocupación del personal sanitario en cada área, la saturación de recursos críticos como camas o boxes, y el número de pacientes atendidos respecto a la capacidad máxima del servicio. Estos indicadores constituyen las métricas fundamentales que retroalimentan la simulación y guían al algoritmo MCSA en la búsqueda de configuraciones de personal que optimicen simultáneamente eficiencia y sostenibilidad.

6.2. Objetivo de la propuesta

El objetivo principal del problema consiste en identificar configuraciones de recursos que permitan mejorar simultáneamente varios indicadores operativos, equilibrando la eficiencia del servicio con la carga laboral del personal. En particular, se analizan indicadores como el *LoS* y las tasas de ocupación del personal en las distintas áreas. Aunque el planteamiento podría constituir una formulación multiobjetivo estricta en sentido clásico (por ejemplo, basada en frentes de Pareto), hemos adoptado, en cambio, una perspectiva orientada al equilibrio entre métricas relevantes. Este enfoque permite interpretar los resultados desde una lógica de compromiso entre rendimiento asistencial y sostenibilidad operativa, especialmente al analizar de forma conjunta los distintos KPI(s) obtenidos para cada configuración. Por tanto, la relevancia práctica de este problema es significativa, dado que permite evaluar distintas estrategias de planificación bajo restricciones reales de disponibilidad de personal y equipamiento, proporcionando un marco para simular y analizar escenarios alternativos antes de implementar cambios en la operación diaria del hospital.

Para ello, se emplea el EDSimulator, validado con datos del Hospital Parc Taulí con el fin de evaluar distintas configuraciones de personal. Este simulador reproduce el comportamiento dinámico del Servicio de Urgencias y las interacciones entre pacientes, personal y recursos, generando KPI(s) para cada configuración evaluada. La combinación de MCSA con este modelo de simulación permite explorar un amplio espacio de soluciones de manera eficiente, identificando regiones prometedoras donde se concentran configuraciones de alto rendimiento.

Un aspecto clave del enfoque MCSA es su capacidad de reducir progresivamente el espacio de búsqueda. Inicialmente se realiza un muestreo amplio mediante técnicas de Montecarlo para explorar diversas configuraciones, seguido de una Fase de clustering que permite identificar y explotar regiones factibles donde es más probable encontrar soluciones de calidad. Esta estrategia iterativa mejora la eficiencia computacional al concentrar los esfuerzos de búsqueda en las áreas más prometedoras del espacio de soluciones.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

En conjunto, este segundo caso de estudio demuestra la versatilidad del heurístico MCSA, al aplicar la misma metodología de búsqueda a un problema de optimización multidimensional, combinando simulación realista y análisis metaheurístico para generar decisiones estratégicas con impacto operativo tangible en un entorno hospitalario de alta complejidad.

6.3. Metodología para la configuración de personal en un Servicio de Urgencias

Sistemas de información hospitalarios cada vez más avanzados proporcionan datos reales del entorno operativo, lo cual mejora el análisis y la obtención de información útil para la toma de decisiones por parte de los gestores del ED. Sin embargo, aprovechar eficazmente estos datos añade complejidad al proceso de gestión, ya que la optimización de la asignación de recursos y de los flujos de trabajo implica la Exploración de espacios de búsqueda amplios y multidimensionales. Esto convierte al problema no solo en un problema computacional, sino también organizativo, requiriendo una metodología robusta capaz de identificar soluciones factibles y eficientes.

La simulación y los métodos heurísticos constituyen estrategias complementarias ampliamente utilizadas para abordar problemas complejos de optimización en ámbitos como la logística, la ingeniería y la salud. La simulación permite modelar sistemas dinámicos mediante experimentación computacional, facilitando la evaluación de distintas alternativas en entornos controlados. Por su parte, los métodos heurísticos ofrecen enfoques prácticos para explorar espacios de soluciones de alta dimensionalidad, especialmente útiles cuando el problema supera la capacidad de los métodos analíticos tradicionales o de búsqueda exhaustiva.

Adicionalmente, la validación de nuevas políticas de personal en entornos reales suele ser inviable, lo que convierte a la simulación en una herramienta ideal para anticipar el impacto de las decisiones relacionadas con la dotación de personal, a través del análisis de KPI(s), como el tiempo de estancia de los pacientes

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

en el servicio. Cuando se combina con una búsqueda heurística, la simulación proporciona un marco sólido para explorar y evaluar estrategias bajo condiciones realistas.

En este último capítulo se presenta la metodología de búsqueda MCSA, una heurística estocástica que es capaz de explorar configuraciones de personal dentro de un espacio de búsqueda combinatorio, multidimensional y discretizado. Cada configuración representa una asignación específica de profesionales sanitarios, tanto sénior como júnior, incluyendo médicos(as), enfermeros(as) y personal de apoyo, a distintas unidades del Servicio de Urgencias. A medida que aumenta el número de variables, el espacio de soluciones crece exponencialmente, haciendo inviable el uso de métodos de búsqueda exhaustiva. Para evaluar el desempeño de cada configuración, el MCSA se apoya en un modelo de ABM validado de forma independiente, que emula el comportamiento dinámico de un Servicio de Urgencias hospitalarias bajo condiciones realistas. Este modelo ha sido desarrollado por el grupo de investigación HPC4EAS de la Universitat Autònoma de Barcelona (UAB) en colaboración con el Hospital Universitario Parc Taulí, y proporciona los valores de los KPI correspondientes a cada configuración evaluada.

Una característica fundamental del MCSA es su capacidad para reducir progresivamente el espacio de búsqueda. Inicialmente, se utiliza un muestreo aleatorio basado en el método de Montecarlo para explorar una amplia variedad de soluciones. Posteriormente, se aplica una Fase de clustering que permite identificar y explotar regiones factibles, donde tienden a concentrarse configuraciones de alta calidad. Esta refinación iterativa mejora la eficiencia computacional al enfocar la búsqueda en las zonas más prometedoras del espacio de soluciones. El contexto operativo y los datos reales utilizados en las simulaciones se basan en el funcionamiento del Hospital Taulí, que atiende a más de 160,000 pacientes anualmente [60].

6.4. Simulación como herramienta de evaluación y soporte a la optimización

El uso de técnicas de simulación en la planificación hospitalaria ha demostrado ser muy útil cuando se trata de modelar sistemas complejos, donde múltiples agentes interactúan de forma dinámica, incierta y no lineal. En el contexto específico de los Servicios de Urgencias, donde convergen pacientes con distintas prioridades, recursos humanos limitados y restricciones físicas compartidas entre áreas funcionales, la simulación ofrece una representación más realista y flexible que los modelos analíticos tradicionales.

Entre los enfoques de simulación más utilizados en entornos sanitarios se encuentran la Simulación de Eventos Discretos (DES), la Dinámica de Sistemas (SD) y la Simulación Basada en Agentes (ABM) [82]. Si bien la DES ha sido ampliamente aplicada en la gestión de colas y procesos lineales, su capacidad para capturar comportamientos emergentes, dinámicas adaptativas o interacciones complejas entre entidades individuales es limitada. La ABM, por el contrario, permite modelar actores heterogéneos con comportamientos autónomos, lo que resulta más adecuado para representar tanto la variabilidad del personal sanitario como la diversidad en los perfiles de pacientes que transitan por el ED.

En la siguiente Tabla 6.1. se enumeran algunas estrategias que utilizan la simulación como herramienta para evaluar estrategias de management bajo condiciones variables.

En la literatura, cada estrategia de simulación estudiada aporta ventajas particulares dependiendo del objetivo del estudio. La simulación DES nos sería útil si hubiésemos requerido representar procesos estructurados y flujos de trabajo definidos, como el paso de pacientes entre diferentes estaciones de atención. Sin embargo, su rigidez limita la representación de decisiones individuales y adaptativas. La SD, en cambio, permite modelar el comportamiento agregado de poblaciones o servicios completos, siendo ideal para el análisis de políticas de salud a largo plazo. No obstante, su nivel de abstracción impide capturar dinámicas

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Tabla 6.1: Comparación de técnicas de simulación en la gestión hospitalaria. En el presente estudio, se utiliza ABM implementada en NetLogo por su flexibilidad, portabilidad y capacidad de integración con entornos de alto rendimiento computacional (HPC).

Tipo de simulación	Objetivos y características principales	Herramientas conocidas
Simulación de eventos discretos (DES)	Modela sistemas mediante una secuencia de eventos que ocurren en puntos discretos en el tiempo. Utilizada para analizar flujos de pacientes, tiempos de espera, utilización de recursos y eficiencia operativa. Aporta resultados detallados en procesos lineales y bien definidos, como atención en urgencias o planificación quirúrgica. Ejemplo: optimización de salas de espera en un ED. Otro caso clásico es el estudio de recursos en unidades de cuidados intensivos [53, 19, 38].	AnyLogic, Simul8, Arena
Dinámica de sistemas (SD)	Se enfoca en modelar comportamientos agregados y tendencias a largo plazo en sistemas complejos. Usa ecuaciones diferenciales y bucles de retroalimentación para estudiar cambios en políticas de salud, gestión de camas o impacto de epidemias. No modela individuos sino poblaciones agregadas. Ejemplo: evaluación del efecto de políticas de contratación a largo plazo. Otro ejemplo es el análisis de expansión de capacidad hospitalaria durante una pandemia [88].	Vensim, Stella, Powersim
Simulación basada en agentes (ABM)	Modela la interacción entre agentes autónomos (pacientes, médicos(as), enfermeros(as) y su entorno. Captura comportamientos heterogéneos, toma de decisiones local y fenómenos emergentes. Es ideal para simular sistemas en los cuales, las reglas locales producen dinámicas globales no triviales. Ejemplo: El desarrollo de un modelo estocástico basado en agentes para pronosticar la disponibilidad futura de médicos(as) en Portugal, mostrando cómo pequeñas variaciones en parámetros de entrada, como tasas de emigración o abandono, pueden generar resultados muy diferentes. Este estudio utiliza ABM implementada en NetLogo por su flexibilidad, portabilidad y capacidad de integración con entornos de alto rendimiento computacional (HPC) [10]. En este sentido, la incorporación de agentes autónomos permite integrar estos aspectos de forma más realista y responder con mayor validez a cuestiones de planificación económica y organizativa [87].	NetLogo, Repast, AnyLogic

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

individuales críticas en servicios como urgencias.

Por su parte, la ABM se ha consolidado como una herramienta para analizar la interacción entre múltiples actores con comportamientos heterogéneos. Esta metodología permite incluir variabilidad individual, adaptabilidad y fenómenos emergentes, aspectos esenciales en contextos hospitalarios donde las decisiones de cada profesional o paciente pueden alterar el sistema global. En este trabajo se ha optado por ABM como estrategia de simulación debido a su capacidad de capturar estas dinámicas, y se ha utilizado la herramienta NetLogo, por ser una plataforma validada, de bajo coste computacional, portable y compatible con procesos paralelos en entornos de alto rendimiento (HPC). NetLogo además ofrece una sintaxis accesible y una comunidad activa, facilitando la replicabilidad y la expansión del modelo. Para trabajar en combinación con el algoritmo heurístico, NetLogo se integra mediante un módulo automatizado que permite la Exploración eficiente del espacio de búsqueda, optimizando asignaciones de personal en un entorno realista y validado empíricamente.

Simulación basada en agentes para la optimización de recursos en Servicios de Urgencias

Aunque la simulación DES es un enfoque estocástico ampliamente utilizado para modelar sistemas dinámicos y complejos, como los del sector salud, su aplicación práctica sigue estando limitada. En una revisión sistemática reciente de 231 estudios centrados en DES en salud [99], se identificó que aproximadamente la mitad de las aplicaciones se enfocan en mejorar métricas relacionadas con el tiempo y la eficiencia operativa, y que el área más común de implementación es el departamento de emergencias. Además, se observó una tendencia creciente hacia el uso de modelos híbridos que combinan DES con otras técnicas analíticas, como optimización o estrategias de mejora de procesos operativos.

No obstante, uno de los hallazgos más relevantes de dicha revisión es que menos del 10% de las aplicaciones basadas en simulación DES culminan con una implementación real tras la etapa de modelado. Esta brecha entre modelado teórico

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

y aplicación práctica pone en evidencia las limitaciones del enfoque, especialmente cuando se trata de representar la autonomía de los actores involucrados, su adaptación al entorno o el surgimiento de comportamientos colectivos.

Por otro lado, la SD ofrece una visión macro útil para analizar políticas de salud a largo plazo, pero al trabajar con variables agregadas no resulta adecuada para capturar la toma de decisiones individuales ni las interacciones específicas entre agentes, aspectos fundamentales en entornos críticos como los ED. En este sentido, mientras que DES se enfoca en procesos operativos locales, SD permite capturar interdependencias entre servicios y analizar impactos a nivel macro. A diferencia de la simulación de eventos discretos, que suelen centrarse en procesos operativos locales, la dinámica de sistemas permite capturar interdependencias entre servicios y analizar impactos a nivel macro. Por ejemplo, el estudio de [66] demostró cómo un modelo SD pudo identificar cuellos de botella reales en un Servicio de Urgencias, incluyendo momentos en los que agregar personal no mejoraba el flujo de pacientes debido a restricciones aguas abajo.

En este trabajo se ha seleccionado la plataforma **NetLogo** para desarrollar la simulación por varias razones: (i) existe un simulador de urgencias previamente implementado y validado sobre esta herramienta, (ii) permite realizar múltiples ejecuciones de manera paralela, (iii) es compatible con entornos de computación de alto rendimiento (*High-Performance Computing* - HPC), y (iv) su portabilidad facilita la ejecución en diversos entornos, desde ordenadores personales hasta *clusters* distribuidos.

La simulación ABM, proporciona ventajas clave para el tipo de optimización propuesta:

- Capacidad de representar comportamientos autónomos y adaptativos de los distintos agentes (pacientes, médicos(as), enfermeros(as), técnicos(as)), lo que permite capturar mejor la dinámica real del servicio.
- Flexibilidad para modelar la interacción entre recursos humanos y físicos, incluyendo la compartición entre distintas áreas funcionales del servicio (p. ej., áreas A y B con niveles de severidad diferenciados).

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

- Simplicidad de implementación y portabilidad del simulador, desarrollado en NetLogo, una plataforma ligera y abierta que permite realizar experimentos tanto en máquinas locales como en sistemas HPC, lo cual es especialmente relevante dadas las necesidades computacionales del problema.
- Facilidad para ejecutar múltiples simulaciones simultáneamente, lo que permite analizar configuraciones bajo condiciones estocásticas y extraer métricas robustas como el *LoS*, la utilización de recursos o la congestión en áreas críticas.
- Validación del simulador en colaboración con un entorno hospitalario real, lo cual incrementa la relevancia práctica y la aplicabilidad de los resultados obtenidos.

Además, como parte del desarrollo de este trabajo, se ha llevado a cabo la implementación de un módulo que automatiza la integración entre el modelo heurístico propuesto y el simulador de NetLogo, permitiendo que cada solución generada sea evaluada automáticamente a través de múltiples escenarios. Este módulo opera como un orquestador que lanza simulaciones, recopila los KPI y retroalimenta al heurístico para guiar la Exploración del espacio de búsqueda. De esta manera, la simulación no solo actúa como herramienta de análisis posterior, sino como parte activa del ciclo de optimización.

Esta estrategia potencia la Exploración del espacio de búsqueda y permite una optimización basada en simulación que favorece soluciones robustas y replicables, adaptadas a entornos de alta incertidumbre y dinámica compleja que contribuye directamente a una mejor toma de decisiones.

Finalmente, esta propuesta de optimización guiada por simulación permite abordar la complejidad combinatoria del problema sin necesidad de simplificaciones excesivas. Así, el proceso de búsqueda no solo se dirige hacia soluciones factibles, sino también hacia aquellas que presentan un buen desempeño en condiciones realistas.

6.5. Hipótesis

El aumento sostenido del flujo de pacientes en los ED constituye un problema constante en la gestión de recursos, impactando directamente en los tiempos de espera y en la calidad del servicio. A pesar de los avances en modelado y simulación, las estrategias tradicionales de asignación de personal suelen basarse en reglas fijas o políticas estáticas que no se adaptan de manera eficiente a la variabilidad y la incertidumbre de la demanda asistencial. Esta problemática plantea la necesidad de enfoques que combinen técnicas de búsqueda estocástica con simulación basada en agentes, permitiendo explorar de manera coordinada espacios de soluciones complejos y altamente dimensionales en la gestión sanitaria.

En este marco, se propone la aplicación del algoritmo de búsqueda heurística MCSA, diseñado para identificar configuraciones de personal eficientes sin recurrir a métodos de búsqueda exhaustiva. Las hipótesis de trabajo que guían este enfoque son las siguientes:

- **Hipótesis 1:** La asignación óptima de configuraciones de personal en el ED puede reducir significativamente el tiempo de permanencia de los pacientes, medido a través del indicador *LoS*.
- **Hipótesis 2:** El algoritmo MCSA es capaz de explorar eficientemente un espacio de soluciones de alta dimensionalidad, respetando las restricciones operativas y de disponibilidad de recursos reales.
- **Hipótesis 3:** La heurística MCSA ofrece ventajas computacionales frente a métodos de búsqueda exhaustiva, permitiendo obtener soluciones de calidad en tiempos de cómputo razonables, al optimizar KPI(s) relacionados con la asignación de personal.

Para la validación de estas hipótesis, se simularon escenarios representativos del entorno del Servicio de Urgencias, donde MCSA propone configuraciones de personal que son evaluadas mediante el EDSimulator previamente calibrado con datos reales. Esta integración permite no solo explorar el espacio de soluciones de manera eficiente, sino también alinear las decisiones de asignación de recursos

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

con objetivos operativos concretos, proporcionando una herramienta robusta para la toma de decisiones estratégicas y la mejora de la eficiencia operativa en entornos de alta complejidad. La metodología busca demostrar que es posible encontrar soluciones efectivas sin necesidad de explorar completamente el espacio combinatorio, y que la retroalimentación basada en los resultados de la simulación permite alinear las decisiones con objetivos operativos reales.

6.6. Estado del Arte

La optimización en el contexto de los EDs ha sido ampliamente abordada mediante técnicas de búsqueda heurística [25, 94, 57] y metodologías de simulación, cada una con ventajas particulares.

Los métodos heurísticos, entre los que se incluyen algoritmos evolutivos y metaheurísticas como los Algoritmos Genéticos (GA), la Optimización por Enjambre de Partículas (PSO) o el Recocido Simulado (SA), entre otros [2, 100], han sido utilizados de forma extensiva para resolver problemas complejos de planificación de personal en el ámbito sanitario. Estas aproximaciones, si bien eficaces para identificar soluciones cercanas al óptimo, tienden a centrarse en componentes aislados del problema, como la asignación de turnos a enfermería o la programación sensible al cansancio bajo supuestos estáticos. Por ejemplo, Ceschia et al. [24] propusieron un modelo de Recocido Simulado que incorpora habilidades y preferencias, aplicado en hospitales reales en Italia. De forma similar, Amindoust et al. [9] desarrollaron un algoritmo evolutivo híbrido que considera la fatiga en contextos pandémicos, mientras que Valoux et al. [96] introdujeron un enfoque evolutivo multiobjetivo que equilibra la seguridad, la calidad del servicio y el bienestar del personal.

No obstante, una limitación común en estos enfoques reside en su escasa capacidad para operar con datos operacionales o dinámicas de sistemas en tiempo real. En su mayoría, dependen de conjuntos de datos sintéticos o restricciones simplificadas, lo que dificulta su adaptación a entornos reales y de alta dimensionalidad como los EDs.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

En cuanto a la simulación, se han propuesto diversas técnicas para modelar los flujos de trabajo hospitalarios. La DES se ha empleado para analizar el flujo de pacientes y el rendimiento de colas [30, 97], mientras que la SD se centra en tendencias agregadas, proporcionando información útil para evaluar el impacto de políticas a largo plazo. Sin embargo, ambos enfoques carecen del nivel de granularidad necesario para evaluar las consecuencias operativas de ajustes a nivel micro en la dotación de personal. En contraste, la ABM [93] modela interacciones individuales entre pacientes, profesionales y recursos, capturando comportamientos emergentes en sistemas complejos. El uso de ABS en entornos de urgencias, como se muestra en los trabajos de Cabrera et al. [21] y Bruballa et al. [20], ha demostrado su capacidad para reflejar la variabilidad del sistema y apoyar la evaluación de escenarios.

Sin embargo, estos estudios basados en simulación suelen limitarse a la evaluación de escenarios predefinidos sin incorporar mecanismos activos de optimización de configuraciones. Si bien proporcionan información valiosa sobre el rendimiento del sistema, no integran procesos de búsqueda para la mejora de configuraciones.

Con el objetivo de superar estas limitaciones, estudios recientes han comenzado a explorar modelos híbridos que combinan simulación y optimización heurística. Estos enfoques aprovechan el realismo de la simulación junto con la capacidad exploratoria de las heurísticas, buscando configuraciones eficientes sin sacrificar la fidelidad del sistema. Las aplicaciones van desde la planificación de desinfección hospitalaria [52] hasta la logística dinámica [101, 95], demostrando su versatilidad en entornos complejos. No obstante, muchos de estos marcos híbridos se centran en dimensiones limitadas o carecen de mecanismos formales para explorar grandes espacios de soluciones.

A diferencia de los modelos anteriores, ya sean impulsados por búsqueda bajo supuestos estáticos o centrados en simulación sin optimización, la presente propuesta introduce una estrategia coordinada: un proceso de búsqueda heurística (por medio de MCSA) guiado por retroalimentación del EDSimulator, que ha sido validado de forma independiente y construido a partir de un ED real. Esta metodología aborda la multidimensionalidad inherente a los EDs,

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

considerando múltiples roles profesionales, e incorpora restricciones como los límites mínimos/máximos de personal y las dependencias entre roles. El método propuesto no se limita a simular para obtener KPI(s), sino que refina activamente las configuraciones mediante un bucle de retroalimentación basado en dichos indicadores. Esta contribución se diferencia de la literatura existente en aspectos clave:

1. La Exploración estocástica se basa en evaluaciones de simulación, proporcionando tanto calidad de solución como fidelidad al sistema.
2. Aborda la dimensionalidad y las restricciones reales, validado con datos operacionales del Hospital Taulí, un ED de alto volumen en España.
3. Evita la necesidad de búsquedas exhaustivas, ofreciendo un enfoque escalable y trazable para la toma de decisiones en contextos críticos.

Aunque esta aplicación se enfoca en el ámbito sanitario, el marco metodológico subyacente, es decir, la coordinación heurística - simulación híbrida, ha demostrado ser útil en otros dominios como la planificación de manufactura, las telecomunicaciones y la respuesta ante desastres. Al reconocer estos contextos más amplios, buscamos situar nuestra propuesta dentro de una tradición más extensa de resolución de problemas complejos de asignación mediante métodos flexibles. Estas contribuciones posicionan nuestro trabajo en la intersección entre la búsqueda heurística avanzada y la simulación de alta fidelidad, ofreciendo un marco novedoso para la optimización de personal basada en datos.

En contraste con optimizadores generalistas recientes, como el algoritmo RUN [5], que aprovecha estrategias matemáticas como el método de Runge-Kutta para ir más allá de heurísticas basadas en metáforas [83], este enfoque es específico del dominio y diseñado para su aplicabilidad práctica en entornos de alta dimensionalidad y guiados por simulación, como los EDs. Aunque los algoritmos sin metáforas y con fundamentos teóricos han mostrado eficacia en diversos contextos ingenieriles [7], su uso es limitado cuando se requiere la evaluación de soluciones mediante simulaciones complejas y estocásticas. Nuestra metodología no busca superar otros algoritmos de optimización, sino proporcionar a los responsables

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

sanitarios un marco escalable para la planificación operativa realista.

Nuestro enfoque no asume un modelo paramétrico predefinido; en su lugar, emplea una heurística (MCSA) guiada por evaluaciones de simulación para identificar configuraciones viables de personal bajo condiciones complejas, dinámicas e inciertas. A diferencia de modelos comunes en áreas como la genética poblacional o la epidemiología, nuestra metodología se enfoca en la asignación de recursos locales (p. ej., médicos(as), enfermeros(as)) y la optimización de métricas de rendimiento. Esta diferencia en escala y alcance subraya la naturaleza de nuestro trabajo: proporcionar herramientas de optimización táctica para responsables hospitalarios que buscan mejorar la eficiencia en entornos altamente variables.

Para ilustrar la estructura y dinámica del modelo de simulación empleado en este estudio, la Figura 6.2 muestra un esquema del simulador de ED utilizado conjuntamente con nuestra metodología. Este simulador es tratado como un sistema calibrado, previamente validado con datos empíricos del Hospital Taulí. Opera con dos tipos de entradas:

- **Entradas externas:** Representan elementos reales fuera del control de los responsables de planificación, como los patrones de llegada de pacientes y su distribución temporal. Estos valores se derivan de datos históricos y reflejan variaciones observadas por hora, día o estación del año.
- **Entradas internas:** Definidas por el usuario y que representan la configuración del personal médico (por ejemplo, número de médicos(as) senior y junior, enfermeros(as) y técnicos(as)). Estas variables constituyen el espacio de búsqueda sobre el cual opera nuestro algoritmo heurístico (MCSA).

En cuanto a las salidas, el simulador proporciona:

- Una lista de KPI(s), que incluyen el LoS, la ocupación del personal y las colas de pacientes en admisión y triaje. Estas métricas se utilizan para evaluar la calidad de cada configuración.
- Otras salidas, como el comportamiento de derivación de pacientes simulado,

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

que reflejan la dinámica del sistema aunque no se utilicen directamente en nuestra metodología.

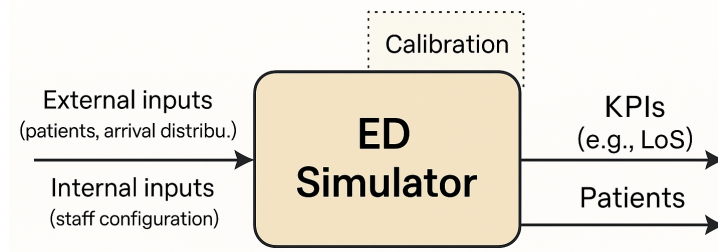


Figura 6.2: Este esquema ilustra cómo la simulación desempeña un papel clave al conectar la búsqueda abstracta (MCSA) con la evaluación del rendimiento en el mundo real.

6.7. El heurístico: aplicación del MCSA al problema de gestión del personal sanitario

El algoritmo *MCSA*, desarrollado y descrito en detalle en capítulos anteriores, es una metaheurística concebida para resolver problemas de optimización combinatoria de alta dimensionalidad. En este capítulo, se presenta su aplicación particular al problema de asignación de personal en un ED, una de las instancias más complejas abordadas mediante dicha metodología.

En este contexto, cada solución candidata representa una configuración discreta de recursos humanos y materiales (por ejemplo, número de médicos(as) senior y junior, enfermeros(as) y técnicos(as)), cuya viabilidad y rendimiento son evaluados a través de un modelo de simulación basado en agentes. Esta simulación, calibrada y validada con datos reales del Hospital Taulí, devuelve KPI(s) que son utilizados como base para guiar el proceso de optimización.

La adaptación del MCSA a este entorno conserva la estructura original del algoritmo, compuesta por dos fases principales:

- **Fase de Exploración:** Se utiliza una estrategia de muestreo de Montecarlo para generar un conjunto diverso de configuraciones en el espacio de

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

búsqueda discreto de n dimensiones. Cada una de estas configuraciones es evaluada mediante simulación, y su calidad se determina a partir de una función de aptitud construida con los valores de los KPI(s), tales como el *LoS*, la ocupación del personal o colas en admisión y triaje.

- **Fase de Explotación:** Se activa un mecanismo de agrupamiento que permite refinar la búsqueda, identificando regiones densas en el espacio de soluciones. En primer lugar, un filtro basado en densidad detecta subregiones prometedoras, que luego se exploran con mayor precisión mediante un agrupamiento centrado en centroides, enfocando el muestreo en torno a configuraciones especialmente prometedoras.

Cabe recordar que el MCSA ha sido validado en trabajos previos [43] utilizando funciones benchmark clásicas, en los que demostró una elevada capacidad de convergencia, repetibilidad y escalabilidad. Asimismo, ha sido aplicado exitosamente a problemas de optimización discreta como el problema de la mochila binaria (KP01) [42, 77], lo cual confirma su flexibilidad ante distintas estructuras de búsqueda.

En comparación con otros algoritmos recientes, como el *Gradient-Based Optimizer* (GBO) [4] o el algoritmo *Weighted Mean of Vectors* (INFO) [6], MCSA presenta una ventaja diferencial para problemas donde las funciones objetivo no están explícitamente definidas, sino que emergen del comportamiento estocástico de un sistema simulado. En este tipo de entornos, como ocurre en los EDs reales, el foco del algoritmo no es la eficiencia matemática en términos abstractos, sino su aplicabilidad operativa bajo condiciones inciertas, no lineales y sujetas a múltiples restricciones.

Por ello, en este capítulo se destaca el uso del MCSA como herramienta de apoyo a la toma de decisiones en contextos sanitarios complejos. La meta no es superar algoritmos generalistas en benchmarks estandarizados, sino ofrecer un marco robusto, escalable y trazable para planificadores hospitalarios que deben tomar decisiones basadas en datos, dentro de sistemas con alta variabilidad y presión asistencial constante (ver overview Figura 4.1).

6.7.1. Integración de MCSA con el simulador de urgencias

Como se ha desarrollado en capítulos anteriores, el algoritmo *Montecarlo Clustering Search Algorithm* (MCSA) constituye una herramienta heurística robusta para la resolución de problemas de optimización de alta dimensionalidad. A lo largo de este capítulo, se aborda su integración con nuestro simulador basado en agentes EDSimulator específicamente diseñado para representar el funcionamiento de un Servicio de Urgencias (ED), con el fin de explorar y generar configuraciones óptimas de personal bajo condiciones realistas y operativas.

La estrategia híbrida implementada permite combinar el poder exploratorio estocástico del MCSA con la capacidad del simulador para capturar la complejidad dinámica y no lineal del entorno hospitalario. Esta integración tiene como objetivo principal la generación de soluciones viables y aplicables para la asignación de recursos humanos, priorizando la mejora de indicadores clave de rendimiento, en particular el tiempo de estancia de los pacientes (*LoS*), un KPI fundamental en la gestión asistencial de urgencias.

El simulador utilizado, y validado tal como se documenta en [90], reproduce de forma detallada el comportamiento del sistema ante diferentes configuraciones de dotación de personal y cargas de pacientes. Este modelo considera tanto entradas externas (por ejemplo, distribución temporal de llegadas de pacientes) como entradas internas (asignación de médicos(as), enfermería y técnicos(as)), lo que permite una evaluación realista y granular del impacto de cada configuración en el rendimiento global del ED.

La interacción entre MCSA y el simulador se articula mediante un ciclo iterativo de retroalimentación. En cada iteración del proceso, MCSA genera un conjunto de soluciones candidatas, es decir, cada muestra representa configuración específica de personal, que es evaluada por el simulador. Este devuelve un conjunto de KPI(s) que cuantifican el comportamiento del sistema bajo dicha configuración. A partir de estos valores, el algoritmo ajusta su búsqueda, concentrándose progresivamente en regiones prometedoras del espacio de soluciones.

Tal como se ilustra en el Diagrama 6.3, este proceso forma un bucle cerrado donde

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

la simulación informa directamente al heurístico, y el heurístico explora el espacio de soluciones en función del desempeño simulado. Esta retroalimentación continua permite guiar la búsqueda de forma informada, adaptativa y contextualizada, sin recurrir a modelos matemáticos analíticos o funciones objetivo simplificadas.

Esta aplicación demuestra cómo el MCSA puede integrarse eficazmente en entornos complejos y estocásticos, como los que caracterizan la operación de un Servicio de Urgencias, proporcionando un marco de optimización sensible a las restricciones del mundo real y al comportamiento emergente del sistema. La propuesta no se limita a identificar soluciones de forma abstracta, sino que incorpora directamente la dinámica del entorno, ofreciendo a los responsables de planificación hospitalaria una herramienta robusta para la toma de decisiones informadas y basadas en evidencia simulada.

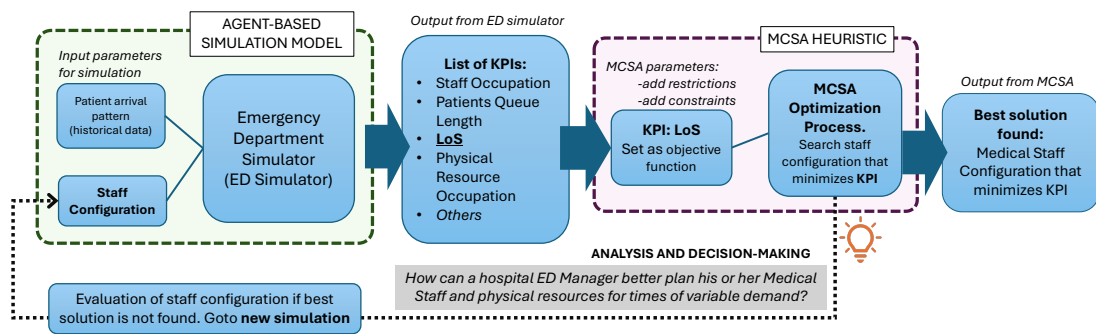


Figura 6.3: Diagrama de flujo metodológico que muestra la interacción entre el modelo basado en agentes (ABM) y el algoritmo MCSA. El esquema representa el bucle de retroalimentación en el que MCSA propone configuraciones de personal como entrada al simulador. Este realiza la simulación de cada configuración y genera los indicadores clave de rendimiento (KPI(s)), los cuales guían la siguiente iteración del proceso de Exploración. La estructura refleja la lógica central del marco de optimización.

6.7.2. Formulación del problema: Fase de Exploración mediante MCSA

La configuración óptima del personal en un Servicio de Urgencias se plantea, en este capítulo, como un problema de optimización combinatoria, en el que deben

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

asignarse de manera adecuada tanto recursos humanos como físicos. Cada solución candidata se representa mediante un vector de variables discretas de decisión, donde cada componente del vector hace referencia a un tipo específico de recurso (por ejemplo, número de médicos(as), enfermeros(as), técnicos(as)) o a elementos físicos críticos (como boxes de atención, máquinas de rayos X o unidades de soporte vital).

La elevada dimensionalidad del problema, unida a la existencia de restricciones interdependientes entre variables, da lugar a un espacio de búsqueda extenso, no lineal y altamente complejo. Además, debido al comportamiento estocástico del flujo de pacientes y a la interacción no lineal entre recursos, la formulación analítica de una función objetivo cerrada resulta impracticable. A fin de explorar de manera eficiente este espacio, el algoritmo MCSA emplea una estrategia de muestreo estocástico basada en Montecarlo, con el objetivo de generar configuraciones viables y diversas en las primeras etapas del proceso.

Cada punto muestreado corresponde a una posible asignación de recursos, generada dentro de límites predefinidos que reflejan las restricciones operativas reales del entorno hospitalario. Estas restricciones incluyen tanto la disponibilidad efectiva de personal como las capacidades físicas del servicio (por ejemplo, número máximo de boxes o puestos de triaje simultáneo). Las configuraciones generadas son simuladas mediante el EDSimulator, lo cual permite al algoritmo estimar el impacto operativo de cada asignación a partir de KPI(s).

El objetivo principal de la optimización en esta fase consiste en minimizar el tiempo medio de estancia de los pacientes (LoS), reconocido en la literatura como un indicador crítico de eficiencia asistencial y calidad del servicio [68, 67]. Para asegurar la viabilidad de las soluciones generadas, las restricciones del problema, tales como la disponibilidad del personal, las limitaciones de espacio físico y las reglas operativas del servicio, se han incorporado directamente en los rangos de muestreo. De este modo, el algoritmo evita regiones no factibles del espacio de búsqueda, centrándose únicamente en configuraciones potencialmente implementables.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Esta formulación estructurada y guiada de la Fase de Exploración permite que MCSA actúe sin conocimiento previo desde el inicio del proceso, pero optimizando recursos en un entorno realista y sujeto a múltiples restricciones. Además, la metodología facilita una integración fluida con la posterior Fase de Explotación, en la que se refina la búsqueda sobre las regiones más prometedoras identificadas en esta etapa inicial.

Variables de decisión y espacio de búsqueda

Cada configuración candidata puede representarse mediante un vector de decisión discreto:

- $\mathbf{x} = (x_1, x_2, \dots, x_n)$: Vector de variables discretas, donde cada x_j indica la cantidad de personal asignado al rol j (por ejemplo, médicos(as) senior, enfermeros(as) junior, técnicos(as) de triaje).
- Ω : Región factible definida por restricciones operativas, tales como disponibilidad de personal, normativa interna y límites estructurales:

$$x_j^{\min} \leq x_j \leq x_j^{\max}, \quad \forall j = 1, \dots, n \quad (6.1)$$

Estas cotas se establecen a partir de datos reales observados en el entorno hospitalario (como turnos disponibles o máximos por categoría profesional), permitiendo restringir el espacio de búsqueda únicamente a configuraciones implementables.

Evaluación mediante simulación

Para cada configuración $\mathbf{x}$ generada por el algoritmo de búsqueda, el simulador ejecuta un conjunto de N réplicas independientes mediante Montecarlo, a fin de capturar la variabilidad inherente al sistema. Cada réplica devuelve un valor de LoS :

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

$$\text{LoS}_{\text{MCN}}(\mathbf{x}) = \{\text{LoS}_1, \text{LoS}_2, \dots, \text{LoS}_N\} \quad (6.2)$$

La media de estos resultados representa el rendimiento esperado de la configuración $\mathbf{x}$:

$$\mu_{\text{LoS}}(\mathbf{x}) = \frac{1}{N} \sum_{j=1}^N \text{LoS}_j \quad (6.3)$$

Este valor medio constituye la función objetivo utilizada por el MCSA para guiar la búsqueda:

$$f(\mathbf{x}) = \mu_{\text{LoS}}(\mathbf{x}) \quad (6.4)$$

Problema de optimización

A partir de lo anterior, el problema de optimización puede formularse de la siguiente manera:

$$\min_{\mathbf{x} \in \Omega} f(\mathbf{x}) = \mu_{\text{LoS}}(\mathbf{x}) \quad (6.5)$$

Es decir, se busca la configuración $\mathbf{x}^* \in \Omega$ que minimice el tiempo medio de estancia de los pacientes, respetando las restricciones definidas para cada variable (ver [6.1](#)):

$$x_j^{\min} \leq x_j \leq x_j^{\max}, \quad \forall j = 1, \dots, n \quad (6.6)$$

Relación entre simulación y optimización

Desde un punto de vista conceptual, el modelo de simulación puede expresarse como una función que relaciona los KPI(s) con los parámetros de entrada:

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

$$\text{KPI} = f(\text{parámetros internos}, \text{parámetros externos})$$

En este marco:

- Los **parámetros internos** son fijos y forman parte del diseño del simulador, tales como las distribuciones de tiempos de atención, reglas de enrutamiento, prioridades de triaje, entre otros.
- Los **parámetros externos** corresponden al vector de decisión $\mathbf{x}$, y son las variables que el algoritmo MCSA modifica durante el proceso de optimización (por ejemplo, número de médicos(as), enfermeros(as) o técnicos(as) asignados a cada turno).

A partir de los KPI(s) generados por el simulador para cada configuración $\mathbf{x}$, se aplica una función de evaluación g que sintetiza el desempeño:

$$y = g(\text{KPI})$$

En este caso, la función g consiste en calcular la media del *LoS* sobre las N réplicas de simulación. Este valor resume el impacto operativo de cada configuración y permite comparar alternativas en función de su efectividad relativa.

En conjunto, esta formulación metodológica permite integrar directamente la simulación con el proceso de búsqueda heurística, garantizando que las decisiones propuestas estén fundamentadas en una evaluación empírica robusta, adaptada al contexto real del sistema hospitalario bajo estudio.

El proceso completo descrito garantiza que cada evaluación proporcione una estimación estadísticamente robusta del rendimiento de la configuración analizada, permitiendo comparaciones significativas entre alternativas, a pesar de la naturaleza estocástica del sistema simulado. La repetición de simulaciones mediante Montecarlo asegura que los KPI(s) se mantengan consistentes entre ejecuciones, aportando estabilidad frente a la variabilidad inherente de este tipo de entornos operativos.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

La Figura 6.4 ilustra el procedimiento de muestreo llevado a cabo por el algoritmo MCSA dentro del espacio multidimensional de decisión, representado aquí de forma simplificada en dos dimensiones para facilitar la visualización conceptual. Aquí, cada eje representa una variable de decisión discreta, como puede ser el número de médicos(as) o enfermeros(as) asignados. Debido a la estructura combinatoria del problema, el tamaño del espacio de búsqueda crece exponencialmente conforme se incrementa el número de variables o se amplían sus respectivos dominios, lo que pone de manifiesto la complejidad intrínseca de la tarea de optimización.

La misma figura muestra cómo el algoritmo genera una muestra de configuraciones candidatas, cada una de las cuales es evaluada individualmente mediante el simulador del ED. Este proceso devuelve una serie de KPI(s) que informan al algoritmo sobre el desempeño de cada solución, orientando así los siguientes pasos de la búsqueda. Esta dinámica de muestreo y evaluación se repite de forma iterativa hasta que el algoritmo converge hacia una distribución estable de soluciones, que puede interpretarse como un mapa de Montecarlo del espacio de búsqueda, con regiones identificadas como más prometedoras desde el punto de vista operativo.

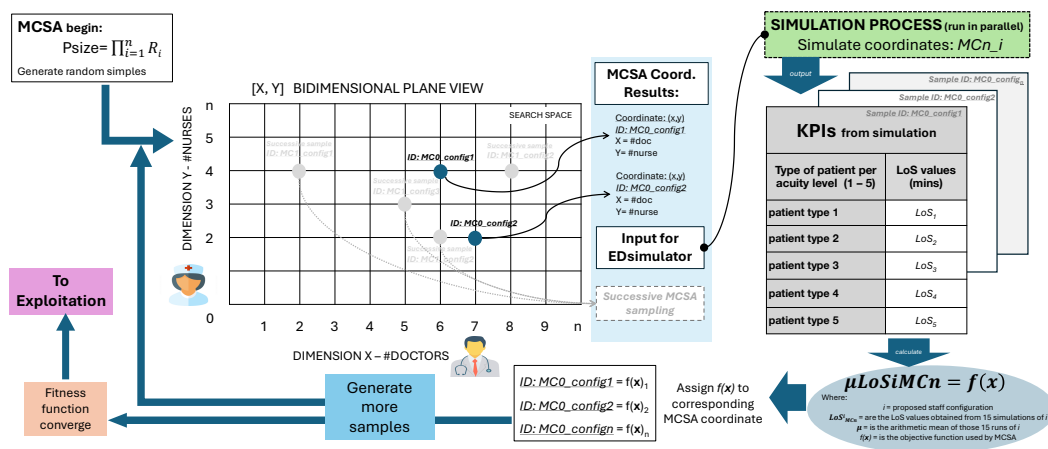


Figura 6.4: Diagrama que ilustra el proceso de muestreo realizado por el algoritmo MCSA para generar configuraciones de personal, seguido por su evaluación mediante el modelo basado en agentes (ABM), el cual devuelve los valores correspondientes de los indicadores clave de rendimiento (KPI(s)). MCSA genera y refina iterativamente nuevas muestras hasta alcanzar la convergencia.

A partir de la simulación en el EDSimulator se obtiene un conjunto de valores de

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

KPI(s), y tomamos en este caso el LoS como referencia. Tras ejecutar múltiples simulaciones independientes para cada configuración propuesta por MCSA, se promedian estos valores para calcular la función objetivo $f(\mathbf{x})$, cuyo resultado es devuelto al entorno del algoritmo MCSA, permitiéndole asignar una puntuación de desempeño a dicha muestra específica en la forma de un índice que corresponderá con su valor de μ_{LoS} .

Este bucle iterativo de retroalimentación se implementa según lo mostrado en el pseudocódigo del Algoritmo 2, que describe la función encargada de orquestar las llamadas al simulador, gestionar la espera de resultados y recuperar el valor medio del KPI. Esta función es imprescindible en el proceso de optimización, ya que permite guiar la búsqueda progresivamente hacia regiones factibles del espacio de soluciones, en función del rendimiento observado.

Algorithm 2 `get_func_val(samples)`

```
1: if number_of_generated_samples < SamplesPerItMC then
2:   if list is valid according to verify_list (restrictions) then
3:     Append list to ListCoord
4:   end if
5: end if
6: if number_of_generated_samples = SamplesPerItMC then
7:   Write ListCoord to disk using WriteEDSamples
8:   Initialize empty list of processes
9:   Build command to execute all samples with EdLauncher.sh to run  $N$  ED simulators
10:  Launch the command
11:  Initialize empty KPIList
12:  for each process launched do
13:    Wait until it finishes
14:    for each sample processed do
15:      Process sensor file to obtain queues, occupancy, and  $LoS$ 
16:      Compute KPI using SelectKPI
17:      Append KPI to KPIList
18:    end for
19:  end for
20:  Increment Montecarlo iteration number (HIteration)
21:  return ListCoord, KPIList
22: else
23:   return None, None
24: end if
```

Una vez finalizada la simulación, los valores de los KPI(s) obtenidos en cada ejecución se agregan en un único valor de función asociado al correspondiente

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

vector de configuración $\mathbf{x}^{(i)}$. Específicamente, para cada configuración candidata, el simulador se ejecuta $N = 15$ veces con el fin de capturar la variabilidad estocástica del sistema.

En cada una de estas 15 simulaciones, se calcula el LoS por separado para cada uno de los cinco tipos de pacientes definidos en el modelo. A continuación, estos cinco valores se combinan mediante un promedio ponderado, produciendo así un único valor representativo de LoS por ejecución.

Denotamos el conjunto de estos 15 valores representativos como:

$$LoS_{MC15}(\mathbf{x}^{(i)}) = \{LoS_1^{(i)}, LoS_2^{(i)}, \dots, LoS_{15}^{(i)}\} \quad (6.7)$$

El KPI global, denotado como $\mu LoS_{KPI}^{(i)}$, se define como la media aritmética de estos valores:

$$\mu LoS_{KPI}^{(i)} = \frac{1}{15} \sum_{j=1}^{15} LoS_j^{(i)} \quad (6.8)$$

Este valor agregado representa el rendimiento esperado de la configuración $\mathbf{x}^{(i)}$ y se asigna como valor de la función objetivo correspondiente:

$$f(\mathbf{x}^{(i)}) = \mu LoS_{KPI}^{(i)} \quad (6.9)$$

Esta formulación asegura que cada configuración sea evaluada de manera equitativa y comparable bajo condiciones de incertidumbre, proporcionando así una métrica robusta $f(\mathbf{x})$ que guía el proceso de búsqueda y optimización.

El tiempo de simulación, en este caso, constituye un factor clave en la integración entre la heurística y el modelo del sistema. Dado que la evaluación de cada configuración requiere un coste computacional elevado, el MCSA selecciona estratégicamente un número limitado de muestras, optimizando así el uso de los recursos disponibles. Mediante la gestión del número de evaluaciones y la ejecución simultánea de múltiples configuraciones en modo *batch*, MCSA garantiza un proceso de búsqueda eficiente sin necesidad de realizar una Exploración exhaustiva

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

del espacio de soluciones.

Durante la simulación, los agentes que representan al personal y los recursos interactúan con el flujo de pacientes y desencadenan eventos que influyen directamente en los indicadores clave de rendimiento (KPI(s)). Este proceso iterativo pone de manifiesto cómo la variabilidad en la asignación de recursos afecta dinámicamente al comportamiento del sistema a lo largo del tiempo, lo que subraya la necesidad de optimizar las configuraciones de personal para mantener la eficiencia operativa del servicio.

Criterio de convergencia del muestreo

Para evaluar la convergencia durante la Fase de muestreo, el algoritmo de optimización compara los valores medios de la función objetivo entre conjuntos sucesivos de configuraciones. En cada iteración, el MCSA genera una nueva combinación de asignación de personal sanitario, la cual se introduce en el simulador. Este calcula los KPI(s), y el valor promedio de estos KPI(s) se utiliza como la función de aptitud F . De esta manera, la comparación de medias entre iteraciones permite determinar si el algoritmo ha alcanzado una convergencia en los resultados de la Exploración o si se requiere generar nuevas muestras.

El proceso de muestreo continúa hasta que la diferencia entre los valores medios de la función objetivo de dos conjuntos consecutivos es inferior a un umbral predefinido. Formalmente, se considera que se ha alcanzado la convergencia cuando se cumple la siguiente condición:

$$|\mu_i - \mu_{i-1}| < \epsilon$$

donde μ_i y μ_{i-1} representan los valores medios de la función objetivo correspondientes al conjunto de muestras actual y al inmediatamente anterior, respectivamente, y ϵ es un umbral pequeño y positivo (por ejemplo, 0,01 o 0,02).

Una vez satisfecho este criterio de parada, se considera que las mejoras potenciales adicionales son insignificantes, por lo que el algoritmo establece el mapa de

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

soluciones de Montecarlo final y procede a la Fase de Explotación, donde se concentra la búsqueda en las regiones más prometedoras identificadas.

6.7.3. Restricciones

El modelo de optimización integra un conjunto de restricciones diseñadas para asegurar que todas las configuraciones evaluadas resulten viables desde un punto de vista operativo en el contexto de un ED. Estas restricciones reflejan tanto las limitaciones prácticas en la disponibilidad de personal como los márgenes de capacidad de los recursos, definidos por políticas institucionales y directrices asistenciales.

1. **Regla de no nulidad:** Cada tipo de personal sanitario (médicos(as), enfermeros(as), técnicos(as), entre otros), en sus categorías *junior* y *senior*, debe respetar un rango válido de asignación, es decir, niveles mínimos y máximos de personal. Formalmente:

$$0 \leq P_{\text{tipo},\text{junior}} \leq \text{Max}_{\text{tipo}} \quad \forall \text{tipo} \in \{\text{categorías profesionales}\} \quad (6.10)$$

$$0 \leq P_{\text{tipo},\text{senior}} \leq \text{Max}_{\text{tipo}} \quad \forall \text{tipo} \in \{\text{categorías profesionales}\} \quad (6.11)$$

Si bien cada categoría puede tomar individualmente el valor cero, no es posible que ambas lo hagan simultáneamente, ya que una configuración $[0, 0]$ dejaría sin cobertura funciones esenciales del ED.

2. **Regla de capacidad máxima:** La suma de profesionales *junior* y *senior* de una misma categoría no puede superar el límite superior predefinido:

$$P_{\text{tipo},\text{junior},i} + P_{\text{tipo},\text{senior},i} \leq \text{Max}_{\text{tipo}} \quad \forall \text{tipo}, \quad \forall i \in \{\text{Configuraciones de Personal}\} \quad (6.12)$$

Donde:

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

- $P_{\text{tipo},\text{junior}}$ representa el número de profesionales de nivel *junior* en una categoría específica (p. ej., enfermero/a junior).
- $P_{\text{tipo},\text{senior}}$ representa el número de profesionales de nivel *senior* de la misma categoría.
- Max_{tipo} indica la capacidad máxima permitida en dicha categoría profesional, independientemente del nivel de experiencia.

Este esquema de restricciones combina flexibilidad y rigor: permite asignar valores nulos en determinadas categorías cuando la redundancia funcional lo compensa, pero garantiza en todo momento la coherencia asistencial y la cobertura mínima.

En términos computacionales, dichas condiciones están incorporadas en el algoritmo MCSA, que descarta automáticamente configuraciones no factibles y concentra la búsqueda en soluciones viables dentro del espacio multidimensional. Esto reduce el riesgo de Exploración de escenarios irrelevantes y mejora la eficiencia del proceso de optimización, preservando la integridad operativa del modelo simulado.

6.7.4. Fase de Explotación

Una vez completada la Fase de Exploración, en la que se han generado y evaluado múltiples configuraciones del personal del ED mediante la simulación, el MCSA transita hacia la segunda etapa de su funcionamiento: la *Fase de Explotación*, con el objetivo de refinar la búsqueda. Esta segunda etapa no introduce nuevos principios algorítmicos, sino que aplica de forma focalizada la lógica previamente descrita para refinar la búsqueda en el contexto particular del problema que nos ocupa: la identificación de configuraciones factibles y eficientes del personal sanitario, que minimicen el *LoS* de los pacientes.

En esta etapa, la información obtenida durante la Exploración es analizada en detalle con el fin de concentrar los esfuerzos de búsqueda en aquellas regiones del espacio de soluciones que han mostrado un rendimiento destacado. La simulación ha permitido obtener, para cada configuración de personal, una serie

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

de KPI(s) que reflejan directamente la capacidad del sistema para responder eficientemente a la demanda asistencial. Este indicador, en particular, permite discriminar entre configuraciones viables y no viables sin necesidad de definir explícitamente una función objetivo formal, ya que los KPI se derivan directamente del comportamiento simulado del sistema.

A fin de delimitar las zonas más prometedoras del espacio de búsqueda, se implementa el procedimiento de agrupamiento basado en densidad para detectar regiones del mapa de soluciones que contienen una alta concentración de configuraciones con buen desempeño. Estas regiones densas (en las que se concentran configuraciones que dan lugar a un mejor rendimiento operacional) se interpretan como zonas de interés operativo, mientras que aquellas más dispersas se descartan por su escasa relevancia.

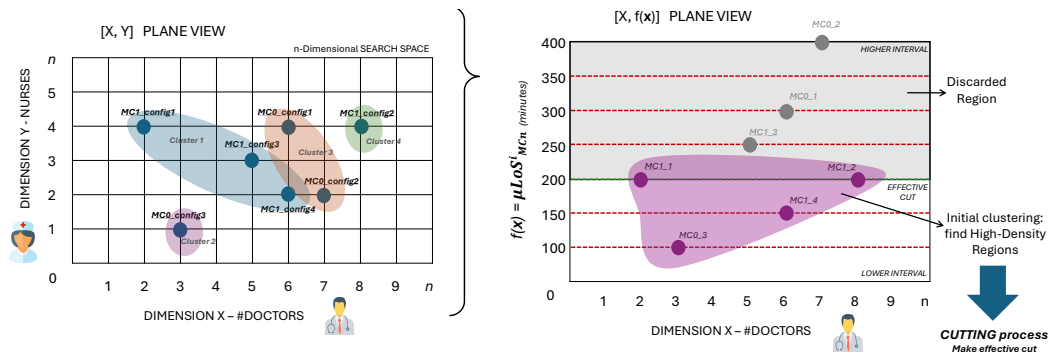


Figura 6.5: Visualización del espacio de búsqueda antes y después de la detección de regiones densas, es decir, áreas con alta concentración de muestras. Para llevar a cabo el proceso de corte efectivo, primero se divide el mapa a lo largo del eje $f(\mathbf{x})$ para localizar regiones de alta densidad.

En la Figura 6.5 se muestra un ejemplo visual de este análisis de densidad aplicado al mapa de Montecarlo generado por la anterior etapa y que proviene de las simulaciones del ED. Las configuraciones agrupadas en clusters forman lo que se denomina *Regiones Factibles*, facilitando así la identificación de soluciones de alta calidad. El proceso de detección de estas regiones densas se puede ver también en el ejemplo simplificado del espacio de búsqueda. En el panel izquierdo se visualiza el mapa completo de configuraciones de personal generadas durante la Fase de Exploración, y en el panel derecho se observa una vista en plano vertical en la que

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

los valores de la función objetivo $f(\mathbf{x}) = \mu(\text{LoS}_{\text{MCn}})$ se representan en relación con sus respectivas configuraciones.

La estrategia de segmentación del espacio de soluciones se realiza a lo largo del eje de valores de la función $f(\mathbf{x})$, es decir, del μ_{LoS} medio. Esta segmentación tiene como objetivo estructurar el análisis posterior, permitiendo identificar las zonas que concentran regiones factibles. La división se lleva a cabo de forma progresiva, comenzando desde el valor mínimo y avanzando hacia el máximo, lo cual es especialmente adecuado para este problema. En cada tramo, se aplica un análisis de densidad para detectar posibles regiones factibles, entendidas como agrupaciones de soluciones con características similares. La segmentación permite acotar progresivamente el área de búsqueda hacia regiones más prometedoras, facilitando así un refinamiento guiado del proceso. De este modo, se identifican de manera sistemática las regiones del espacio de soluciones donde se concentran configuraciones con comportamientos similares, lo que facilita el análisis posterior.

Criterios de corte adaptados al problema del ED

Aunque los criterios de parada utilizados en esta Fase derivan directamente del diseño original del algoritmo MCSA, su aplicación al problema del ED son una extensión específica del criterio de parada general que requiere una interpretación específica orientada a las particularidades del entorno hospitalario:

- **Criterio 1:** Si se alcanza el límite superior del intervalo sin que se detecten nuevas regiones densas de configuraciones viables, el análisis se detiene, por lo que se procede a analizar la totalidad del espacio resultante.
- **Criterio 2:** Si en dos intervalos consecutivos se detecta el mismo número de regiones densas, el algoritmo retrocede dos niveles y realiza el corte efectivo sobre el intervalo anterior en el cual hay un mayor número de agrupaciones. Este mecanismo permite profundizar en zonas con alta concentración de configuraciones potencialmente viables, y es particularmente útil para problemas como el del ED, donde pequeñas variaciones en el número de personal pueden generar efectos no lineales.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Este procedimiento permite identificar no solo configuraciones puntuales de buen desempeño, sino regiones enteras del espacio de búsqueda en las que existen múltiples alternativas operativamente viables. En el contexto del ED, esto resulta fundamental, ya que las decisiones sobre dotación de personal no siempre pueden aplicarse de manera rígida, y disponer de un conjunto robusto de configuraciones equivalentes ofrece mayor flexibilidad para la gestión de turnos, licencias, o contingencias inesperadas.

Por tanto, la Fase de Explotación no solo refina la búsqueda iniciada en la etapa exploratoria, sino que también aporta una herramienta de decisión concreta para los gestores hospitalarios, permitiendo identificar, comparar y validar diversas configuraciones del personal sanitario bajo condiciones de incertidumbre y alta variabilidad.

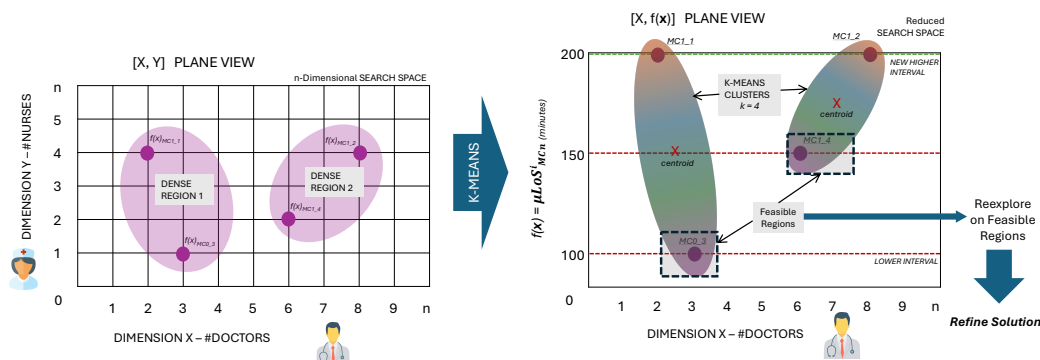


Figura 6.6: Una vez identificadas las regiones densas, se aplica el algoritmo de K-Means para segmentar los datos en cuatro clusters

Una vez identificadas las regiones de alta densidad, las cuales indican configuraciones potencialmente viables con propiedades deseables, se prioriza su Exploración en etapas posteriores. Las zonas menos prometedoras, localizadas por encima del límite superior de corte efectivo, son descartadas, mientras que aquellas situadas por debajo del límite inferior, que presentan una mayor densidad muestral, se conservan como áreas de interés.

Con el fin de refinar aún más la agrupación de soluciones dentro de estas regiones densas, se aplica el algoritmo *K-Means* [48, 104], utilizando la distancia euclidiana como métrica de agrupamiento. Esta estrategia en dos pasos permite, en

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

primer lugar, capturar áreas de alta concentración de soluciones, y posteriormente, optimizar la posición de los centroides y los límites de los clusters resultantes, como se muestra en la Figura 6.6. Tras una evaluación empírica, se estableció el valor de $k = 4$, logrando un equilibrio entre eficiencia computacional y capacidad de diferenciación efectiva.

El algoritmo desplaza iterativamente los centroides hacia zonas con mayor densidad de muestras, subdividiendo así los clusters en subgrupos significativos. Los *k-clusters* resultantes contienen configuraciones que representan posibles soluciones óptimas; la presencia de múltiples agrupamientos relevantes sugiere la existencia de múltiples mínimos globales en ubicaciones distintas del espacio de soluciones.

En este punto, concluye la ejecución completa del algoritmo MCSA. No obstante, dada la naturaleza iterativa de la metodología, el proceso continúa mediante la focalización en las Regiones Factibles identificadas, que pasan a constituir el nuevo espacio de búsqueda. El objetivo es persistir en la localización y Explotación de nuevas soluciones dentro de este espacio reducido, con la finalidad de mejorar progresivamente los resultados obtenidos y evitar quedar atrapados en mínimos locales.

Además, cada nueva solución obtenida es comparada con la anterior, permitiendo así una estrategia de mejora continua que afina progresivamente el espacio de búsqueda y eleva la calidad de las soluciones, hasta que se alcanza un punto en el cual no es posible lograr mejoras adicionales.

6.8. Validación Experimental

Con el objetivo de evaluar la eficacia del algoritmo MCSA en la identificación de configuraciones óptimas del personal sanitario en el ED, se diseñó un marco experimental estructurado. Este proceso de validación incluye ambas fases del algoritmo Exploración y Explotación y se desarrolló sin conocimiento previo del espacio de búsqueda ni supuestos predefinidos sobre la distribución óptima de recursos.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

A diferencia de otros métodos de resolución que podrían basarse en análisis de sensibilidad local o pruebas exhaustivas, esta metodología evita la Exploración completa del espacio combinatorio, que resulta computacionalmente inviable, y se centra, en cambio, en una estrategia iterativa de muestreo y refinamiento progresivo. MCSA inicia su búsqueda generando un conjunto inicial de configuraciones aleatorias, cada una representando una posible asignación de personal y recursos físicos. Estas configuraciones son evaluadas mediante el EDSimulator, que devuelve los KPI(s), y destacamos principalmente el *LoS*, que se utiliza como objetivo principal para guiar la búsqueda.

Aunque sería posible realizar análisis de sensibilidad locales, como estudiar el impacto de pequeñas variaciones en el número de enfermeros(as) o en la capacidad de ciertos recursos, este procedimiento implicaría ejecutar una nueva simulación completa por cada cambio, lo que incrementa significativamente el coste computacional. Por ello, MCSA opta por restringir estratégicamente el número de muestras y enfocar la búsqueda hacia aquellas regiones del espacio que presentan mayor potencial, balanceando la precisión con la eficiencia operativa.

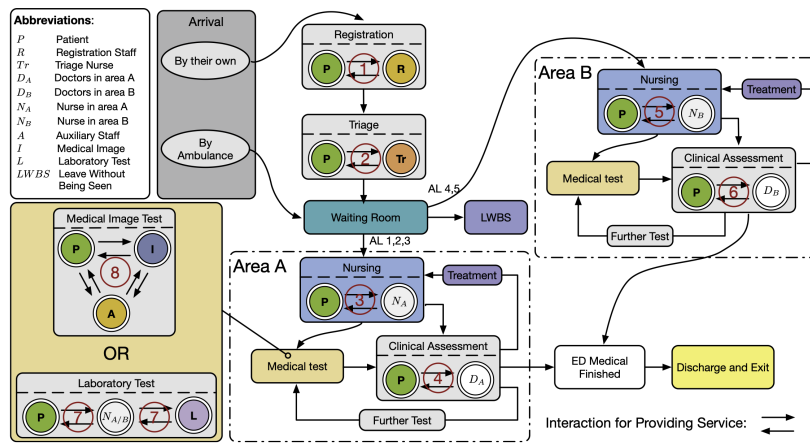


Figura 6.7: Flujo de pacientes a través del ED. El sistema se compone de ocho procesos interdependientes, numerados en los círculos, que regulan el tránsito de los pacientes. Las áreas A y B representan flujos diferenciados para pacientes urgentes y no urgentes, respectivamente, cada una con personal independiente. Modelo de: [61].

La Figura 6.7 representa el flujo de pacientes dentro del ED modelado por el

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

simulador. Esta estructura constituye uno de los insumos configurables del modelo, y puede adaptarse para representar distintos escenarios hipotéticos o condiciones reales. Para esta validación, se asumió un patrón de llegada constante de pacientes, derivado de registros hospitalarios típicos a lo largo del día y de la semana. Esta decisión permite aislar el efecto de la configuración del personal sobre el *LoS*, eliminando la variabilidad asociada a los picos y valles de demanda asistencial. Las simulaciones se ejecutaron sobre un periodo continuo de 90 días, asegurando una representación robusta del comportamiento operacional del sistema.

6.8.1. Diseño Experimental

El experimento se desarrolló dentro de un espacio de búsqueda combinatorio definido por 12 variables, cada una con seis posibles valores enteros en el rango $[0, 5]$, tal como se indica en la Tabla 6.2. Este diseño genera un espacio teórico de $6^{12} = 2,176,782,336$ combinaciones posibles. Sin embargo, la aplicación de restricciones operativas reduce sustancialmente el número de configuraciones viables. En particular, se impuso como condición mínima la presencia de al menos un miembro del personal en cada área funcional del ED, limitando el espacio efectivo a $5^{12} = 244,140,625$ configuraciones.

A pesar de su magnitud, este espacio reducido sigue siendo demasiado amplio para ser evaluado exhaustivamente. Por ello, MCSA emplea estrategias de muestreo inteligente y restricciones adicionales para focalizar la búsqueda en regiones prometedoras. Esta aproximación permite alcanzar soluciones factibles de alta calidad sin necesidad de explorar la totalidad del espacio.

El simulador del ED utilizado en este estudio es una herramienta modular y preconfigurada, capaz de ejecutar simulaciones independientes para múltiples configuraciones de entrada. Esta característica permitió evaluar de manera simultánea distintos escenarios de personal mediante la ejecución paralela de instancias del simulador, haciendo un uso eficiente de recursos computacionales. Para gestionar la carga de procesamiento requerida, se utilizó una infraestructura de alto rendimiento provista por el grupo de investigación, consistente en un cluster

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

de 8 nodos con 512 núcleos AMD Opteron™6262 HE y 2TB de memoria RAM.

Este entorno permitió realizar múltiples ejecuciones de simulación en paralelo, acelerando significativamente el proceso de validación y facilitando el análisis comparativo entre configuraciones bajo condiciones de incertidumbre.

Tabla 6.2: Valores mínimos y máximos del eje para cada tipo de personal (Sr y Jr son dimensiones separadas). La restricción $Sr + Jr \leq 10$ se aplica por categoría.

Staff/Resource	Axis Min	Axis Max	Adjusted Range [min,max]
Sr Admission	0	5	[1, 5]
Jr Admission	0	5	[1, 5]
Sr Triage	0	5	[1, 5]
Jr Triage	0	5	[1, 5]
Sr Doctor (Area A)	0	5	[1, 5]
Jr Doctor (Area A)	0	5	[1, 5]
Sr Doctor (Area B)	0	5	[1, 5]
Jr Doctor (Area B)	0	5	[1, 5]
Sr Nurse (Area A)	0	5	[1, 5]
Jr Nurse (Area A)	0	5	[1, 5]
Sr Nurse (Area B)	0	5	[1, 5]
Jr Nurse (Area B)	0	5	[1, 5]

6.8.2. Resultados Experimentales

En la fase inicial, el algoritmo MCSA generó un conjunto diverso de configuraciones de personal sanitario, que fueron evaluadas mediante el simulador en función del LoS . Este proceso permitió obtener una caracterización amplia del espacio de soluciones, reflejando la variabilidad de combinaciones entre recursos médicos(as) y físicos.

Sobre este mapa de configuraciones simuladas, se aplicó posteriormente un esquema de agrupamiento en dos pasos (*two-phase clustering*). Dicho procedimiento facilitó la identificación de regiones del espacio de búsqueda con mayor densidad de configuraciones viables y con un desempeño especialmente favorable, aportando una base más precisa para el análisis en escenarios de alta dimensionalidad y variabilidad, como los que caracterizan al ED.

La interacción iterativa entre MCSA y el simulador permitió focalizar progresivamente la Exploración sobre las regiones más prometedoras, optimizando

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

también el uso de recursos computacionales y evitando la evaluación de configuraciones poco plausibles o de bajo rendimiento. El proceso culminó con la identificación de una configuración del personal (representada como un vector en el espacio de decisiones) que minimiza el valor esperado del *LoS*, cumpliendo con las restricciones operativas previamente impuestas.

Cabe destacar que la solución propuesta por MCSA constituye una recomendación óptima desde el punto de vista cuantitativo, basada en simulación estocástica y optimización heurística. Sin embargo, corresponde a los responsables de la toma de decisiones validar su factibilidad práctica, considerando aspectos cualitativos, objetivos institucionales, normativa vigente y disponibilidad real de recursos. La flexibilidad de la metodología permite incorporar estos factores adicionales en etapas posteriores, ya sea mediante ajuste de parámetros, redefinición de restricciones o validación cruzada con expertos clínicos y operativos.

Tabla 6.3: Resumen de parámetros clave del problema y métricas obtenidas con el heurístico MCSA. Incluye la definición del espacio de búsqueda, tiempos de simulación, esfuerzo de muestreo y configuración eficiente identificada (asociada al *LoS*).

Parámetro del problema	Valor
$f(x) = \mu\text{LoSMCn}$	Tiempo en minutos
Rango	[0, 5]
# Dimensiones	12
Tamaño del problema (Psize)	6^{12}
Posibles combinaciones	2,176,782,336
Tiempo de simulación EDSim (seg.)	600 aprox.
# muestras heurístico MCSA	6,350
% vs Psize	0.00029
Resultado	Valor
Tiempo de simulación MCSA (min.)	1,557
LoS mejorado encontrado por MCSA (min.)	188
Tiempo de simulación exhaustiva (min.)	48,828,125
Configuración final: Coord.	[4, 4, 4, 2, 2, 5, 4, 3, 3, 3, 5, 5]

Los resultados de esta primera fase experimental se resumen en la Tabla 6.3. La función objetivo en el proceso de optimización se centra en la minimización del

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

LoS. Aunque cada tipo de paciente dentro de una configuración de staff presenta su propio valor de *LoS*, se adopta un único valor representativo que refleja el comportamiento global del sistema, en línea con el enfoque integral planteado previamente en la subsección 6.7.2.

En esta etapa, se han considerado exclusivamente variables correspondientes a recursos humanos, como parte de una primera aproximación acotada del problema. Así, cada configuración generada representa una asignación discreta del personal disponible, evaluada a través del simulador del ED previamente integrado en el marco de trabajo.

Respecto al tiempo medio de simulación por configuración (*SimTime EDSim*), expresado en segundos, proporciona una estimación del coste computacional asociado a cada escenario, el cual varía en función de la complejidad de la asignación evaluada. Para optimizar el uso de recursos computacionales, las simulaciones fueron ejecutadas en paralelo en lotes, lo que permitió evaluar 6350 configuraciones en aproximadamente 25 horas, en comparación con las más de 1050 horas que habrían sido necesarias en un esquema secuencial. Este resultado demuestra la escalabilidad y viabilidad operativa del enfoque adoptado. Finalmente, la configuración identificada por MCSA, detallada en la Tabla 6.4, logró una reducción significativa del *LoS* sin requerir la evaluación exhaustiva del espacio completo de soluciones. Estos resultados reafirman la utilidad del enfoque heurístico propuesto para identificar configuraciones de personal eficientes en entornos hospitalarios complejos y estocásticos.

Cabe destacar que el algoritmo MCSA explora el espacio de soluciones sin conocimiento previo sobre qué configuraciones podrían minimizar el *LoS*. En su lugar, mejora iterativamente este indicador en función de los datos empíricos obtenidos mediante simulación. Cada configuración propuesta es evaluada a través de 15 ejecuciones independientes del simulador, con distintas semillas aleatorias, pero manteniendo la asignación de personal constante según la propuesta sugerida por MCSA. Aunque no se puede garantizar la obtención del óptimo global, evaluaciones anteriores del algoritmo sobre funciones de referencia han demostrado un comportamiento consistente cercano al óptimo, lo cual respalda su aplicación

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Tabla 6.4: Configuración final de personal identificada por MCSA. Cada coordenada indica el número de profesionales asignados por rol y área, correspondientes a las dimensiones exploradas durante la optimización.

Tipo de personal o recurso	Coordenada MCSA
Admisión senior	4
Admisión junior	4
Triage senior	4
Triage junior	2
Médico(a) senior (Área A)	2
Médico(a) junior (Área A)	5
Médico(a) senior (Área B)	4
Médico(a) junior (Área B)	3
Enfermero(a) senior (Área A)	3
Enfermero(a) junior (Área A)	3
Enfermero(a) senior (Área B)	5
Enfermero(a) junior (Área B)	5

en contextos complejos como el presente.

6.9. Validación Extendida: Integración de Variables de Recursos Humanos y Físicos

Tras la primera fase experimental, centrada exclusivamente en la asignación de personal, se propone una extensión del modelo orientada a validar la robustez del enfoque en un escenario más realista. En esta segunda etapa, además de las dotaciones de personal, se incluyen variables asociadas a recursos físicos críticos, lo que incrementa notablemente la complejidad y dimensionalidad del problema.

Con esta ampliación, se plantea que el espacio de búsqueda sea definido ahora por un total de 17 variables, en contraste con las 12 de la fase inicial. Esta integración permite representar con mayor fidelidad las dinámicas operativas de un ED, considerando de manera conjunta tanto la disponibilidad de profesionales como la capacidad material del entorno asistencial (p. ej., número de boxes de

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

atención, puestos de laboratorio).

Las configuraciones exploradas en esta fase corresponden a combinaciones factibles de recursos humanos y materiales, evaluadas mediante el mismo simulador previamente calibrado. El *LoS* continúa siendo el principal indicador de desempeño, sirviendo como métrica central para guiar el proceso iterativo de optimización a través del algoritmo MCSA.

La Tabla 6.5 recoge el listado completo de variables incluidas en esta fase, junto con los rangos de valores permitidos definidos conforme a restricciones operativas realistas. Esta configuración experimental aproxima el modelo al funcionamiento real de un ED, donde las decisiones de planificación no pueden desvincularse de la disponibilidad de recursos físicos indispensables para la atención de pacientes.

Las categorías de personal, diferenciadas en niveles *Senior* y *Junior*, se presentan de forma separada en la Tabla 6.5 con el propósito de reflejar de manera explícita las restricciones estructurales del modelo de optimización. Esta decisión de modelado responde a la necesidad de representar el grado de heterogeneidad en la dotación de recursos humanos, así como las limitaciones prácticas que condicionan la asignación del personal en un ED.

Aunque ambas categorías se tratan como dimensiones independientes dentro del espacio de búsqueda, su asignación conjunta está gobernada por dos reglas operativas fundamentales, diseñadas para garantizar la coherencia funcional del sistema:

1. **Regla de no nulidad:** Cada categoría debe contar con al menos un profesional asignado. En consecuencia, configuraciones del tipo $[0, 0]$ son consideradas inviables, ya que representan escenarios asistenciales sin cobertura mínima en funciones críticas.
2. **Regla de capacidad máxima:** La suma de personal *Senior* y *Junior* dentro de una misma categoría no puede superar el umbral superior predefinido por el modelo. Por ejemplo, una configuración $[5, 4]$ sería inválida si el límite máximo fijado fuera 8 profesionales.

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

La representación explícita de estas restricciones en la tabla no solo facilita la visualización de los márgenes operativos de cada dimensión, sino que también asegura que la búsqueda heurística se lleve a cabo en un espacio factible y clínicamente realista. De este modo, el algoritmo MCSA se centra exclusivamente en configuraciones viables, evitando tanto combinaciones redundantes como escenarios inoperantes.

En términos metodológicos, esta codificación de restricciones dentro del propio diseño experimental constituye una estrategia eficaz para reducir el riesgo de Exploración de soluciones irrelevantes, garantizando al mismo tiempo la integridad del modelo multidimensional. Así, se optimiza el equilibrio entre la expresividad

Tabla 6.5: Definición del espacio de búsqueda multidimensional utilizado en la heurística MCSA. La tabla muestra todas las variables de decisión (personal médico y recursos físicos), junto con sus límites completos del eje y los límites operativos ajustados aplicados durante el proceso de optimización.

Staff or Type of Resource	Axis Min.	Axis Max.	Adjusted Staff + Resource	
			Min	Max
Senior Admission	0	8	3	8
Junior Admission	0	8	3	8
Senior Triage	0	8	3	8
Junior Triage	0	8	3	8
Senior Doctor (Area A)	0	8	3	8
Junior Doctor (Area A)	0	8	3	8
Senior Doctor (Area B)	0	8	3	8
Junior Doctor (Area B)	0	8	3	8
Senior Nurse (Area A)	0	8	3	8
Junior Nurse (Area A)	0	8	3	8
Senior Nurse (Area B)	0	8	3	8
Junior Nurse (Area B)	0	8	3	8
Area B Capacity	0	52	5	52
Carebox Capacity	0	49	4	49
Internal Test Capacity	0	6	3	6
Lab Test Capacity	0	6	3	6
Assistants	0	3	1	3

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

del espacio de búsqueda y la eficiencia computacional requerida para abordar problemas de gran dimensionalidad en entornos críticos como los ED.

6.9.1. Configuración y Ejecución

La presente fase experimental se inicia con la definición de un espacio de búsqueda, en el que cada tipo de recurso (humano o físico) se modela como una variable de decisión independiente (véase Tabla 6.6). Con el objetivo de reflejar las restricciones normativas y las limitaciones operativas específicas del entorno hospitalario, se han acotado parcialmente los valores admisibles para cada variable. Por ejemplo, la capacidad del Área B (*AreaBcap*) se ha restringido al intervalo entre 5 y 52 unidades, estableciendo un umbral mínimo que garantice la seguridad operativa y la continuidad asistencial. Estas restricciones no eliminan la variabilidad del sistema, sino que delimitan una región factible dentro del espacio de búsqueda, alineada con condiciones realistas y viables en un entorno clínico. De este modo, el proceso de optimización se orienta hacia soluciones prácticas y aplicables, evitando configuraciones que resulten operativamente inviables o clínicamente inseguras, sin comprometer la diversidad necesaria para una optimización robusta.

Con el objetivo de replicar condiciones de presión asistencial, se introdujo un incremento del 10% en el flujo de pacientes durante todos los intervalos temporales. Esta modificación busca simular escenarios de alta demanda y evaluar el comportamiento del algoritmo bajo condiciones de estrés operativo. El resto de los parámetros de simulación se mantuvo constante respecto a las etapas experimentales anteriores, con el fin de asegurar la comparabilidad de los resultados.

Una diferencia clave en este escenario reside en el incremento de la dimensionalidad del problema, lo que conlleva una expansión significativa del espacio de búsqueda. Para gestionar la carga computacional asociada, se recurrió a la infraestructura de computación en la nube de Amazon Web Services (AWS) [8], aprovechando sus capacidades de paralelización para la ejecución distribuida de simulaciones. En concreto, se utilizaron instancias *c5.18xlarge*, cada una equipada con 72 vCPUs

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

y 144 GB de RAM.

Esta arquitectura permitió la ejecución simultánea de hasta 50 simulaciones por lote, reduciendo el tiempo total requerido para evaluar 2 586 configuraciones de más de 258 horas (en ejecución secuencial) a aproximadamente 5 horas. Cada lote de 50 simulaciones se completó en un promedio de 6 minutos. Este resultado pone de manifiesto la importancia crítica del cómputo paralelo para viabilizar procesos de optimización basados en simulación, y reafirma la escalabilidad del marco metodológico propuesto.

La incorporación de variables adicionales amplifica considerablemente la complejidad combinatoria del espacio de búsqueda. Cada variable presenta un rango de valores diferente, algunas acotadas en intervalos estrechos (por ejemplo, entre 1 y 8 miembros de personal), otras con márgenes más amplios (por ejemplo, entre 5 y 52 unidades de recursos físicos), lo que da lugar a un paisaje de optimización altamente irregular y de gran dimensionalidad. Esta diversidad introduce complejas interacciones entre variables, que pueden generar efectos contrapuestos sobre el rendimiento global del sistema.

En este contexto, una Exploración puramente aleatoria o uniforme resultaría computacionalmente ineficiente. Para mitigar esta dificultad, la metodología propuesta combina una Exploración guiada con técnicas de clustering, orientando el proceso hacia las regiones del espacio de soluciones con mayor potencial. Mientras la Fase de Exploración asegura una cobertura amplia del espacio, el clustering permite refinar la búsqueda al identificar regiones donde las configuraciones muestran un desempeño prometedor respecto al objetivo de optimización. Esta estrategia dual permite reducir significativamente las evaluaciones innecesarias y acelerar la convergencia del proceso hacia soluciones de alta calidad.

A pesar de la existencia de una configuración base conocida y actualmente utilizada por la dirección del hospital, el algoritmo MCSA inicia el proceso de búsqueda de manera completamente autónoma, sin incorporar ningún conocimiento previo ni suposiciones derivadas de soluciones existentes. El tamaño teórico del espacio

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

de búsqueda, considerando todas las combinaciones posibles dentro de los rangos definidos para las variables de decisión, supera los 249 billones de configuraciones ($2,49 \times 10^{14}$). En este contexto, los enfoques de fuerza bruta resultan computacionalmente inviables, lo que refuerza la necesidad de recurrir a métodos heurísticos.

Para gestionar esta elevada complejidad, se aplicó un ajuste de rangos restrictivo, con el fin de limitar la búsqueda a configuraciones que fueran tanto operacionalmente viables como clínicamente seguras. Esta técnica de reducción *a priori* permite acotar de forma significativa el espacio efectivo de búsqueda y orientar al algoritmo hacia regiones viables del espacio de soluciones. Aunque esta estrategia no garantiza por sí sola la optimización, actúa como un mecanismo de refinamiento que acelera la convergencia hacia soluciones de alta calidad.

La configuración final identificada por MCSA fue contrastada con la configuración

Tabla 6.6: Resumen de los principales parámetros y resultados del proceso de optimización mediante MCSA. La tabla presenta la dimensionalidad del problema, el tamaño efectivo del espacio de búsqueda, y el número total de evaluaciones necesarias para identificar una configuración óptima. El resultado final se expresa en forma de coordenada, representando una solución factible que alcanzó el menor valor observado de *LoS*.

Problem Parameter	Value
$f_i = \mu\text{LoSMCn}$	Time in minutes
Range	[3 to 8(Staff), 5 to 52(BCap), 4 to 49(Carebox), 3 to 6(Int, Lab), 1 to 3(Assistants)]
Effective Range	$6^{12} \times 48 \times 46 \times 4 \times 4 \times 3$
# Dim	17
Theoretical Problem Size (Psize)	$9^9 \times 53 \times 50 \times 7 \times 7 \times 4$
Possible Combinations	249,609,796,080,000
SimTime (in seconds)	358 approximate
# samples Heuristic	2586
% vs Psize	1.12 E-09
Result	Value
Improved LoS found by MCSA (min)	145.32
Final configuration: Coord. form	[1, 3, 2, 3, 3, 2, 4, 3, 0, 7, 5, 3, 51, 27, 1, 3, 2]

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

base del hospital, utilizada como punto de referencia. Los resultados evidencian que la solución propuesta supera de manera consistente a la configuración base en diversos indicadores clave de rendimiento, destacando especialmente la mejora en la media del LoS. Estos hallazgos validan la eficacia de la metodología híbrida basada en simulación y optimización. En un sentido más amplio, este enfoque se presenta como una herramienta práctica y orientada a datos para apoyar la toma de decisiones estratégicas en la planificación de recursos a medio y largo plazo, especialmente en escenarios de variabilidad estacional, aumentos previstos de demanda o ausencias temporales de personal por bajas o licencias.

6.9.2. Discusión de los Resultados

La comparación entre la configuración de personal optimizada mediante la heurística MCSA y la configuración base actualmente implementada por la dirección hospitalaria se presenta en la Tabla 6.7, mientras que los KPI(s) relativos a cada configuración de personal se detalla en la Tabla 6.8.

Los resultados evidencian una mejora sustancial en el desempeño del sistema, destacando una reducción del 23,94% en el *LoS*, disminuyendo de 191,05 a 145,32 minutos. Este indicador resume el impacto acumulado de las decisiones operativas, por lo que su mejora refuerza la primera hipótesis de investigación: ajustes dirigidos en la asignación de personal pueden traducirse en ganancias operativas significativas.

El análisis detallado de los KPI(s) revela además cambios estructurales relevantes. En términos de longitud de colas, se registran disminuciones notables en Admisión (de 19 a 6) y Triage (de 33 a 9), lo cual sugiere una mayor capacidad del sistema para absorber la demanda inicial y reducir cuellos de botella en las etapas críticas de entrada. En cuanto a la ocupación máxima, los valores se mantienen en niveles altos (cerca del 100%) en ambas configuraciones, lo cual es esperable dada la naturaleza del entorno. Sin embargo, sí se observan variaciones importantes: el MCSA reduce significativamente la saturación en Enfermería A (del 100% al 86,26%), a la vez que incrementa la utilización de Enfermería B y del personal

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Tabla 6.7: Comparación entre la Asignación de Personal y Recursos de la Configuración base del personal sanitario y la Configuración encontrada por el MCSA.

Tipo de Recurso	Config. Base	Config. por MCSA
Admisión Senior	0	1
Admisión Junior	3	3
Triaje Senior	1	2
Triaje Junior	2	3
Médico(a) Senior (A)	2	3
Médico(a) Junior (A)	2	2
Médico(a) Senior (B)	2	4
Médico(a) Junior (B)	3	3
Enfermero(a) Senior (A)	2	0
Enfermero(a) Junior (A)	2	7
Enfermero(a) Senior (B)	5	5
Enfermero(a) Junior (B)	7	3
Capacidad Área B	52	51
Boxes Atención	49	27
Pruebas Internas	6	1
Laboratorio	6	3
Personal Auxiliar	3	2

auxiliar. Esto indica una redistribución más equilibrada de la carga de trabajo, evitando sobrecargas puntuales y utilizando mejor los recursos disponibles.

Finalmente, en el tiempo de estancia por tipo de paciente, el método propuesto logra reducir el *LoS* en los tipos 1, 2 y 3 de forma consistente (hasta un 48 % en el caso de los pacientes tipo 1), lo que demuestra que el sistema mejora especialmente en los perfiles de mayor complejidad. En contraste, los tipos 4 y 5 experimentan incrementos moderados, lo cual sugiere que la priorización de casos más severos conduce a tiempos de procesamiento ligeramente mayores en los pacientes de baja urgencia, un comportamiento que es coherente con la lógica de atención clínica.

Dichas mejoras se lograron explorando un espacio de soluciones altamente dimensional y sujeto a restricciones realistas, lo que respalda la segunda hipótesis. A pesar de que el espacio teórico de búsqueda supera los $2,49 \times 10^{14}$ escenarios

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Tabla 6.8: Comparación entre los KPI(s) asociados a la configuración base del hospital y la solución propuesta por el método.

KPI(s)	Configuración Base	Configuración MCSA
Longitud de Colas		
Admisión	19	6
Triaje	33	9
Boxes	0	0
Espera Área B	0	19
Ocupación Máxima (%)		
Admisión	100.00	98.95
Triaje	100.00	100.00
médicos(as) A	100.00	100.00
médicos(as) B	100.00	100.00
Enfermería A	100.00	87.26
Enfermería B	82.36	97.70
Auxiliar	78.88	100.00
Laboratorio	100.00	100.00
Imagenología	100.00	100.00
<i>Length of Stay (LoS en min.)</i>		
Pacientes Tipo 1	211.02	112.67
Pacientes Tipo 2	238.72	133.49
Pacientes Tipo 3	232.29	129.96
Pacientes Tipo 4	143.83	183.28
Pacientes Tipo 5	129.42	167.19
Promedio LoS	191.06	145.32

posibles, el algoritmo necesitó únicamente 2,586 simulaciones, menos del $1,12 \times 10^{-9}$ del total, para encontrar una solución de alta calidad. Esta capacidad de convergencia con un número tan reducido de evaluaciones confirma la tercera hipótesis: MCSA es significativamente más eficiente que los enfoques exhaustivos en contextos combinatorios de gran escala.

Más allá del indicador principal (*LoS*), los análisis complementarios muestran que la metodología mantiene o incluso mejora las tasas de ocupación en áreas críticas

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

como admisión y triaje. Esto indica que el algoritmo no solo contribuye a reducir colas y tiempos de espera, sino que también redistribuye la carga de trabajo de forma equilibrada sin requerir un incremento global del personal. La configuración final propuesta favorece un flujo más estable a lo largo del sistema, preservando niveles de utilización compatibles con una operación sostenible.

Los resultados también subrayan el impacto de introducir variaciones estratégicas en la composición del equipo. El refuerzo de personal médico en las etapas iniciales (particularmente en admisión y triaje), junto con una redistribución más eficiente del personal de enfermería, se identifican como elementos clave para la mitigación de cuellos de botella. Aunque el objetivo principal del estudio fue la reducción del *LoS* promedio, la interacción entre los distintos KPI(s) sugiere la conveniencia de adoptar, en trabajos futuros, un enfoque explícitamente multiobjetivo. Por ejemplo, desagregar el *LoS* por nivel de severidad (tipos 1 a 5) permitiría una evaluación más granular del desempeño asistencial y abriría oportunidades para estrategias de optimización diferenciadas.

En conjunto, los resultados confirman la viabilidad y robustez del enfoque heurístico guiado por simulación. Frente a métodos tradicionales, habitualmente más rígidos o costosos en términos computacionales, MCSA se presenta como una alternativa flexible, escalable y basada en datos, capaz de generar configuraciones aplicables con un coste evaluativo muy reducido. Esto lo posiciona como una herramienta valiosa para la planificación y gestión de recursos en entornos hospitalarios dinámicos y de alta complejidad.

6.10. Conclusiones del Capítulo

Este capítulo ha puesto de relieve el potencial del enfoque combinado de optimización heurística y simulación basada en agentes (ABM) como herramienta de apoyo a la toma de decisiones en la asignación de personal en los Servicios de Urgencias (EDs). La aplicación del algoritmo MCSA permitió identificar configuraciones de recursos capaces de mejorar de forma significativa el rendimiento operativo del sistema, medido principalmente mediante el tiempo promedio de

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

estancia (*Length of Stay*, LoS).

Los resultados respaldan la hipótesis central: ajustes específicos en la distribución del personal, guiados por técnicas de simulación y optimización, pueden generar mejoras consistentes en la eficiencia global. La reducción del 23,94 % en el LoS respecto al escenario base, obtenida tras evaluar únicamente una fracción ínfima del espacio teórico de búsqueda (menos del $1,12 \times 10^{-9}$), demuestra la capacidad del método para explorar dominios combinatorios de gran escala de forma eficiente y bajo restricciones realistas.

Es importante destacar que la metodología propuesta no se limita a encontrar una única configuración adecuada, sino que proporciona un marco flexible y orientado por datos para analizar cómo distintas asignaciones influyen en el comportamiento del sistema. Esta característica es especialmente relevante en entornos sujetos a variaciones constantes, como fluctuaciones estacionales de demanda o cambios no previstos en la disponibilidad del personal.

La experimentación se desarrolló en dos etapas complementarias. La primera se centró exclusivamente en la asignación de recursos humanos; la segunda incorporó recursos físicos críticos, aumentando notablemente la complejidad combinatoria. Aun así, el rendimiento del algoritmo se mantuvo gracias al uso de muestreo estocástico, técnicas de segmentación y agrupamiento, y el apoyo de infraestructuras de computación de alto rendimiento tanto locales como en la nube, lo que permitió validar la escalabilidad del enfoque.

Desde una perspectiva metodológica, uno de los principales aportes radica en la integración eficaz entre la heurística MCSA y un simulador calibrado con datos reales del Hospital Parc Taulí (Sabadell). Esta integración permitió evaluar múltiples KPI(s) a partir de cada configuración, al tiempo que redujo el coste computacional mediante evaluaciones progresivas y agrupadas. La incorporación de restricciones operativas garantizó, además, que todas las soluciones exploradas fueran factibles y clínicamente seguras.

En el plano práctico, la solución obtenida no solo redujo el LoS promedio, sino que mantuvo las tasas de ocupación del personal en niveles compatibles con la

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

sostenibilidad operativa. La redistribución estratégica de profesionales senior en etapas críticas del flujo asistencial (especialmente en admisión y triaje) se reveló clave para mitigar cuellos de botella sin necesidad de incrementar la plantilla total.

Como línea de trabajo futuro, se propone evolucionar hacia un enfoque explícitamente multiobjetivo que permita equilibrar simultáneamente distintos indicadores, como la reducción del LoS por nivel de severidad, la distribución equitativa de la carga laboral o la mejora en la atención diferenciada. Este avance otorgaría al modelo una mayor capacidad para adaptarse a la complejidad inherente de los entornos sanitarios actuales.

En conjunto, los resultados obtenidos confirman la viabilidad y utilidad del enfoque MCSA como herramienta de planificación de recursos en Servicios de Urgencias. Frente a métodos tradicionales más rígidos o costosos, la metodología presentada destaca por su flexibilidad, escalabilidad y fundamentación empírica, constituyendo una contribución relevante para la gestión de sistemas de salud en escenarios caracterizados por alta variabilidad y demanda operativa.

Glosario del Capítulo

A continuación se presenta el glosario del Capítulo 6.9, que incluye los términos y símbolos más relevantes utilizados en esta sección para facilitar la comprensión de la terminología empleada y servir de referencia rápida.

Tabla 6.9: Glosario del Capítulo

Término	Descripción
ED	<i>Emergency Department</i> (Servicio de Urgencias Hospitalarias)
MTS	<i>Manchester Triage System</i> (Sistema de Triage de Manchester)
LoS	<i>Length of Stay</i> (Tiempo medio de estancia del paciente)
KPI	<i>Key Performance Indicator</i> (Indicador clave de desempeño)
DES	<i>Discrete-Event Simulation</i> (Simulación de Eventos Discretos)
ABM	<i>Agent-Based Modeling</i> (Simulación Basada en Agentes)
SD	<i>System Dynamics</i> (Dinámica de Sistemas)
Area A	Zona destinada a atención de pacientes críticos (triaje 1–3)
Area B	Zona destinada a atención de pacientes menos graves (triaje 4–5)
Variables y Conjuntos de Decisión	
$\mathbf{x} = (x_1, \dots, x_n)$	Vector de decisión que define la configuración de recursos del ED.

Continúa en la siguiente página

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Glosario del Capítulo (continuación)

Término	Descripción
x_j	Variable discreta asociada al recurso j (médicos(as), enfermería, técnicos(as), etc.).
$x_j^{(i)}$	Valor de la variable x_j en la configuración candidata i .
Ω	Región factible definida por disponibilidad de personal y restricciones operativas.
$x_j^{\min}, x_j^{\max}$	Límite inferior y superior permitidos para el recurso j .
Parámetros de Personal	
$P_{\text{tipo}, \text{junior}}$	Número de profesionales de nivel <i>junior</i> en la categoría “tipo”
$P_{\text{tipo}, \text{senior}}$	Número de profesionales de nivel <i>senior</i> en la categoría “tipo”
Max_{tipo}	Número máximo permitido de profesionales en la categoría “tipo”
Parámetros de Simulación y Evaluación	
N	Número de réplicas Montecarlo por configuración.
$\text{LoS}_{\text{MCN}}(\mathbf{x})$	Conjunto de valores LoS obtenidos en N simulaciones bajo $\mathbf{x}$.
$\text{LoS}_{\text{MC15}}(\mathbf{x}^{(i)})$	Conjunto de 15 valores representativos del LoS para la configuración i .
$\text{LoS}_{\text{MCN}, k}(\mathbf{x})$	Valor de LoS para el nivel de triaje k bajo la configuración $\mathbf{x}$.
$\mu_{\text{LoS}, k}(\mathbf{x})$	Media de LoS del nivel de triaje k bajo $\mathbf{x}$.
$\mu_{\text{LoS}, \text{KPI}}^{(i)}$	Media global ponderada del LoS para la configuración candidata i (KPI global).

Continúa en la siguiente página

6. APLICACIÓN DEL MODELO EN LA GESTIÓN DEL PERSONAL EN UN SERVICIO DE URGENCIAS HOSPITALARIAS UTILIZANDO LA SIMULACIÓN

Glosario del Capítulo (continuación)

Término	Descripción
$f(\mathbf{x})$	Función objetivo basada en la métrica $\mu_{\text{LoS}}(\mathbf{x})$.
$\mathbf{x}^*$	Configuración óptima que minimiza la función objetivo.
ϵ	Umbral de convergencia del criterio de parada del algoritmo MCSA.
$y = g(\text{KPI})$	Función de evaluación que resume el desempeño global de la configuración.

Capítulo 7

Conclusiones, principales aportaciones y líneas abiertas

La tesis ha desarrollado y validado una metodología de búsqueda estocástica avanzada para problemas de optimización de alta dimensionalidad, basada en la combinación de muestreo aleatorio tipo Montecarlo y técnicas de clusterización en dos pasos, aplicada tanto a entornos sanitarios como a problemas clásicos de referencia (benchmarks), o el problema de la mochila binaria.

7.1. Hallazgos y Conclusiones

Los principales hallazgos y conclusiones son los siguientes:

1. **Exploración eficiente de espacios grandes:** La heurística MCSA permite recorrer espacios de búsqueda extremadamente grandes haciendo un uso eficiente de recursos computacionales, evitando la búsqueda exhaustiva y reduciendo significativamente el coste de cómputo. Esta capacidad se validó tanto en problemas clásicos de optimización combinatoria, como el problema de la mochila binaria, como en entornos simulados reales, como el Servicio de Urgencias (ED).

7. CONCLUSIONES, PRINCIPALES APORTACIONES Y LÍNEAS ABIERTAS

2. **Identificación de regiones relevantes:** El algoritmo incorpora un proceso de selección y acotamiento de regiones factibles mediante clusterización en dos pasos, concentrando la búsqueda en zonas del espacio donde las soluciones cumplen las restricciones y presentan altos valores de la función objetivo o indicadores operativos favorables. Esto mejora la calidad de las soluciones y evita evaluaciones innecesarias.
3. **Validación con escenarios reales y benchmarks:** La metodología se validó usando funciones de referencia y entornos reales. En el ED, se logró reducir significativamente el *Length of Stay* (LoS) de los pacientes; en el problema de la mochila, MCSA halló soluciones óptimas o cercanas al óptimo con un número reducido de evaluaciones, demostrando robustez y generalidad.
4. **Flexibilidad y adaptabilidad:** La metodología no solo produce soluciones de alta calidad, sino que ofrece un marco flexible capaz de integrarse con simuladores realistas o modelos específicos de distintos dominios, adaptándose a restricciones complejas, múltiples variables discretas y requisitos operativos estrictos.
5. **Escalabilidad computacional:** La combinación de muestreo estocástico, clusterización y refinamiento iterativo demuestra ser escalable y computacionalmente eficiente, incluso en espacios de búsqueda con más de 10^{14} configuraciones posibles, asegurando convergencia hacia soluciones de alto desempeño con un número reducido de evaluaciones.

Se ha demostrado ampliamente que la metodología propuesta no se limita a producir una solución única, sino que constituye una herramienta de apoyo flexible y adaptable, capaz de ofrecer, por ejemplo, al gestor sanitario un abanico de configuraciones viables y validadas por simulación. Esta característica resulta particularmente valiosa en escenarios dinámicos, como aquellos en los que hay variaciones en la demanda, picos estacionales, ausentismo del personal o cambios estructurales en la organización del servicio.

Desde un punto de vista metodológico, la integración coherente de MCSA con un

7. CONCLUSIONES, PRINCIPALES APORTACIONES Y LÍNEAS ABIERTAS

simulador calibrado con datos reales del Hospital Parc Taulí en Sabadell, como se presentó en el último caso de estudio, representa un avance relevante respecto a los enfoques tradicionales. La combinación de mecanismos de muestreo Montecarlo con técnicas de refinamiento basadas en clustering en dos pasos, ha permitido reducir el coste computacional y mejorar la eficiencia exploratoria, validando la escalabilidad del método tanto en infraestructuras locales de alto rendimiento como en entornos de computación en la nube.

En conjunto, los resultados obtenidos en esta tesis confirman la viabilidad, robustez y aplicabilidad del enfoque desarrollado, situándolo como una contribución novedosa y práctica al campo de la optimización de recursos en sistemas sanitarios complejos. La metodología no solo proporciona mejoras cuantitativas en métricas clave, sino que también abre la puerta a nuevas formas de integrar herramientas de simulación y optimización dentro de los procesos de planificación y gestión hospitalaria.

Por último, la tesis valida que la heurística MCSA constituye una técnica novedosa y eficaz para la optimización de sistemas complejos. El marco desarrollado combina Exploración amplia y enfoque estratégico en regiones factibles, integrando teoría de Montecarlo, técnicas de clustering y simulación de sistemas realistas, ofreciendo una herramienta sólida tanto para la investigación académica como para la toma de decisiones aplicada en entornos operativos complejos.

7.2. Líneas Futuras

El marco metodológico desarrollado en esta tesis, basado en el algoritmo *MCSA* con muestreo estocástico y doble clusterización, ha demostrado su eficacia tanto en problemas de referencia como en la asignación de personal en servicios de urgencias (ED). No obstante, existen varias líneas de trabajo futuras que podrían ampliar su aplicabilidad, mejorar la toma de decisiones y aumentar la robustez del enfoque.

1. **Desarrollo de un índice combinado multiobjetivo:** Actualmente, la optimización se ha centrado principalmente en el tiempo promedio de

7. CONCLUSIONES, PRINCIPALES APORTACIONES Y LÍNEAS ABIERTAS

estancia (*LoS*). Una extensión natural consiste en construir un índice agregado que combine múltiples KPIs, incluyendo *LoS* desagregado por tipo de paciente, tasas de ocupación del personal y utilización de recursos físicos. Este índice podría ponderar cada componente según su importancia estratégica, permitiendo una evaluación integral del rendimiento y facilitando decisiones más equilibradas.

2. **Optimización multiobjetivo y *Pareto front*:** Incorporar formalmente técnicas de optimización multiobjetivo permitiría identificar configuraciones que representen un compromiso eficiente entre múltiples indicadores. Por ejemplo, se podría buscar simultáneamente la minimización del *LoS*, la carga laboral del personal y la equidad en la atención por tipo de paciente. La normalización de indicadores y la generación de fronteras de Pareto de tres o más objetivos ofrecerían una visión completa de las soluciones factibles y de sus trade-offs.
3. **Integración con gemelos digitales en tiempo real:** Una línea futura prometedora consiste en integrar la metodología MCSA con un Gemelo Digital del servicio de urgencias, alimentado por datos en tiempo real. Esto permitiría realizar simulaciones continuas y ajustar dinámicamente la asignación de personal y recursos según la demanda efectiva, anticipando cuellos de botella y optimizando la operación de manera predictiva.
4. **Refinamiento de la clusterización y adaptatividad del muestreo:** La estrategia de doble clusterización puede extenderse para incluir métodos dinámicos de selección de regiones factibles, ajustando progresivamente la granularidad del muestreo según la densidad de soluciones prometedoras. Esto mejoraría la eficiencia exploratoria, especialmente en espacios de búsqueda extremadamente grandes o con alta incertidumbre.
5. **Validación en escenarios ampliados y heterogéneos:** Aunque la metodología se ha validado con éxito en el ED y en benchmarks clásicos, sería recomendable evaluar su desempeño en escenarios más heterogéneos, incluyendo otros servicios hospitalarios, variaciones estacionales o

7. CONCLUSIONES, PRINCIPALES APORTACIONES Y LÍNEAS ABIERTAS

simulaciones de crisis sanitarias. Esto permitiría generalizar la aplicabilidad del enfoque y ajustar sus parámetros a contextos distintos.

En conjunto, estas líneas futuras apuntan a transformar el enfoque MCSA en una herramienta de optimización estratégica aún más robusta y flexible, capaz de combinar eficiencia computacional, soporte a decisiones multiobjetivo y adaptabilidad a entornos dinámicos, acercándose a un sistema de gestión proactivo basado en simulación y datos en tiempo real.

7. CONCLUSIONES, PRINCIPALES APORTACIONES Y LÍNEAS ABIERTAS

Apéndice

Apéndice A

Resultados adicionales

Aquí se presentan tablas y figuras complementarias que no se incluyen en el cuerpo principal de la tesis.

A.1. Pruebas adicionales sobre funciones *benchmark*

*viene de la Sección 3.7.3 del Capítulo 3

A continuación se presentan pruebas adicionales utilizando la función **Eggholder**, ampliamente referenciada [71] en la literatura de optimización global debido a su topología altamente compleja, con numerosos óptimos locales, regiones irregulares y dispersas en el espacio de búsqueda. Esta función ilustra un caso extremo para técnicas de clustering como K-Means, donde el agrupamiento por coordenadas resulta insuficiente debido a la distribución de las regiones de interés.

En este análisis se aplica un procedimiento preliminar similar a los filtros descritos en el capítulo de metodología, cuya función es reducir el espacio de búsqueda, concentrando la atención en las regiones más prometedoras. Aunque esta prueba se realiza como un estudio adicional y no forma parte de la metodología final

A. RESULTADOS ADICIONALES

del MCSA, permite visualizar cómo la selección de centroides puede extraer información relevante incluso en escenarios con múltiples óptimos locales y estructuras irregulares, anticipando los conceptos de reducción y clustering que se detallan posteriormente en el Capítulo de Metodología.

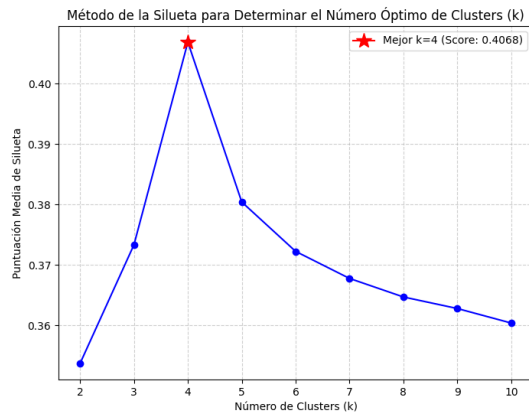
Además, se exploró en un contexto de estado del arte el uso de técnicas clásicas para determinar el número óptimo de clusters, tales como el método del codo (*Elbow Method*) [40] y el método de la silueta [81]. El primero consiste en graficar la variación explicada (o la suma de errores *intra-cluster*) frente al número de clusters y seleccionar el codo de la curva como punto de equilibrio. Por su parte, el coeficiente de silueta, propuesto por Rousseeuw, proporciona una medida de qué tan bien cada punto está asignado a su propio cluster en comparación con otros, con valores entre -1 y +1. En el caso de Eggholder, esta metodología de análisis permitió estudiar la coherencia y separación de los clusters detectados, aportando una visión crítica sobre cuántos centroides son realmente significativos en presencia de una topología tan compleja.

Las pruebas realizadas demostraron que las limitaciones observadas no derivan de los algoritmos de agrupamiento en sí, sino del tipo de datos generados. La adaptación metodológica propuesta se orienta, por tanto, a reforzar esta etapa intermedia entre la generación y el análisis de las muestras, asegurando que el modelo conserve coherencia interpretativa en dominios abstractos y de alta dimensionalidad.

En la Figura A.1a y en la tabla A.1b, el método de la silueta sugiere que el número óptimo de clusters para las muestras Montecarlo de la función Eggholder es $k = 4$. Este resultado coincide con la distribución visual de los datos, donde se distinguen cuatro regiones de alta densidad. Para valores de $k > 4$, los índices de silueta disminuyen notablemente, evidenciando una fragmentación artificial del espacio, en la que clusters naturalmente continuos se dividen sin justificación estructural.

Sin embargo, como parte del análisis exploratorio, se probaron subdivisiones mayores, por ejemplo $k = 7$ y $k = 40$, con el fin de evaluar la consistencia de la segmentación y la capacidad de K-Means para capturar estructuras más finas.

A. RESULTADOS ADICIONALES



k	Puntuación Silueta
2	0.3537
3	0.3733
4	0.4068
5	0.3804
6	0.3722
7	0.3677
8	0.3647
9	0.3628
10	0.3604

(a) Gráfica de puntuaciones del método de la silueta para la función Eggholder.

(b) Tabla de puntuaciones medias de silueta para cada k .

Figura A.1: Determinación del número óptimo de clusters (k) para las muestras Montecarlo de la función Eggholder mediante el método de la silueta. El valor máximo de puntuación se obtiene para $k = 4$, lo que confirma que cuatro clusters son los más representativos de la estructura de los datos.

Aunque la silueta indicaba que $k = 4$ era el valor más representativo, estas pruebas adicionales permitieron observar cómo el algoritmo se comporta ante situaciones de *overclustering* y cómo se generan centroides en regiones de menor densidad.

Estos resultados confirman las limitaciones de K-Means frente a distribuciones irregulares y heterogéneas, y al mismo tiempo demuestran la utilidad del método de la silueta como guía para seleccionar un número de clusters representativo. La elección final de $k = 4$ se fundamenta así en un análisis empírico combinado, visual, cuantitativo y comparativo, que garantiza una segmentación consistente con la naturaleza del muestreo Montecarlo, mientras que las subdivisiones adicionales muestran el comportamiento del algoritmo en escenarios más complejos y permiten un estudio de estado del arte.

Como se ve en la Figura A.2, la aplicación de K-Means sobre los datos generados mediante muestreo Montecarlo de la función Eggholder evidencia limitaciones significativas derivadas de la naturaleza del espacio de datos. A pesar de variar

A. RESULTADOS ADICIONALES

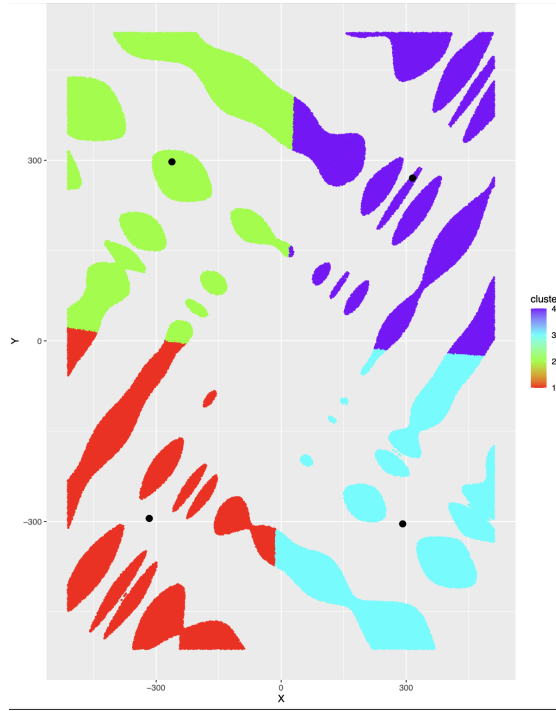


Figura A.2: Visualización de la clusterización de la función *benchmark* Eggholder usando K-Means con $k = 4$. Se observa que los centroides detectados no reflejan correctamente la estructura de los datos provenientes del muestreo Montecarlo, dividiendo un mismo conglomerado en múltiples clusters aparentemente arbitrarios y agrupando puntos alejados en la misma clase.

el número de clusters (k), probando valores desde 4 hasta 40, los centroides identificados no logran reflejar la distribución real de las muestras: en algunos casos, dividen un único conjunto de soluciones cercanas en múltiples grupos, y en otros, agrupan puntos distantes en una misma clase.

Este comportamiento no indica un fallo del algoritmo en sí, sino una incompatibilidad entre sus supuestos y las características de los datos generados por Montecarlo, donde las regiones de alta densidad pueden adoptar formas irregulares y presentar fronteras difusas. Los valores obtenidos mediante el método de la silueta, generalmente bajos o inconsistentes, confirman esta falta de ajuste y justifican la necesidad de emplear técnicas más flexibles, capaces de adaptarse a geometrías complejas y distribuciones no uniformes, que reflejen con mayor fidelidad la naturaleza estocástica del espacio de soluciones generado

A. RESULTADOS ADICIONALES

por Montecarlo y permitan identificar adecuadamente las estructuras subyacentes, incluso en regiones no convexas con alta concentración de muestras.

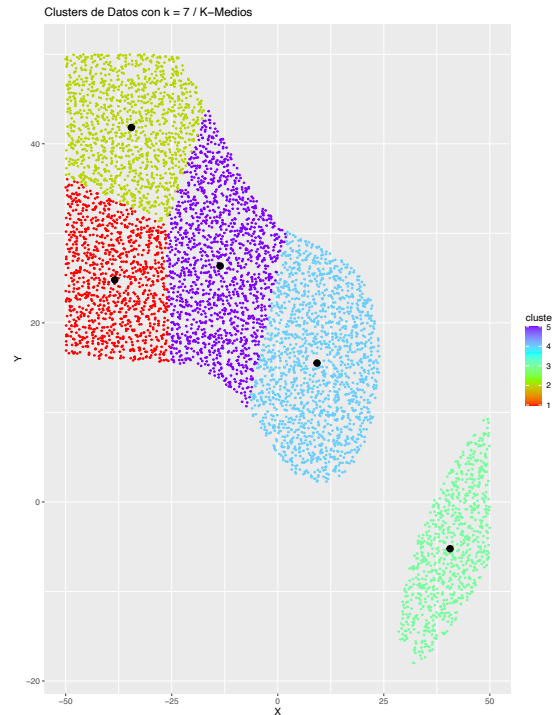


Figura A.3: Resultados de K-Means con ($k=7$) a un conjunto de puntos de muestreo en un espacio de decisión bidimensional (X,Y), típico de las regiones óptimas de la función Eggholder.

Esta visualización de la Figura A.3 demuestra una limitación inherente de la aplicación del algoritmo K-Means en conjuntos de datos cuya estructura natural no es esférica. Los puntos de muestreo se distribuyen en varias concentraciones densas, separadas por regiones de baja densidad, y estas concentraciones a menudo poseen formas irregulares y cóncavas (como la región cian en el centro-derecha). Al imponer la partición en $K = 7$ clusters, K-Means se ve obligado a dibujar fronteras de decisión lineales (hiperplanos) que minimizan la varianza cuadrática dentro de cada grupo. Esta restricción geométrica provoca que K-Means corte y separe forzosamente las distribuciones densas naturales. Por ejemplo, la gran concentración a la izquierda se segmenta en tres clusters que se extienden más allá de sus núcleos de densidad, y el cluster hacia la derecha con forma cóncava

A. RESULTADOS ADICIONALES

es partido artificialmente para ajustarse a un centroide central. Por lo tanto, el resultado no refleja la conectividad por densidad del espacio de muestreo, sino una partición geométrica forzada del volumen total del dataset.

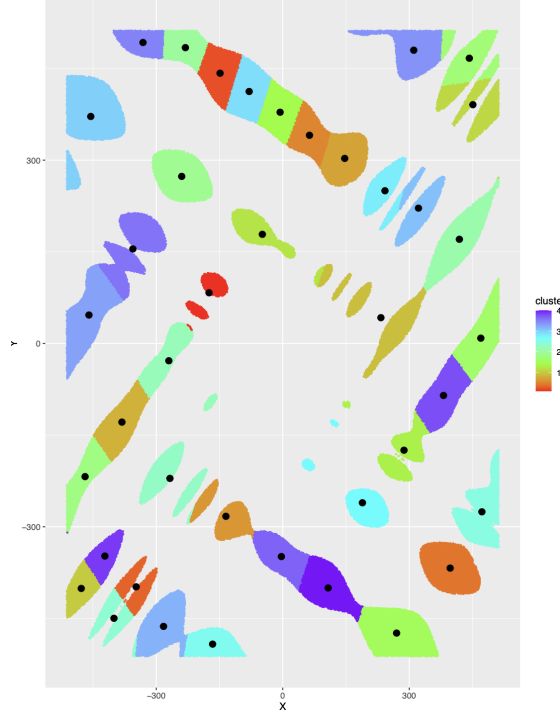


Figura A.4: Visualización de la clusterización de la función *benchmark* Eggholder usando K-Means con $k = 40$. Se observa la aparición de múltiples centroides que no representan adecuadamente la estructura del mapa de Montecarlo, generando clusters fragmentados y dispersos. Esta configuración es además costosa desde el punto de vista computacional y no ofrece mejoras significativas en la interpretación de la estructura de los datos.

En la Figura A.4 se aprecia que, incluso al aumentar el número de clusters a $k = 40$, K-Means no consigue capturar la estructura real del espacio de soluciones. Esto se debe a que el paisaje del problema Eggholder presenta regiones altamente no convexas, discontinuas y con múltiples mínimos locales profundos. Este tipo de geometría genera patrones irregulares y zonas de alta concentración de puntos que no pueden ser representadas adecuadamente mediante las particiones lineales que impone K-Means.

El incremento de k produce una proliferación de centroides que fragmentan el

espacio en grupos pequeños y dispersos, sin corresponder con las regiones densas y continuas observadas en un mapa de Montecarlo. Como consecuencia, los centroides se distribuyen en formaciones sin coherencia topológica, dando lugar a *islas* artificiales que no reflejan la estructura subyacente del espacio de soluciones.

Además de incrementar el coste computacional, esta configuración no aporta una mejora significativa en la interpretación de la distribución de soluciones. Estas observaciones refuerzan la idea de que los métodos basados en particiones requieren un espacio previamente estructurado para operar de manera efectiva. En consecuencia, se propuso el desarrollo de un mecanismo previo de análisis y reducción del dominio, tal como se describe en el Capítulo 4.

A.2. Resultados detallados: Boxplots de las funciones del CEC2017

*viene de la Sección 4.5.1 del Capítulo 4

Esta sección complementa los resultados presentados en el Capítulo 4, mostrando la dispersión y robustez de los resultados obtenidos por el MCSA mediante boxplots. Se realizaron 200 ejecuciones promedio del algoritmo para cada función de benchmark, incluyendo funciones con múltiples dimensiones, rotadas o desplazadas, tal como aparecen en la evaluación del Capítulo 4 (Tabla 4.2).

Como se aprecia en las Figuras A.5 y A.6, el MCSA logra resultados relevantes de manera consistente. En algunas funciones se alcanza el óptimo exacto, mientras que en otras se obtienen soluciones cercanas, lo que indica que los valores de los resultados son razonablemente simétricos. Cabe destacar que a medida que aumenta la dimensión y el tamaño del espacio de búsqueda, la posibilidad de obtener muestras aleatorias distribuidas homogéneamente disminuye, lo que puede llevar a una convergencia prematura de la función objetivo y, por tanto, a ineficiencias. Estos resultados sugieren oportunidades de mejora en la reducción del espacio de búsqueda y en la Exploración de soluciones, especialmente para problemas de gran escala, reforzando la necesidad de un análisis profundo de la

A. RESULTADOS ADICIONALES

estructura del espacio de soluciones para optimizar la robustez del algoritmo.

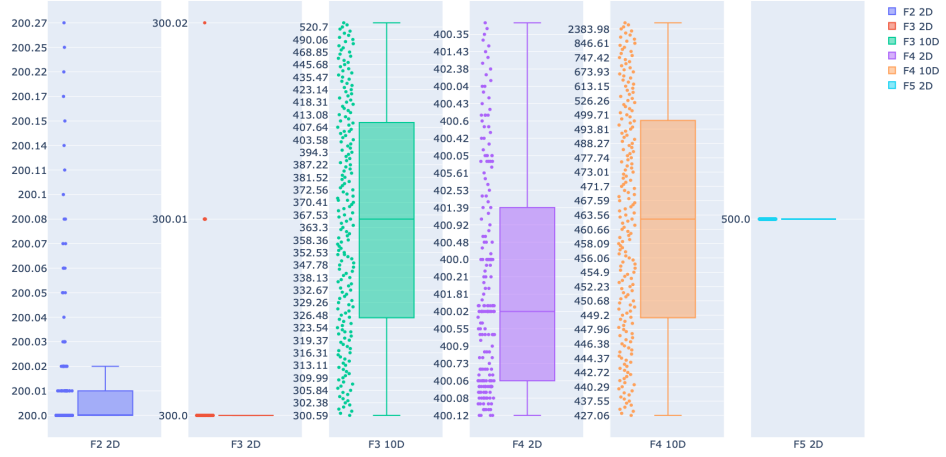


Figura A.5: Boxplots de los resultados del MCSA para las funciones benchmark del CEC2017, que evidencian la robustez del algoritmo frente a la variabilidad derivada de múltiples ejecuciones.

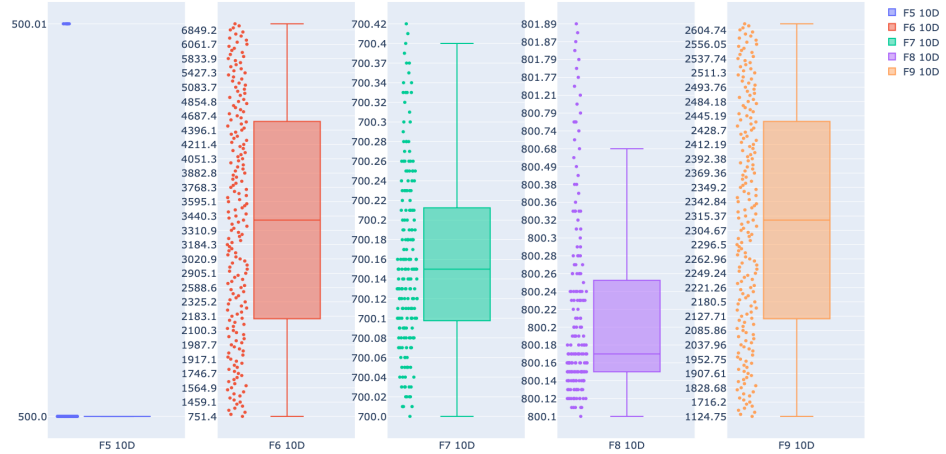


Figura A.6: Distribución de los resultados obtenidos por el MCSA en las funciones benchmark del CEC2017, mostrando la consistencia del algoritmo frente a la variabilidad entre ejecuciones.

Nomenclatura

Nota: las entradas se muestran en orden alfabético por Concepto / Acrónimo / Símbolo.

Nomenclatura

ABM	Agent-Based Modeling (Simulación Basada en Agentes)
ACO	Ant Colony Optimization
Area A	Zona destinada a atención de pacientes críticos (triaje 1–3)
Area B	Zona destinada a atención de pacientes menos graves (triaje 4–5)
BBOB	Black-Box Optimization Benchmarking
Benchmark	Funciones matemáticas de referencia para medir el rendimiento
border point	Punto que no es core pero pertenece al vecindario de un core point
CEC	Congress on Evolutionary Computation
COCO	COmparing Continuous Optimizers

NOMENCLATURA

core point	Punto central, muestra con suficientes vecinos en su vecindario ϵ
DBSCAN	Density-Based Spatial Clustering of Applications with Noise, algoritmo de clustering basado en densidad
DES	Discrete-Event Simulation (Simulación de Eventos Discretos)
ED	Emergency Department (Servicio de Urgencias Hospitalarias)
eps	Radio del vecindario considerado en DBSCAN
GA	Genetic Algorithm
HPC	High-Performance Computing
K-Means	Algoritmo de clustering basado en centroides que asume convexidad de los clusters
KPI	Key Performance Indicator (Indicador clave de desempeño)
LoS	Length of Stay (Tiempo medio de estancia)
MC	Montecarlo
MCSA	Montecarlo Search Algorithm
min_samples	Número mínimo de puntos requeridos para que una muestra sea core point
minPts	Notación alternativa usada en la literatura para min_samples
MTS	Manchester Triage System (Sistema de Triage de Manchester)

NOMENCLATURA

noise point / outlier	Punto considerado ruido, no pertenece a ningún cluster
Optimización	Proceso de encontrar la mejor solución posible dentro de un conjunto de restricciones
PSO	Particle Swarm Optimization
SA	Simulated Annealing
scikit-learn	Librería de Python para aprendizaje automático y análisis de datos
SD	System Dynamics (Dinámica de Sistemas)
SOO	Single-Objective Optimization
TSO	Tabu Search Optimization
$\bar{f}_{\text{prev}}, \bar{f}_{\text{curr}}$	Valor promedio de la función objetivo en bloques consecutivos
Δ	Amplitud de cada intervalo
ϵ	Radio que define el vecindario en DBSCAN (ϵ -neighborhood)
ϵ	Umbral de convergencia en el criterio de parada del algoritmo MCSA
λ_i	Peso relativo asignado al objetivo i en la combinación lineal ponderada, $\lambda_i \geq 0$, $\sum_i \lambda_i = 1$
$\mathbb{R}^n$	Espacio n-dimensional de los números reales
$\mathbf{x}^*$	Configuración óptima que minimiza la función objetivo
$\mathbf{x}^{(i)}$	Solución individual dentro del conjunto $\mathcal{S}$
$\mathbf{Z}(x)$	Vector de funciones objetivo de la solución x : $\mathbf{Z}(x) =$

NOMENCLATURA

	$(z_1(x), \dots, z_m(x))$
$\mathbf{z}_i$	Centroides individuales del conjunto $\mathcal{Z}$
$\mathcal{C}$	Conjunto de clusters obtenidos a partir del conjunto de soluciones $\mathcal{S}$
$\mathcal{C}^*$	Conjunto de clusters seleccionados con densidad máxima
$\mathcal{C}^{(m)}$	Conjunto de clusters en el intervalo I_m
$\mathcal{D}$	Dominio factible del problema
$\mathcal{E}_m$	Eficiencia de muestreo: proporción de muestras generadas respecto al tamaño del espacio de búsqueda exhaustivo
$\mathcal{I}$	Conjunto de intervalos en el rango de la función objetivo
$\mathcal{M}$	Mapa Montecarlo que contiene las muestras aptas
$\mathcal{S}$	Conjunto de soluciones candidatas generadas en la iteración del MCSA
$\mathcal{S}^{(m)}$	Submuestra de soluciones en el intervalo I_m
$\mathcal{S}_F$	Subconjunto factible de $\mathcal{S}$
$\mathcal{Z}$	Conjunto de centroides representativos obtenidos con K-Means
$\text{LoS}_{\text{MC15}}(\mathbf{x}^{(i)})$	Conjunto de 15 valores representativos de LoS para la configuración candidata i
$\text{LoS}_{\text{MCN},k}(\mathbf{x})$	Valor de LoS asociado al nivel de triaje k en la configuración $\mathbf{x}$

$\text{LoS}_{\text{MCN}}(\mathbf{x})$	Conjunto de valores de estancia obtenidos en N simulaciones para la configuración $\mathbf{x}$
$\mu\text{LoS}_{\text{KPI}}^{(i)}$	Media de LoS ponderada sobre los niveles de triaje y réplicas Montecarlo para la configuración i , usada como KPI global
μ_k	Media del valor objetivo de las muestras en el intervalo k
$\mu_{\text{LoS},k}(\mathbf{x})$	Media del LoS para nivel de triaje k asociada a la configuración $\mathbf{x}$
Ω	Espacio de soluciones del problema, que incluye todas las posibles combinaciones de variables antes de aplicar restricciones.
$\rho(C_j)$	Densidad de un cluster C_j
AreaA, AreaB	Subíndices para diferenciar recursos o pacientes asignados a las áreas A y B
Max_{tipo}	Número máximo permitido de profesionales en la categoría "tipo"
B	Tamaño del bloque de muestreo
C_j	Cluster individual dentro del conjunto $\mathcal{C}$
c_j^i	Beneficio del elemento j según el objetivo i en MOKP
C_p	Criterio de parada del procedimiento de reducción
D_k	Densidad local de muestras en el intervalo k
$f(\mathbf{x})$	Función objetivo definida a partir de $\mu\text{LoS}_{\text{KPI}}(\mathbf{x})$
$g_i(x)$	Restricciones de desigualdad, $i = 1, \dots, m$

NOMENCLATURA

$h_j(x)$	Restricciones de igualdad, $j = 1, \dots, p$
I_m	Intervalo m -ésimo del rango de valores de la función objetivo
M	Número de intervalos en el rango de la función objetivo
m	Número de objetivos en un problema multiobjetivo
m^*	Índice del intervalo con el máximo número de clusters
M_{red}	Mapa de Montecarlo reducido tras selección
N	Número de réplicas Montecarlo ejecutadas por configuración
n	Número de elementos disponibles o variables en el problema combinatorio
$P_{\max}$	Porcentaje máximo del espacio de búsqueda muestreado
$P_{\text{tipo, junior}}$	Número de profesionales de nivel <i>junior</i> en la categoría "tipo"
$P_{\text{tipo, senior}}$	Número de profesionales de nivel <i>senior</i> en la categoría "tipo"
q_m	Número de clusters en el intervalo I_m
S_k	Subconjunto de muestras en el intervalo k
$U(\Omega)$	Distribución uniforme sobre el espacio de búsqueda Ω
v_i	Valor o beneficio del elemento i
W	Capacidad máxima de la mochila o restricción global
w_i	Peso o coste del elemento i

NOMENCLATURA

x^1, x^2	Soluciones del MOKP usadas para definir dominancia de Pareto
x_i	Variable binaria que indica inclusión del elemento i en la mochila o solución combinatoria (1) o no (0)
x_j	Variable discreta de decisión para el recurso j (ej. médico, enfermero, técnico)
$y = g(\text{KPI})$	Función de evaluación que sintetiza el desempeño global de la configuración
Z	Valor total de la solución o función objetivo del problema
$Z_\lambda(x)$	Combinación lineal ponderada de objetivos: $Z_\lambda(x) = \sum_{i=1}^m \lambda_i z_i(x)$
$z_i(x)$	Valor de la función objetivo i para la solución x
MaxSteps	Número máximo de cortes o intervalos en la reducción del mapa

Referencias

- [1] A systematic review on the validity and reliability of an emergency department triage scale, the manchester triage system. *International Journal of Nursing Studies*, 51(7):1062–9, July 2014. ISSN 0020-7489. doi: 10.1016/j.ijnurstu.2014.01.013. 146
- [2] Z. A. Abdalkareem, A. Amir, M. A. Al-Betar, P. Ekhan, and A. I. Hammouri. Healthcare scheduling in optimization context: a review. *Health and Technology*, 11:445–469, 2021. 157
- [3] Benyamin Abdollahzadeh, Farhad Soleimanian Gharehchopogh, and Seyedali Mirjalili. African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Computers and Industrial Engineering*, 158:107408, August 2021. 111
- [4] Iman Ahmadianfar, Omid Bozorg-Haddad, and Xuefeng Chu. Gradient-based optimizer: A new metaheuristic optimization algorithm. *Information Sciences*, 540:131–159, 2020. doi: 10.1016/j.ins.2020.06.037. URL <https://www.sciencedirect.com/science/article/pii/S0020025520306241>. 162
- [5] Iman Ahmadianfar, Ali Asghar Heidari, Amir H. Gandomi, Xuefeng Chu, and Huiling Chen. Run beyond the metaphor: An efficient optimization algorithm based on runge kutta method. *Expert Systems with Applications*, 181:115079, 2021. doi: 10.1016/j.eswa.2021.115079. URL <https://www.sciencedirect.com/science/article/pii/S0957417421005200>. 159

REFERENCIAS

- [6] Iman Ahmadianfar, Ali Asghar Heidari, Saeed Noshadian, Huiling Chen, and Amir H. Gandomi. Info: An efficient optimization algorithm based on weighted mean of vectors. *Expert Systems with Applications*, 195: 116516, 2022. doi: 10.1016/j.eswa.2022.116516. URL <https://www.sciencedirect.com/science/article/pii/S0957417422000173>. 162
- [7] Iman Ahmadianfar, Arvin Samadi-Koucheksaraee, and Saman Razavi. Design of optimal operating rule curves for hydropower multi-reservoir systems by an influential optimization method. *Renewable Energy*, 211: 508–521, 2023. doi: 10.1016/j.renene.2023.04.113. URL <https://www.sciencedirect.com/science/article/pii/S0960148123005797>. 159
- [8] Inc. Amazon Web Services. Amazon web services (aws), 2024. URL <https://aws.amazon.com>. Accessed: 2024-11-13. 187
- [9] Atefeh Amindoust, Milad Asadpour, and Samineh Shirmohammadi. A hybrid genetic algorithm for nurse scheduling problem considering the fatigue factor. *Journal of Healthcare Engineering*, 2021:Article ID 5563651, 2021. doi: 10.1155/2021/5563651. 157
- [10] Mário Amorim Lopes, Álvaro Almeida, and Bernardo Almada-Lobo. Forecasting the medical workforce: a stochastic agent-based simulation approach. *Health Care Management Science*, 21, 2018. doi: 10.1007/s10729-016-9379-x. 152
- [11] Sigrún Andradóttir. A method for discrete stochastic optimization. *Management Science*, 41(12):1946–1961, December 1995. 110
- [12] Emanouil Atanassov and Sofiya Ivanovska. On the use of sobol’ sequence for high dimensional simulation. In *International Conference on Computational Science*, pages 646–652. Springer, 2022. 74
- [13] Hadi Bayzidi, Siamak Talatahari, Meysam Saraee, and Charles-Philippe Lamarche. Social network search for solving engineering optimization problems. *Computational Intelligence and Neuroscience*, 2021:32, 2021. doi: 10.1155/2021/8548639. 48

- [14] Richard Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, 1957. 10, 39
- [15] Leonora Bianchi et al. A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing*, 8(2):239–287, June 2009. 110
- [16] Marco Boresta, Tommaso Giovannelli, and Massimo Roma. Managing low-acuity patients in an emergency department through simulation-based multiobjective optimization using a neural network metamodel. *Health Care Management Science*, 27:415–435, 2024. URL <https://api.semanticscholar.org/CorpusID:270369303>. 143
- [17] Ilhem BoussaïD, Julien Lepagnot, and Patrick Siarry. A survey on optimization metaheuristics. *Inf. Sci.*, 237:82–117, July 2013. ISSN 0020-0255. doi: 10.1016/j.ins.2013.02.041. URL <https://doi.org/10.1016/j.ins.2013.02.041>. 36
- [18] Stephen Boyd and Lieven Vandenbergh. *Convex optimization*. Cambridge university press, 2004. 38
- [19] Sally C. Brailsford. Tutorial: Advances and challenges in healthcare simulation modeling. In *2007 Winter Simulation Conference*, pages 1436–1448. IEEE, 2007. 152
- [20] Eva Bruballa, Álvaro Wong, Dolores Rexachs, Emilio Luque, and Francisco Epelde. Evaluation of response capacity to patient attention demand in an emergency department. *International Journal on Advances in Systems and Measurements*, 10:11–22, 2017. 158
- [21] E. Cabrera, M. Taboada, M. L. Iglesias, F. Epelde, and E. Luque. Optimization of healthcare emergency departments by agent-based simulation. *Procedia Computer Science*, 4:1880–1889, 2011. 158
- [22] Eduardo Cabrera Flores. *Optimisation via Simulation for Healthcare Emergency Departments*. PhD thesis,

REFERENCIAS

- Universitat Autònoma de Barcelona, November 2013.
URL <https://portalrecerca.uab.cat/es/studentTheses/optimisation-via-simulation-for-healthcare-emergency-departments>.
Accessed 2025-09-12. 43
- [23] Jianbo Cai and Georg Thierauf. Evolution strategies for solving discrete optimization problems. *Advances in Engineering Software*, 25(2–3): 177–183, March 1996. 111
- [24] Sara Ceschia, Luca Di Gaspero, Vincenzo Mazzaracchio, Giuseppe Policante, and Andrea Schaerf. Solving a real-world nurse rostering problem by simulated annealing. *Operations Research for Health Care*, 36, 2023. doi: 10.1016/j.orhc.2023.100379. 157
- [25] P. S. Chen, W. T. Huang, T. H. Chiang, and G. Y.-H. Chen. Applying heuristic algorithms to solve inter-hospital hierarchical allocation and scheduling problems of medical staff. *International Journal of Computational Intelligence Systems*, 13(1):318–331, 2020. 157
- [26] Matej Črepinšek, Shih-Hsi Liu, and Marjan Mernik. Exploration and exploitation in evolutionary algorithms: A survey. *ACM Computing Surveys (CSUR)*, 45(3):1–33, July 2013. 111
- [27] Tome Eftimov and Peter Korošec. A novel statistical approach for comparing meta-heuristic stochastic optimization algorithms according to the distribution of solutions in the search space. *Information Sciences*, 489:255–273, July 2019. 111
- [28] M. Ester, H.P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of 2nd International Conference on Knowledge, Discovery and Data Mining*, volume 96, pages 226–231, August 1996. 72
- [29] Fedor V. Fomin and Petteri Kaski. Exact exponential algorithms. *Commun. ACM*, 56(3):80–88, March 2013. ISSN 0001-0782. doi: 10.1145/

- 2428556.2428575. URL <https://doi.org/10.1145/2428556.2428575>. 36
- [30] John J. Forbus and Daniel Berleant. Discrete-event simulation in healthcare settings: a review. *Modelling*, 3(4):417–433, 2022. 158
- [31] Francky Fouedjio. Clustering of multivariate geostatistical data. *Wiley Interdisciplinary Reviews: Computational Statistics*, 12(5):1510, September 2020. 111
- [32] A. Gaudiani, E. Luque, P. García, M. Re, M. Naiouf, and A. Di Giusti. Computing, a powerful tool for improving the parameters simulation quality in flood prediction. *Procedia Computer Science*, 29:299–309, January 2014. 116
- [33] Michel Gendreau, Jean-Yves Potvin, et al. *Handbook of metaheuristics*, volume 2. Springer, 2010. 40
- [34] Nigel Gilbert. *Agent-Based Models*. Quantitative Applications in the Social Sciences. Sage Publications, London, 2008. ISBN 9781412949644. 143
- [35] Fred Glover and Gary A. Kochenberger, editors. *Handbook of Metaheuristics*. International Series in Operations Research & Management Science. Springer, Boston, MA, 2003. ISBN 978-0-306-48056-0. doi: 10.1007/b101874. 41
- [36] W. Gong, L. Pang, J. Wang, M. Xia, and Y. Zhang. A social-aware k means clustering algorithm for d2d multicast communication under sdn architecture. *AEU-International Journal of Electronics and Communications*, 132:153610, April 2021. 111
- [37] W.-B. Gong, Y.-C. Ho, and Wengang Zhai. Stochastic comparison algorithm for discrete optimization with estimation. In *Proceedings of the 31st IEEE Conference on Decision and Control*, pages 795–800, December 1992. 110

REFERENCIAS

- [38] M. M. Gunal and M. Pidd. Discrete event simulation for performance modelling in health care: a review of the literature. *Journal of Simulation*, 4(1):42–51, 2010. doi: 10.1057/jos.2009.25. 152
- [39] Gurobi Optimization, LLC. Gurobi optimizer reference manual, 2024. URL <https://www.gurobi.com>. 114
- [40] HandWiki. Determining number of clusters in a data set. Encyclopedia, 2022. Disponible en: <https://encyclopedia.pub/entry/35594>, consultado el 04 de noviembre de 2025. 208
- [41] N. Hansen, A. Auger, R. Ros, O. Mersmann, T. Tušar, and D. Brockhoff. COCO: A platform for comparing continuous optimizers in a black-box setting. *Optimization Methods and Software*, 36(1):114–144, 2021. doi: 10.1080/10556788.2020.1808977. 50
- [42] M. Harita, A. Wong, D. Rexachs, and E. Luque. Kp01 solved by an n-dimensional sampling and clustering heuristic. In *International Conference on Computational Science*, pages 229–236. Springer International Publishing, June 2022. 162
- [43] Maria Harita, Alvaro Wong, Dolores Rexachs del Rosario, and Emilio Luque Fadón. Evaluation of a heuristic search algorithm based on sampling and clustering. In *IX Jornadas de Cloud Computing, Big Data & Emerging Topics (Modalidad virtual, 22 al 25 de junio de 2021)*, 2021. 162
- [44] Maria Harita, Alvaro Wong, Remo Suppi, Dolores Rexachs, and Emilio Luque. A metaheuristic search algorithm based on sampling and clustering. *IEEE Access*, 12:15493–15508, 2024. doi: 10.1109/ACCESS.2024.3354714. 70
- [45] Maria Harita, Alvaro Wong, Dolores Rexachs, Emilio Luque, Eva Bruballa, and Francisco Epelde. Application of a sampling and clustering-based heuristic search algorithm to find an efficient staff configuration in an emergency department. *Expert Systems with*

- Applications*, 295:128803, 2026. ISSN 0957-4174. doi: <https://doi.org/10.1016/j.eswa.2025.128803>. URL <https://www.sciencedirect.com/science/article/pii/S0957417425024212>.
- [46] G. Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, 14(1):55–63, 1968. doi: 10.1109/TIT.1968.1054102.
- [47] IBM. Ibm ilog cplex optimization studio, 2024. URL <https://www.ibm.com/products/ilog-cplex-optimization-studio>.
- [48] Abiodun M. Ikotun, Mubarak S. Almutari, and Absalom E. Ezugwu. K-means-based nature-inspired metaheuristic algorithms for automatic data clustering problems: Recent advances and future directions. *Applied Sciences*, 11(23), 2021. doi: 10.3390/app112311246.
- [49] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999. ISSN 0360-0300. doi: 10.1145/331499.331504. URL <https://doi.org/10.1145/331499.331504>.
- [50] M Janga Reddy and D Nagesh Kumar. Evolutionary algorithms, swarm intelligence methods, and their applications in water resources engineering: a state-of-the-art review. *H2Open Journal*, 3:135–188, 06 2020. doi: 10.2166/h2oj.2020.128.
- [51] Geethu Joy, Christian Huyck, and Xin-She Yang. Parameter tuning of the firefly algorithm by three tuning methods: Standard monte carlo, quasi-monte carlo and latin hypercube sampling methods. *Journal of Computational Science*, 87:102588, 2025. ISSN 1877-7503. doi: 10.1016/j.jocs.2025.102588. URL <https://www.sciencedirect.com/science/article/pii/S1877750325000651>.
- [52] Angel A. Juan, Javier Faulin, Scott E. Grasman, Markus Rabe, and Gonçalo Figueira. A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems. *Operations Research Perspectives*, 2:62–72, 2015. doi: 10.1016/j.orp.2015.03.

REFERENCIAS

001. URL <https://www.sciencedirect.com/science/article/pii/S221471601500007X>.
- [53] J. Jun, Sheldon Jacobson, and James Swisher. Application of discrete-event simulation in health care clinics: A survey. *Journal of the Operational Research Society*, 50:109–123, 1999. doi: 10.1057/palgrave.jors.2600669.
- [54] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Springer, 1972.
- [55] Jong-Hwan Kim and Hyun Myung. Evolutionary programming techniques for constrained optimization problems. *IEEE Transactions on Evolutionary Computation*, 1(2):129–140, July 1997.
- [56] Donald L. Kreher and Douglas R. Stinson. *Combinatorial Algorithms: Generation, Enumeration, and Search*. CRC Press, Boca Raton, FL, 1998. ISBN 0-8493-3988-X.
- [57] A. Lameesa, M. Hoque, M. S. B. Alam, S. F. Ahmed, and A. H. Gandomi. Role of metaheuristic algorithms in healthcare: a comprehensive investigation across clinical diagnosis, medical imaging, operations management, and public health. *Journal of Computational Design and Engineering*, 11(3):223–247, 2024.
- [58] Jiaxuan Liang. Review of combinatorial optimization methods in the intelligent era. In *Proceedings of the 2025 International Conference on Software Engineering and Computer Applications*, pages 128–134, 2025.
- [59] Jing J. Liang, B. Y. Qu, Ponnuthurai Nagaratnam Suganthan, and Alfredo G. Hernández-Díaz. Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization. Technical Report 34, Computational Intelligence Laboratory, Zhengzhou University and Nanyang Technological University, 2013.
- [60] Zhengchun Liu, Eduardo Cabrera, Dolores Rexachs, and Emilio Luque. A generalized agent-based model to simulate emergency departments. In

- Sixth International Conference on Advanced Systems Simulation*, pages 65–70, 2014.
- [61] Zhengchun Liu, Dolores Rexachs, Francisco Epelde, and Emilio Luque. A simulation and optimization based method for calibrating agent-based emergency department models under data scarcity. *Computers & Industrial Engineering*, 103:300–309, 2017.
- [62] Burt Loo and Wade He. Simulations used in geotechnical practice: Part 1—comparing monte carlo and latin hypercube sampling. In *8th International Symposium on Geotechnical Safety and Risk (ISGSR 2022)*, Newcastle, Australia, pages 14–16, 2022.
- [63] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, pages 281–297, 1967.
- [64] Kamran Mardani and Keivan Maghooli. Enhancing retinal blood vessel segmentation in medical images using combined segmentation modes extracted by dbscan and morphological reconstruction. *Biomedical Signal Processing and Control*, 69:102837, August 2021.
- [65] Rafael MARTÍ, J Moreno, et al. Meta heurísticos en optimización combinatoria. *World Wide Web*, <http://www.uv.es/~rmarti/heur.html>, 2000.
- [66] Sue McAvoy, Andrew Staib, and Gregory Treston. Can a system dynamics model of the emergency department show which levers reduce bottlenecks and delays to improve access to care? *Systems Research and Behavioral Science*, 38(1):61–79, 2021.
- [67] Mohammad Hossein Mehroolhassani, Anahita Behzadi, and Elaheh Asadipour. Key performance indicators in emergency department simulation: A scoping review. *Scandinavian Journal of Trauma*,

REFERENCIAS

- Resuscitation and Emergency Medicine*, 33(1):15, 2025. doi: 10.1186/s13049-024-01318-7.
- [68] Mohammad Hossein Mehrolohasani, Anahita Behzadi, and Elaheh Asadipour. Key performance indicators in emergency department simulation: A scoping review. *Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine*, 33(1):15, 2025. doi: 10.1186/s13049-024-01318-7.
- [69] Nicholas Metropolis and Stanislaw Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. URL https://web.williams.edu/Mathematics/sjmiller/public_html/105Sp10/handouts/MetropolisUlam_TheMonteCarloMethod.pdf.
- [70] M. Z. Naser, Mohammad Khaled Al-Bashiti, Arash Teymori Gharah Tapeh, Ahmad Naser, Venkatesh Kodur, Rami Hawileh, Jamal Abdalla, Nima Khodadadi, Amir H. Gandomi, and Armin Dadras Eslamlou. A review of benchmark and test functions for global optimization algorithms and metaheuristics. *WIREs Computational Statistics*, 17(2):e70028, 2025. doi: 10.1002/wics.70028. URL <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.70028>. e70028 EOCS-767.R3.
- [71] MZ Naser, Mohammad Khaled Al-Bashiti, Arash Teymori Gharah Tapeh, Ahmad Naser, Venkatesh Kodur, Rami Hawileh, Jamal Abdalla, Nima Khodadadi, Amir H Gandomi, and Armin Dadras Eslamlou. A review of benchmark and test functions for global optimization algorithms and metaheuristics. *Wiley Interdisciplinary Reviews: Computational Statistics*, 17(2):e70028, 2025.
- [72] Arnold Neumaier. Complete search in continuous global optimization and constraint satisfaction. *Acta Numerica*, 13:271–369, 2004. doi: 10.1017/S0962492904000194.
- [73] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer,

- New York, NY, 2nd edition, 2006. ISBN 978-0387303031. doi: 10.1007/978-0-387-40065-5.
- [74] B. N. Örnek, S. B. Aydemir, T. Düzenli, and B. Özak. A novel version of slime mould algorithm for global optimization and real world engineering problems: enhanced slime mould algorithm. *Mathematics and Computers in Simulation*, 198:253–258, August 2022.
- [75] Vikas Palakonda, Jae-Mo Kang, and Heechul Jung. Benchmarking real-world many-objective problems: A problem suite with baseline results. *IEEE Access*, 12:49275–49290, 2024. URL <https://api.semanticscholar.org/CorpusID:268864348>.
- [76] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [77] M. Penn, D. Hasson, and M. Avriel. Solving the 0–1 proportional knapsack problem by sampling. *Journal of Optimization Theory and Applications*, 80(2):289–297, 1994. doi: 10.1007/BF02192937.
- [78] Tribikram Pradhan, Akash Israni, and Manish Sharma. Solving the 0–1 knapsack problem using genetic algorithm and rough set theory. In *IEEE International Conference on Advanced Communications*, pages 1120–1125, May 2014.
- [79] Python Software Foundation. *Python 3: Powerful Object-Oriented Programming*, 2023. Versión 3.8. Disponible en: <https://www.python.org/>.
- [80] Santhana Raj and Dipanjan Ghosh. Improved and optimal dbscan for embedded applications using high-resolution automotive radar. In *21st International Radar Symposium (IRS)*, pages 343–346. IEEE, October 2020.

REFERENCIAS

- [81] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. doi: 10.1016/0377-0427(87)90125-7. Disponible en: <https://www.sciencedirect.com/science/article/pii/0377042787901257>.
- [82] Sudipendra Nath Roy, Bhavin J. Shah, and Hasmukh Gajjar. Application of simulation in healthcare service operations: A review and research agenda. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 31(1):1–23, 2020.
- [83] Arvin Samadi-Koucheksaraee, Seyedehelham Shirvani-Hosseini, Iman Ahmadianfar, and Bahram Gharabaghi. Optimization algorithms surpassing metaphor. In *Computational Intelligence for Water and Environmental Sciences*, pages 3–33. Springer, 2022.
- [84] Daniel Sanz-Alonso and Omar Al-Ghattas. A first course in monte carlo methods, 2024. URL <https://arxiv.org/abs/2405.16359>.
- [85] Julian F Schumann and Alejandro M Aragón. A machine learning approach for fighting the curse of dimensionality in global optimization. *arXiv preprint arXiv:2110.14985*, 2021.
- [86] Leyuan Shi. Nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability*, 2(3):271–291, September 2000.
- [87] Rainer Sibbel and Christoph Urban. Agent-based modeling and simulation for hospital management. *Coop Agents, Theory Decis Libr*, 32, 2011. doi: 10.1007/978-94-017-1177-7_11.
- [88] John Sterman. Business dynamics, system thinking and modeling for a complex world. [http://lst-iiiep.iiiep-unesco.org/cgi-bin/wwwi32.exe/\[in=epidoc1.in\]/?t2000=013598/\(1019, 2000](http://lst-iiiep.iiiep-unesco.org/cgi-bin/wwwi32.exe/[in=epidoc1.in]/?t2000=013598/(1019, 2000).

- [89] Jörg Stork, Agoston E Eiben, and Thomas Bartz-Beielstein. A new taxonomy of global optimization algorithms. *Natural Computing*, 21(2): 219–242, 2022.
- [90] Manel Taboada González. *Simulación del Servicio de Urgencias Hospitalarias: una aproximación computacional desarrollada mediante técnicas de Modelado Orientadas al Individuo (MoI)*. PhD thesis, Universitat Autònoma de Barcelona, Barcelona, 2013. URL <https://ddd.uab.cat/record/126902>. Tesis doctoral - Universitat Autònoma de Barcelona. Departament d'Arquitectura de Computadors i Sistemes Operatius.
- [91] El-Ghazali Talbi. *Metaheuristics: From Design to Implementation*. Wiley Publishing, 2009. ISBN 0470278587. URL <https://api.semanticscholar.org/CorpusID:5383918>.
- [92] Kay Chen Tan, Swee Chiang Chiam, A. A. Mamun, and Chi Keong Goh. Balancing exploration and exploitation with adaptive variation for evolutionary multi-objective optimization. *European Journal of Operational Research*, 197(2):701–713, September 2009.
- [93] S. Tisue and U. Wilensky. Netlogo: A simple environment for modeling complexity. In *International Conference on Complex Systems*, volume 21, pages 16–21, May 2004.
- [94] V. Tomar, M. Bansal, and P. Singh. Metaheuristic algorithms for optimization: A brief review. *Engineering Proceedings*, 59(1):238, 2024.
- [95] Ozgur Tutsoy and Mehmet Y. Tanrikulu. Priority and age specific vaccination algorithm for the pandemic diseases: A comprehensive parametric prediction model. *BMC Medical Informatics and Decision Making*, 22(1):4, 2022. doi: 10.1186/s12911-021-01720-6. URL <https://doi.org/10.1186/s12911-021-01720-6>.
- [96] C. Valouxis, C. Gogos, G. Goulas, P. Alefragis, and E. Housos. A

- systematic two phase approach for the nurse rostering problem. *European Journal of Operational Research*, 219:425–433, 2012.
- [97] Diego Vecillas Martin, Christian Berruezo Fernández, and Ángel M. Gento Municio. Systematic review of discrete event simulation in healthcare and statistics distributions. *Applied Sciences*, 15(4):1861, 2025.
 - [98] Pradnya A. Vikhar. Evolutionary algorithms: A critical review and its future prospects. In *International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC)*, pages 261–265. IEEE, December 2016.
 - [99] Jesús Isaac Vázquez-Serrano, Rodrigo E. Peimbert-García, and Leopoldo Eduardo Cárdenas-Barrón. Discrete-event simulation modeling in healthcare: a comprehensive review. *International Journal of Environmental Research and Public Health*, 18(22):12262, 2021.
 - [100] C. A. White, E. Nano, D. H. Nguyen-Ngoc, and G. M. White. An evaluation of certain heuristic optimization algorithms in scheduling medical doctors and medical students. In *Practice and Theory of Automated Timetabling VI: 6th International Conference, PATAT 2006 Brno, Czech Republic, August 30–September 1, 2006 Revised Selected Papers*, volume 6, pages 105–115. Springer, 2007.
 - [101] Uri Wilensky and William Rand. *An introduction to agent-based modeling: Modeling natural, social, and engineered complex systems with NetLogo*. MIT Press, 2015.
 - [102] Di Yan and H. Mukai. Stochastic discrete optimization. *SIAM Journal on Control and Optimization*, 30(3):594–612, May 1992.
 - [103] Xin-She Yang. Swarm intelligence based algorithms: A critical analysis. *Evolutionary Intelligence*, 7(1):17–28, 2014. doi: 10.1007/s12065-013-0101-8.
 - [104] Alexander Yastrebov, Łukasz Kubuś, and Katarzyna Poczeta. Multiobjective evolutionary algorithm idea and k-means clustering

- for modeling multidimensional medical data based on fuzzy cognitive maps. *Natural Computing*, 22(3):601–611, 2023.
- [105] Hu Zhang et al. Balancing exploration and exploitation in multiobjective evolutionary optimization. *Information Sciences*, 497:129–148, September 2019.

REFERENCIAS
