

ESCUELA DE INGENIERÍA DE TELECOMUNICACIÓN Y ELECTRÓNICA



TRABAJO FIN DE TÍTULO

ANÁLISIS DE ALGORITMOS PARA LA GENERACIÓN DE TRAYECTORIAS SAR CON IA

Titulación: Grado en Ingeniería en Tecnologías de la
Telecomunicación

Mención: Sistemas de Telecomunicación

Autor/a: María Almudena Jiménez Suárez

Tutor/a: Ricardo Aguasca Colomo, Jezabel M. Molina Gil

Cotutor/a: Francisco J. Navarro Amador (Capt. EAyE)

Fecha: Julio 2025

Agradecimientos

Quisiera expresar mi más sincero agradecimiento a la Base Aérea de Gando y al equipo del Centro de Coordinación de Salvamento Aéreo (ARCC) por la colaboración prestada a lo largo del desarrollo de este Trabajo de Final de Título. Su disposición para compartir experiencias reales, procedimientos operativos y perspectivas profesionales ha sido clave para comprender con mayor profundidad los retos que enfrentan las operaciones de búsqueda y rescate aéreo.

Mi agradecimiento especial al Capitán y al Comandante, cuya atención, cercanía y profesionalidad hicieron posible la visita a las instalaciones del Ala 46 y la participación en conversaciones técnicas de gran valor. Gracias a su implicación, fue posible identificar de forma directa las necesidades actuales del entorno operativo SAR, lo que ha permitido orientar este proyecto hacia soluciones aplicadas y de utilidad práctica.

Del mismo modo, valoro enormemente la oportunidad brindada en el marco del convenio de colaboración entre la Universidad de Las Palmas de Gran Canaria (ULPGC) y el Ejército del Aire y del Espacio (EAyE), que ha hecho posible este acercamiento a la realidad operativa, aportando un contexto realista, motivador y enriquecedor para el desarrollo del presente trabajo.

Finalmente, quiero agradecer a mi familia por estar siempre ahí y apoyarme incondicionalmente. Su presencia, ánimo y comprensión han sido fundamentales a lo largo de todo este proceso.

Resumen

Las operaciones de búsqueda y rescate (SAR) han evolucionado considerablemente con el paso del tiempo, especialmente en contextos marítimos donde la localización de personas desaparecidas representa un desafío técnico y operativo. Estas misiones, llevadas a cabo por aeronaves tripuladas como helicópteros o aviones, requieren de estrategias eficientes para cubrir grandes extensiones de terreno bajo condiciones de alta incertidumbre.

En este contexto, la inteligencia artificial (IA) se posiciona como una herramienta clave para optimizar la generación de trayectorias y mejorar los tiempos de respuesta. Este Trabajo de Final de Título se centra en el análisis y evaluación de cuatro algoritmos clásicos de búsqueda: Búsqueda en Amplitud (BFS), Búsqueda en Profundidad (DFS), Búsqueda Uniforme de Costo (UCS) y Expansión por Ondas (Wavefront Expansion), seleccionados por su aplicabilidad en entornos SAR reales.

A lo largo del proyecto, se ha desarrollado un entorno de simulación controlado que permite comparar el rendimiento de estos algoritmos en condiciones operativas representativas, mediante métricas como el tiempo de ejecución, la distancia recorrida o el número de nodos explorados. El objetivo final es identificar las ventajas y limitaciones de cada método y sentar las bases para futuras investigaciones que integren técnicas más avanzadas de inteligencia artificial en escenarios SAR, con el fin de incrementar la eficacia y seguridad de las misiones.

Abstract

Search and Rescue (SAR) operations have significantly evolved over time, especially in maritime environments where locating missing people poses major technical and logistical challenges. These missions, often conducted by manned aircraft such as helicopters or planes, demand efficient strategies to cover vast areas under high levels of uncertainty.

In this context, Artificial Intelligence (AI) emerges as a key tool to optimize path planning and improve response times. This Final Degree Project focuses on the analysis and evaluation of four classic search algorithms: Breadth-First Search (BFS), Depth-First Search (DFS), Uniform Cost Search (UCS), and Wavefront Expansion, selected for their potential applicability in real SAR scenarios.

Throughout the project, a controlled simulation environment was developed to compare the performance of these algorithms under representative operational conditions, using metrics such as execution time, distance traveled, and number of explored nodes. The goal is to identify the strengths and limitations of each method and lay the groundwork for future research that integrates more advanced AI techniques into SAR scenarios, with the aim of increasing the efficiency and safety of these missions.

Índice

Capítulo 1. Introducción	1
1. 1 Contexto	2
1.1.1 Historia y Creación de las Unidades SAR en el Ejército del Aire y del Espacio.....	2
1.1.2 Función del Grupo 82 de FFAA y Capacidades Operativas	2
1.1.3 Operaciones Notables de Rescate en Gran Canaria	3
1.1.4 El IAMSAR y su Aplicación en el Ejército del Aire y del Espacio para Operaciones SAR	4
1.1.5 Fases Críticas del Protocolo SAR y el Ejercicio del Grupo 82 de FFAA	5
1.1.6 Normativa y Protocolos	6
1.1.7 Rutas de Búsqueda en Operaciones SAR.....	7
1.1.8 Conclusión	9
1.1.9 Evolución de los Algoritmos de Búsqueda con Inteligencia Artificial	10
1.2 Motivación.....	14
1.3 Objetivos.....	15
1.4 Estructura del documento	17
Capítulo 2. Herramientas y Entorno de Desarrollo	19
2.1 Documentación de referencia	20
2.2 Entorno de escritura y organización.....	20
2.3 Entorno de programación y simulación	21
2.4 Librerías y paquetes de Python utilizados.....	22
Capítulo 3. Análisis teórico de los Algoritmos de búsqueda en operaciones SAR marítimas.....	23

3.1	Introducción al análisis de algoritmos de búsqueda.....	24
3.2	Características deseables en algoritmos SAR marítimos	25
3.3	Búsqueda en Amplitud (Breadth-First Search - BFS).....	27
3.3.1	Funcionamiento	27
3.3.2	Características principales.....	27
3.3.3	Ventajas	28
3.3.4	Desventajas	28
3.3.5	Aplicabilidad en SAR marítimo aéreo	28
3.4	Búsqueda en Profundidad (Depth-First Search - DFS).....	30
3.4.1	Funcionamiento	30
3.4.2	Características principales.....	30
3.4.3	Ventajas	30
3.4.4	Desventajas	31
3.4.5	Aplicabilidad en SAR marítimo aéreo	31
3.5	Búsqueda Uniforme de Costo (Uniform-Cost Search - UCS).....	32
3.5.1	Funcionamiento	32
3.5.2	Características principales.....	32
3.5.3	Ventajas	32
3.5.4	Desventajas	33
3.5.5	Aplicabilidad en SAR marítimo aéreo	33
3.6	Wavefront Expansion	34
3.6.1	Funcionamiento	34
3.6.2	Características principales.....	34
3.6.3	Ventajas	34
3.6.4	Desventajas	35
3.6.5	Aplicabilidad en SAR marítimo aéreo	35
3.7	Justificación	36

3.8	Elección metodológica y marco de decisión	38
3.9	Tabla comparativa algoritmos	39
3.10	Conclusión	41
Capítulo 4. Diseño conceptual y representación algorítmica de los métodos de búsqueda.....		42
4.1	Búsqueda en Amplitud (Breadth-First Search - BFS).....	43
4.1.1	Pseudocódigo de BFS	45
4.2	Búsqueda en Profundidad (Depth-First Search - DFS).....	47
4.2.1	Pseudocódigo de DFS	49
4.3	Búsqueda Uniforme de Costo (Uniform Cost Search - UCS).....	51
4.3.1	Pseudocódigo de UCS.....	53
4.4	Algoritmo Wavefront Expansion	55
4.4.1	Pseudocódigo de Wavefront Expansion	57
4.5	Consideraciones de aplicabilidad en operaciones SAR aéreas	61
4.5.1	Búsqueda en Amplitud (BFS).....	61
4.5.2	Búsqueda en Profundidad (DFS).....	61
4.5.3	Búsqueda de Coste Uniforma (UCS).....	62
4.5.4	Wavefront Expansion	62
4.6	Impacto de la incertidumbre y la ausencia de información precisa	64
4.6.1	Influencia del tipo de incertidumbre sobre la selección del algoritmo SAR	65
4.7	Conclusión	66
Capítulo 5. Diseño de escenarios de simulación para evaluación de algoritmos SAR.....		68
5.1	Introducción a los escenarios de simulación.....	69
5.2	Plantilla 1: Zona de búsqueda con expansión controlada	70
5.2.1	Objetivo y lógica del diseño	70
5.2.2	Representación gráfica y codificación.....	71

5.2.3	Funcionalidad específica de las Zonas de Alta Probabilidad (ZAP) 72	
5.2.4	Consideraciones metodológicas importantes	72
5.2.5	Cobertura, puntos ciegos y costos.....	73
5.2.6	Variabilidad y replicabilidad de la simulación	73
5.2.7	Ventajas y relevancia de la estrategia ZAP.....	73
5.2.8	Cono de deriva y efecto Coriolis	74
5.2.9	Caso real: hundimiento del buque Oleg Naydenov.....	75
5.2.10	Utilidad de la plantilla	75
5.2.11	Resultado.....	76
5.3	Plantilla 2: Entorno geográfico simulado en forma de “T”	77
5.3.1	Objetivo y lógica del diseño	77
5.3.2	Representación gráfica y codificación.....	78
5.3.3	Funcionalidad específica de las Zonas de Alta Probabilidad (ZAP) 78	
5.3.4	Consideraciones metodológicas importantes	79
5.3.5	Cobertura, puntos ciegos y costos.....	79
5.3.6	Variabilidad y replicabilidad de la simulación	80
5.3.7	Ventajas y relevancia de la estrategia ZAP.....	80
5.3.8	Caso real: caída del helicóptero Super Puma del SAR (2014) ...	81
5.3.9	Utilidad de la plantilla	81
5.3.10	Resultado.....	83
5.4	Consideración sobre el alcance de la búsqueda y la relevancia del diseño ⁸⁴	
5.5	Comparativa de las plantillas de simulación	85
5.6	Conclusión	86
Capítulo 6. Simulaciones y análisis de resultados.....		87
6.1	Introducción a las simulaciones	88

6.1.1	Consideración sobre el tiempo de ejecución	89
6.2	Análisis de Resultados en la Plantilla 1.....	90
6.2.1	Características observables gracias al diseño de la plantilla	91
6.2.2	Condiciones experimentales	92
6.2.3	Resultados obtenidos en la Plantilla 1	93
6.2.4	Análisis técnico de los resultados	93
6.2.5	Conclusiones específicas de la Plantilla 1	95
6.2.6	Resultados Gráficos Obtenidos Plantilla 1	96
6.3	Análisis de Resultados en la Plantilla 2.....	98
6.3.1	Características observables gracias al diseño de la plantilla	99
6.3.2	Condiciones experimentales	100
6.3.3	Resultados obtenidos en la Plantilla 2	101
6.3.4	Análisis técnico de resultados.....	101
6.3.5	Conclusiones específicas de la Plantilla 2	103
6.3.6	Resultados Gráficos Obtenidos Plantilla 2.....	104
6.4	Conclusiones generales del análisis experimental.....	106
6.4.1	Rendimiento computacional y eficiencia	106
6.4.2	Distancia recorrida y optimización de ruta	107
6.4.3	Cobertura, exploración y garantías	107
6.4.4	Sensibilidad al entorno: cómo afecta el diseño del mapa	108
6.4.5	Estabilidad de resultados y adaptabilidad	109
6.4.6	Aplicación operativa de los algoritmos SAR	109
6.5	Comparación con métodos SAR clásicos	111
6.5.1	Flexibilidad y adaptabilidad	111
6.5.2	Eficiencia de cobertura	111
6.5.3	Coste computacional vs operacional	112
6.5.4	Limitaciones operativas	112

6.5.5	Potencial de integración.....	112
6.5.6	Conclusión	112
6.6	Visualización de los resultados obtenidos en cada una de las simulaciones	114
Capítulo 7.	Líneas futuras	116
7.1	Posibles ampliaciones.....	117
Presupuesto	119
Desglose del Presupuesto	120
Materiales y recursos utilizados	120
Trabajo tarifado por tiempo empleado.....		122
Material fungible		123
Costes asociados a la elaboración y edición de la documentación		124
Tasas correspondientes al visado por parte del COITT		125
Gastos administrativos, de gestión y envío.....		126
Aplicación de los impuestos pertinentes según la normativa vigente		126

Índice de Figuras

Figura 1. Búsqueda en Cuadrado Expansivo	7
Figura 2. Búsqueda en Sectores	8
Figura 3. Búsqueda con regreso a lo largo de la derrota (BCRD(TSR)).....	8
Figura 4. Búsqueda sin regreso de la derrota (BSRD(TSN)).....	8
Figura 5. Búsqueda por Barrido Paralelo	9
Figura 6. Diagrama de Flujo Algoritmo BFS	44
Figura 7. Diagrama de Flujo Algoritmo DFS	48
Figura 8. Diagrama de Flujo Algoritmo UCS.....	52
Figura 9. Diagrama de Flujo Algoritmo Wavefront Expansion	56
Figura 10. Esquema selección del algoritmo SAR.....	65
Figura 11. Resultado Plantilla 1	76
Figura 12. Resultado Plantilla 2.....	83
Figura 13. Resultado simulación 44 con BFS Plantilla 1	96
Figura 14. Resultado simulación 48 con DFS Plantilla 1	96
Figura 15. Resultado simulación 11 con UCS Plantilla 1	97
Figura 16. Resultado simulación 38 con Wavefront Expansion Plantilla 1	97
Figura 17. Resultado simulación 43 con BFS Plantilla 2	104
Figura 18. Resultados simulación 28 con DFS Plantilla 2	104
Figura 19. Resultados simulación 26 con UCS Plantilla 2.....	105
Figura 20. Resultados simulación 12 con Wavefront Expansion Plantilla 2... ..	105

Índice de Tablas

Tabla 1. Librerías	22
Tabla 2. Justificación Aplicación SAR Algoritmo	37
Tabla 3. Tabla comparativa Algoritmos.....	40
Tabla 4. Consideraciones Operaciones SAR.....	63
Tabla 5. Resultados Plantilla 1	93
Tabla 6. Resultados Plantilla 2	101
Tabla 7. Materiales y Recursos.....	121
Tabla 8. Factor de corrección.....	123
Tabla 9. Costes elaboración y edición	124
Tabla 10. Aplicación de impuestos.....	127

Índice de Ecuaciones

Ecuación 1. Coste de amortización	122
Ecuación 2. Trabajo tarificado por tiempo empleado	122
Ecuación 3. Costes elaboración y edición.....	124
Ecuación 4. Derechos de visado del COITT	125

Lista de acrónimos

SAR	Search and Rescue
IA	Inteligencia Artificial
EA	Ejército del Aire y del Espacio
FFAA	Fuerzas Aéreas
IAMSAR	Manual Internacional Aeronáutico y Marítimo de Búsqueda y Rescate
EPIRB	Balizas de emergencia
MAYDAY	Llamadas de socorro
RCC	Centro Coordinador de Salvamento
OMI	Organización Marítima Internacional
OACI	Organización de Aviación Civil Internacional
BCE	Búsqueda en Cuadrado Expansivo
BS	Búsqueda por Sectores
BD	Búsqueda a lo Largo de la Derrota
BP	Búsqueda por Barrido Paralelo
SRI	Stanford Research Institute
NASA	National Aeronautics and Space Administration
RRT	Rapidly-exploring Random Tree
GA	Algoritmos Genéticos
RL	Aprendizaje por Refuerzo
DNNs	Redes Neuronales Profundas
DRL	Deep Reinforcement Learning

DE	Evolución Diferencial
BFS	Búsqueda en Amplitud
DFS	Búsqueda en Profundidad
UCS	Búsqueda de Costo Uniforme
LKP	Last Known Position
LIFO	Last In, First Out
ZAP	Zona de Alta Probabilidad
RNN	Redes Neuronales Recurrentes
COITT	Colegio Oficial de Ingenieros Técnicos de Telecomunicación
TFT	Trabajo de Final de Título

Capítulo 1. Introducción

En este capítulo se presentará el marco teórico necesario para comprender las operaciones de búsqueda y rescate (SAR), así como la evolución histórica de los algoritmos utilizados en estas misiones. Se analizarán los conceptos fundamentales, los avances tecnológicos y la transición de enfoques clásicos hacia técnicas modernas basadas en inteligencia artificial (IA). Este análisis proporcionará una visión integral de las aportaciones de estas herramientas en la mejora de la eficiencia y seguridad de las operaciones SAR, estableciendo una base para los estudios específicos desarrollados en los siguientes capítulos.

1.1 Contexto

El Ejército del Aire y del Espacio de España ha sido el estamento principal en la ejecución de operaciones de búsqueda y rescate (SAR) tanto a nivel nacional como internacional, siendo una de sus áreas de responsabilidad, las Islas Canarias. Estas islas, ubicadas en una posición geoestratégica en el Atlántico, han sido el escenario de numerosas misiones críticas de rescate. El Ejército del Aire y del Espacio, con su capacidad de respuesta con su proyección aérea inmediata y una infraestructura avanzada, ha jugado un papel clave en la región, liderando la respuesta ante emergencias marítimas, terrestres y aéreas, tanto civiles como militares.

1.1.1 Historia y Creación de las Unidades SAR en el Ejército del Aire y del Espacio

El Ejército del Aire y del espacio fue creado en 1939, y poco después se reconoció la necesidad de contar con unidades especializadas en rescate aéreo, dado el crecimiento del tráfico aéreo y el aumento de las operaciones militares en el mar. Durante las décadas posteriores a la Segunda Guerra Mundial, la expansión de rutas aéreas internacionales, la Guerra Fría, y el desarrollo de nuevas aeronaves en España hicieron que el papel de las operaciones SAR cobrara mayor relevancia.[1]

El Grupo 82 de Fuerzas Aéreas (FFAA en adelante), enclavado en la Base Aérea de Gando y Ala 46, es la unidad encargada de las misiones SAR en el archipiélago canario. Su creación fue una respuesta directa a la necesidad de cubrir con rapidez y eficacia emergencias en un área que es conocida por su gran extensión de aguas internacionales y las frecuentes condiciones climáticas adversas.[2]

1.1.2 Función del Grupo 82 de FFAA y Capacidades Operativas

Es responsable de las misiones de búsqueda y rescate y de Vigilancia Marítima dentro de un área de responsabilidad extremadamente amplia, que abarca gran parte del Océano Atlántico cerca de las costas africanas y alrededor de las Islas

Canarias. Esto requiere de una gran movilidad y capacidad operativa para enfrentarse a condiciones desafiantes en altamar.

Las principales aeronaves operadas por el Grupo 82 de FFAA son:

- 1. Helicópteros AS332 Super Puma (H215M):** Estas aeronaves son el núcleo del escuadrón para misiones de rescate aéreo. Los Super Puma están diseñados para operaciones en condiciones extremas, con capacidad para operar de noche, en climas adversos y en situaciones de rescate en el mar. Además, cuentan con cabestrantes de alta capacidad para el izado de personas desde embarcaciones o desde el agua, una herramienta esencial en situaciones de rescate de tripulaciones en peligro.[3]
- 2. Aviones CASA CN-235 VIGMA:** Estos aviones de patrulla marítima tienen la misión de realizar la vigilancia y localización de embarcaciones o aeronaves en peligro. Equipados con radares avanzados y sistemas de detección electrónica, son capaces de operar durante largas horas, cubriendo grandes áreas en misiones de búsqueda a través del océano.[4]

Además de estas aeronaves, el escuadrón SAR del Ejército del Aire está dotado con tecnología de última generación, como sistemas de visión infrarroja para búsqueda nocturna, cámaras de alta definición, y equipos de comunicación satelital que permiten coordinar eficazmente las misiones con otras fuerzas y agencias de rescate.

1.1.3 Operaciones Notables de Rescate en Gran Canaria

A lo largo de los años, el Grupo 82 de FFAA ha llevado a cabo numerosas misiones [5] que han salvado la vida de personas en situaciones críticas. Algunas de las operaciones más destacadas incluyen:

- 1. Rescate de tripulaciones en peligro en el Atlántico:** A lo largo de los años, el escuadrón ha realizado decenas de rescates de barcos que se han visto atrapados en tormentas o que han sufrido fallos en sus sistemas. La localización de pequeñas embarcaciones en un océano vasto es uno de los retos más grandes a los que se enfrenta el escuadrón, pero gracias

a los aviones CN-235 y los helicópteros Super Puma, se han podido salvar vidas en múltiples ocasiones.

- 2. Migraciones en pateras:** Uno de los desafíos más importantes en los últimos años ha sido la creciente llegada de inmigrantes en pateras desde las costas africanas. Muchas veces, estas embarcaciones se encuentran en condiciones extremadamente peligrosas y son difíciles de localizar en el mar. El Grupo 82 de FFAA ha jugado un papel fundamental en la localización y rescate de cientos de personas que se encontraban a la deriva en el Atlántico, muchas veces en situaciones de alto riesgo debido a las condiciones climáticas o al estado de las embarcaciones.
- 3. Accidentes aéreos:** El Ejército del Aire y del Espacio ha sido crucial en la respuesta a incidentes aéreos que involucran tanto a aviones comerciales como militares. En múltiples ocasiones, la rapidez del despliegue SAR ha sido determinante para la supervivencia de los pasajeros.

1.1.4 El IAMSAR y su Aplicación en el Ejército del Aire y del Espacio para Operaciones SAR

El Manual Internacional Aeronáutico y Marítimo de Búsqueda y Rescate (IAMSAR) es una guía fundamental en la coordinación de las operaciones SAR a nivel mundial. Su objetivo es estandarizar los procedimientos, garantizando que las operaciones de rescate sean eficientes y seguras, especialmente en situaciones donde la rapidez es crítica para salvar vidas. El Ejército del Aire y del Espacio de España, y en particular el Grupo 82 de FFAA en Gran Canaria, sigue estrictamente los protocolos del IAMSAR en sus misiones SAR.[6]

El Grupo 82 de FFAA es responsable de realizar operaciones de rescate aéreo en una extensa área que cubre las Islas Canarias y grandes porciones del Atlántico. Su papel es crucial en situaciones de emergencia, ya sea para rescatar tripulaciones de barcos en problemas, ayudar en accidentes aéreos o asistir en incidentes migratorios en alta mar. La integración de los principios IAMSAR en sus operaciones garantiza que se maximice la efectividad de las misiones y que se apliquen las mejores prácticas de búsqueda y rescate.

1.1.5 Fases Críticas del Protocolo SAR y el Ejercicio del Grupo 82 de FFAA

- 1. Notificación de Emergencia:** En el marco del IAMSAR, las operaciones SAR del Ejército del Aire y del Espacio se activan tras recibir una alerta por diversas fuentes: balizas de emergencia (EPIRB), llamadas de socorro (MAYDAY) o reportes visuales de otras aeronaves o embarcaciones. El Grupo 82 de FFAA, equipado con sistemas de detección avanzada, como radares SAR y comunicaciones satelitales, recibe estas alertas en el Centro Coordinador de Salvamento (RCC) en Las Palmas.
- 2. Evaluación de la Situación:** La evaluación inicial de la emergencia es un paso vital que influye directamente en la selección del patrón de búsqueda y el tipo de aeronave a desplegar. Por ejemplo, si se reporta una patera en el Atlántico, el RCC recopila datos de la última ubicación conocida, las condiciones meteorológicas y las corrientes marinas para diseñar la estrategia de búsqueda más efectiva. Los sistemas de inteligencia artificial pueden desempeñar un papel clave en la mejora de esta fase, al procesar grandes volúmenes de datos ambientales y optimizar la predicción de zonas de deriva.
- 3. Planificación de la Operación:** Siguiendo las directrices del IAMSAR, el Grupo 82 de FFAA selecciona rutas y patrones de búsqueda que se adaptan a la emergencia específica. Para incidentes en áreas grandes, se implementan patrones de búsqueda de sector o espiral, mientras que en áreas más reducidas o con mayor precisión de localización, se opta por patrones de cuadrícula o búsqueda paralela. En este aspecto, los algoritmos de búsqueda basados en inteligencia artificial pueden mejorar el cálculo de las rutas optimizadas, reduciendo el tiempo de respuesta y maximizando la cobertura del área de búsqueda.
- 4. Despliegue y Ejecución:** Durante la ejecución de la misión, los helicópteros Super Puma del Grupo 82 de FFAA, junto con los aviones CASA CN-235, siguen las rutas predefinidas para cubrir el área de búsqueda. Las aeronaves de ala fija están equipadas con cámaras de alta resolución, radar de búsqueda SAR y sensores térmicos que permiten

detectar supervivientes incluso en condiciones de baja visibilidad o de noche. Aquí, el IAMSAR establece la importancia de la exactitud en la cobertura del área de búsqueda para evitar pasar por alto posibles supervivientes. Las tecnologías de IA pueden ayudar a reconfigurar las rutas en función de las nuevas informaciones recibidas durante el vuelo.

- 5. Coordinación y Comunicación:** La coordinación es esencial para el éxito de una operación SAR, especialmente en un entorno marítimo y aéreo. Los aviones del Grupo 82 de FFAA están en comunicación constante con el RCC y otras unidades móviles, como barcos de Salvamento Marítimo. El IAMSAR destaca la importancia de la comunicación en tiempo real para ajustar los patrones de búsqueda según la evolución de las condiciones. En el futuro, la IA podría automatizar ciertos aspectos de la toma de decisiones, optimizando la distribución de las unidades SAR en el área de operaciones.
- 6. Rescate y Recuperación:** Una vez localizados los supervivientes, los helicópteros del Ejército del Aire y del Espacio inician la fase de rescate, utilizando sus sistemas de izado y personal altamente entrenado. La rapidez en esta fase es fundamental, y el cumplimiento estricto de los procedimientos establecidos por el IAMSAR garantiza que el rescate se lleve a cabo de manera segura y eficiente.
- 7. Conclusión y Revisión de la Operación:** Después de cada operación, se realiza una revisión exhaustiva en la que se analizan las rutas de búsqueda utilizadas y los resultados obtenidos. Esto es clave para mejorar futuras misiones. La integración de la IA en esta fase puede proporcionar análisis más detallados, detectando patrones y áreas de mejora en la planificación y ejecución de las rutas de búsqueda.

1.1.6 Normativa y Protocolos

Las operaciones SAR se rigen por estándares internacionales establecidos por la OMI y la OACI, y los procedimientos definidos en el IAMSAR. Los algoritmos de decisión guían la selección de patrones de búsqueda y rutas, que son esenciales para la eficacia de las operaciones.[7]

1.1.7 Rutas de Búsqueda en Operaciones SAR

Las rutas de búsqueda son fundamentales en la planificación y ejecución de operaciones SAR. Cada patrón de búsqueda está diseñado para maximizar la cobertura en diferentes escenarios, dependiendo de factores como la precisión de la última posición conocida, el tamaño del área de búsqueda, y las condiciones ambientales. A continuación, se detallan los principales tipos de rutas que se utilizan en operaciones SAR.[8]

1. Búsqueda en Cuadrado Expansivo (BCE o SS)

- **Descripción:** Este patrón se utiliza cuando la última posición conocida es bastante precisa, y se busca cubrir un área circundante en espiral desde el punto central. Es ideal para pequeñas áreas donde se presume que el objetivo está relativamente cerca de la posición informada.
- **Algoritmo de Decisión:** Se utiliza cuando la incertidumbre es baja y se necesita cubrir el área de manera uniforme y desde el centro hacia fuera.

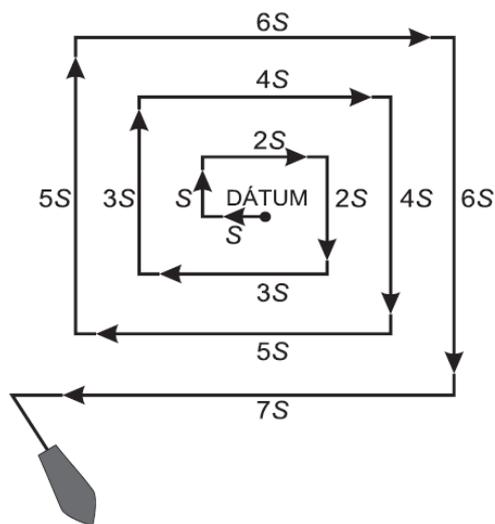


Figura 1. Búsqueda en Cuadrado Expansivo

2. Búsqueda por Sectores (BS o VS)

- **Descripción:** Este patrón divide el área de búsqueda en sectores, generalmente triangulares, y es adecuado cuando se tiene una buena

última posición conocida. Las unidades SAR realizan barridos sectoriales que convergen en el centro.

- **Algoritmo de Decisión:** Adecuado cuando hay buena visibilidad y se necesita una búsqueda intensiva en un área relativamente pequeña.

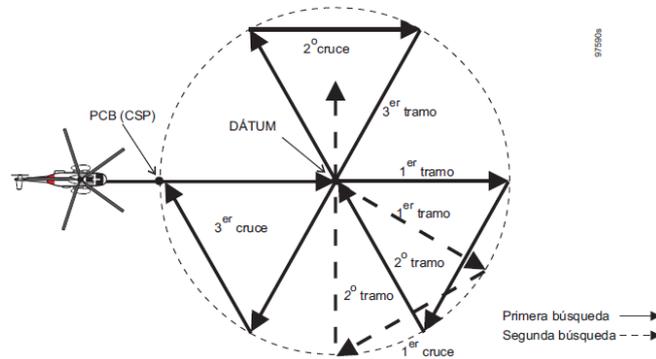


Figura 2. Búsqueda en Sectores

3. Búsqueda a lo Largo de la Derrota (BD o TS)

- **Descripción:** Se utiliza cuando se tiene una trayectoria conocida o probable de la embarcación o aeronave en emergencia. Se sigue la ruta más probable en la que podría haber ocurrido el incidente.
- **Algoritmo de Decisión:** Se elige este patrón cuando se tiene información sobre la ruta del objetivo, pero no sobre su ubicación exacta.

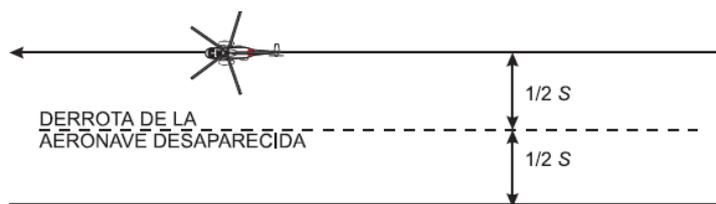


Figura 3. Búsqueda con regreso a lo largo de la derrota (BCRD(TSR))

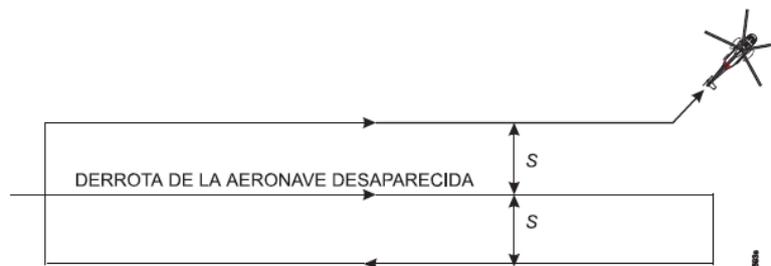


Figura 4. Búsqueda sin regreso de la derrota (BSRD(TSN))

4. Búsqueda por Barrido Paralelo (BP o PS)

- **Descripción:** Este patrón se emplea para cubrir grandes áreas de manera sistemática. Las unidades SAR se desplazan en líneas paralelas a través de la zona de búsqueda.
- **Algoritmo de Decisión:** Se selecciona cuando el área de búsqueda es extensa y se requiere una cobertura uniforme.

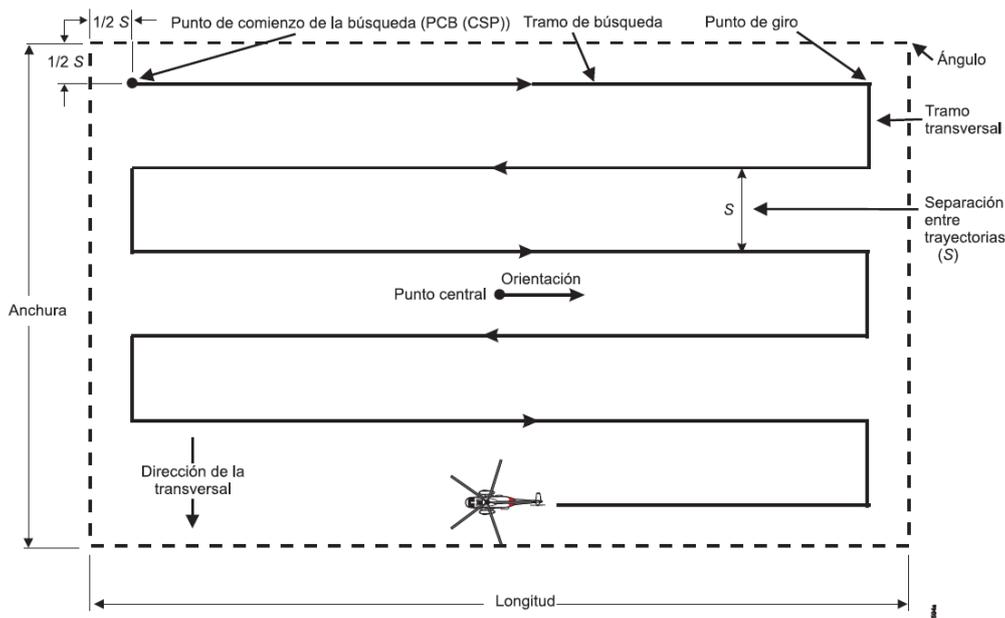


Figura 5. Búsqueda por Barrido Paralelo

1.1.8 Conclusión

Las rutas de búsqueda en operaciones SAR, como las detalladas anteriormente, son esenciales para la efectividad de las misiones de rescate. La incorporación de inteligencia artificial en la planificación y ejecución de estas rutas pueden tener el potencial de mejorar significativamente los tiempos de respuesta, la precisión en la detección y la utilización de recursos. El análisis de estas rutas para su mejora mediante IA permitirá optimizar las operaciones SAR, salvando más vidas y aumentando la eficiencia en situaciones críticas. Este enfoque tecnológico puede transformar la manera en que se abordan las operaciones de búsqueda y rescate, haciendo que sean más adaptativas y efectivas ante la incertidumbre y variabilidad de las condiciones operativas.

1.1.9 Evolución de los Algoritmos de Búsqueda con Inteligencia Artificial

La historia de los algoritmos de búsqueda en inteligencia artificial (IA) está profundamente entrelazada con el desarrollo de la robótica y los sistemas autónomos. Desde los primeros pasos dados por robots pioneros hasta los sistemas avanzados de la actualidad, los algoritmos de búsqueda han evolucionado para resolver problemas cada vez más complejos y adaptarse a entornos cada vez más dinámicos y desafiantes.

1. Década de 1960: SHAKEY y los Primeros Algoritmos de Búsqueda

En la década de 1960, el desarrollo de SHAKEY en el Stanford Research Institute (SRI), representó un hito en la integración de inteligencia artificial en robótica. Aunque SHAKEY no fue diseñado específicamente para misiones de rescate, sus avances en percepción y toma de decisiones fueron cruciales para sentar las bases de la robótica autónoma, que posteriormente se aplicaría en entornos complejos y cambiantes.

- **Algoritmo Clave: A*:** Este enfoque de búsqueda basado en grafos, utilizado por SHAKEY, combinaba la búsqueda por amplitud con heurísticas, permitiendo encontrar rutas óptimas en entornos parcialmente conocidos. Esta capacidad de planificación de rutas ha sido fundamental para aplicaciones donde el terreno y las condiciones pueden ser impredecibles.[9]
- **Aportaciones:** El uso de A* marcó el inicio de técnicas de búsqueda más eficientes, que luego serían adaptadas para mejorar la navegación en entornos desconocidos y peligrosos.

2. Década de 1970: MARS-ROVER y la Exploración en Terrenos Hostiles

En los años 70, la NASA desarrolló el MARS-ROVER, una plataforma diseñada para explorar terrenos hostiles como Marte. Aunque su misión principal no era el rescate, las tecnologías avanzadas utilizadas influyeron en futuras aplicaciones en situaciones de emergencia.[10]

- **Algoritmo Clave: D*:** Una extensión del A*, el D* fue diseñado para recalcular rutas en tiempo real en entornos dinámicos. Esta capacidad de

adaptación es crucial en escenarios donde las condiciones pueden cambiar rápidamente, como ocurre durante operaciones de rescate.

- **Aportaciones:** La habilidad de ajustar trayectorias en tiempo real permitió a las máquinas operar de manera más eficaz en entornos impredecibles, estableciendo un precedente para el desarrollo de sistemas de navegación que pueden reaccionar ante cambios imprevistos en el terreno.

3. Década de 1980: CART del SRI y la Precisión en la Navegación

El desarrollo de CART en los años 80 representó un avance significativo en la precisión de la navegación robótica. La capacidad de modelar y evitar obstáculos con mayor exactitud era un requisito clave en entornos desafiantes.[11]

- **Algoritmo Clave: RRT (Rapidly-exploring Random Tree):** Este algoritmo facilitó la planificación de movimientos en espacios complejos, explorando eficientemente áreas de búsqueda de alta dimensionalidad. Esta tecnología es vital en situaciones donde los robots deben maniobrar en entornos complejos para acceder a zonas difíciles.
- **Aportaciones:** La introducción de técnicas que permiten un modelado más preciso del entorno mejoró significativamente la capacidad de los robots para navegar en áreas complicadas, optimizando rutas no solo en términos de distancia, sino también en seguridad y eficiencia energética.

4. Década de 1990: Algoritmos Genéticos y Optimización Evolutiva

En los años 90, los Algoritmos Genéticos (GA) y otros enfoques evolutivos comenzaron a aplicarse a problemas complejos, incluyendo la optimización de rutas y decisiones en situaciones cambiantes.[12]

- **Aplicaciones:** Los GA se utilizaron para planificar rutas en tiempo real y ajustar los parámetros de control de manera adaptativa, mejorando la capacidad de los sistemas autónomos para responder a entornos dinámicos. Esto fue particularmente útil en la optimización de comportamientos en robots, permitiéndoles adaptarse y evolucionar para enfrentar nuevos desafíos.

- **Aportaciones:** La capacidad de explorar soluciones de manera global y evitar caer en óptimos locales mejoró la eficacia en la búsqueda de soluciones en entornos complicados, abriendo nuevas posibilidades para la automatización de tareas críticas.

5. Década de 2000: Algoritmos Basados en Modelos y Aprendizaje

La integración entre algoritmos de búsqueda y aprendizaje automático en la década de 2000 resultó en enfoques como el Aprendizaje por Refuerzo (RL), aplicados en la mejora continua de sistemas autónomos.

- **Algoritmos Clave:** Q-Learning y Deep Q-Learning (DQN) permitieron que los sistemas aprendieran y optimizaran sus acciones en función de recompensas obtenidas a través de la interacción con el entorno. Por otro lado, los Modelos Predictivos se utilizaron para anticipar y planificar acciones en tiempo real, mejorando la toma de decisiones en entornos críticos.[13]
- **Aportaciones:** Estas técnicas hicieron posible que los sistemas autónomos fueran más adaptativos, aprendiendo y mejorando su rendimiento en tiempo real, lo que es crucial cuando las condiciones pueden cambiar inesperadamente.

6. Década de 2010: Redes Neuronales y Algoritmos de Optimización Global

La década de 2010 estuvo marcada por la adopción masiva de Redes Neuronales Profundas (DNNs) y la combinación de estas con algoritmos de optimización, transformando la capacidad de los sistemas autónomos para resolver problemas complejos.

- **Algoritmos Clave:** El Deep Reinforcement Learning (DRL) combinó las DNNs con RL, permitiendo a los sistemas abordar problemas de búsqueda en entornos altamente complejos, como zonas urbanas colapsadas. La Evolución Diferencial (DE) se aplicó para optimizar rutas y estrategias de búsqueda, lo que fue esencial para mejorar la eficiencia operativa.[14]

- **Aportaciones:** La capacidad de resolver problemas en espacios de solución extremadamente complejos permitió a los sistemas autónomos operar con mayor precisión y adaptabilidad en condiciones extremas.

7. Década de 2020: IA Generativa y Aprendizaje Autónomo

En la década del 2020, la IA generativa y el aprendizaje autónomo llevaron las capacidades de los sistemas autónomos a un nuevo nivel, mejorando significativamente su adaptabilidad en escenarios dinámicos.

- **Algoritmos Clave:** Los modelos generativos como los Transformers se utilizan para desarrollar estrategias de búsqueda y planificación de acciones en tiempo real. Herramientas como AutoML y Meta-learning automatizan la selección y optimización de modelos, acelerando el despliegue de sistemas en situaciones críticas.[15]
- **Aportaciones:** Los avances en IA han incrementado la capacidad de los sistemas autónomos para adaptarse y aprender en tiempo real sin intervención humana, aumentando la eficacia y eficiencia en situaciones donde cada segundo cuenta.

1.2 Motivación

En una visita efectuada a la base aérea de Gando y al Ala 46, se observaron directamente los desafíos que enfrentan los equipos en operaciones de búsqueda y rescate (SAR). A través de conversaciones con los profesionales involucrados, se identificaron diversas dificultades y fuentes de incertidumbre en la ejecución de estas misiones. Los procedimientos operativos en misiones anteriores y, en particular, la falta de herramientas tecnológicas avanzadas fueron señalados como factores que afectan la eficiencia y la seguridad.

Las comunicaciones de selección de ruta aire-tierra se realizan a menudo mediante correo electrónico, un método lento y poco adecuado para situaciones de emergencia, que conlleva riesgos de interpretación y problemas de incomunicación, especialmente cuando las aeronaves se encuentran en zonas sin cobertura. Adicionalmente, se evidenció la carencia de un sistema de monitoreo constante, ya que actualmente se depende de software de rastreo gratuito como *Flightradar24*, una solución que resulta insuficiente para el nivel de precisión y agilidad que requieren estas misiones.

Este Trabajo de Final de Título se desarrolla al amparo del convenio de colaboración entre la Universidad de Las Palmas de Gran Canaria (ULPGC) y el Ejército del Aire y del Espacio (EAyE), lo cual ha permitido un acercamiento directo a las necesidades operativas reales de los equipos desplegados en misiones SAR, aportando así una base sólida y aplicada para el planteamiento de soluciones tecnológicas.

Ante estos problemas, el objetivo de este proyecto es adaptar algoritmos y evaluar su comportamiento con herramientas de búsqueda y rescate que integren inteligencia artificial, con el propósito de mejorar tanto la eficacia como la seguridad de las operaciones SAR. Se pretende ofrecer técnicas de búsqueda que permitan respuestas más rápidas y efectivas, que minimicen los riesgos para los equipos operativos y que, a su vez, optimicen los recursos, logrando así una solución más rentable.

1.3 Objetivos

El objetivo principal de este trabajo es llevar a cabo un análisis exhaustivo de los algoritmos de búsqueda marítima disponibles, enfocado en identificar y evaluar opciones que sean altamente eficaces, precisas y que, mediante la integración de técnicas de inteligencia artificial, optimicen la localización de personas y/o materiales en situaciones de emergencia, minimizando riesgos operativos y mejorando la toma de decisiones en tiempo real. Para alcanzar este objetivo, se han establecido los siguientes objetivos específicos:

Objetivo 1: Examinar y contextualizar las metodologías actuales empleadas en operaciones de búsqueda y rescate (SAR) en entornos marítimos, con especial atención a los procedimientos aprobados por los protocolos internacionales IAMSAR. Esto incluirá un análisis de las técnicas y herramientas comúnmente utilizadas, identificando sus limitaciones y fortalezas en función de factores como condiciones meteorológicas y disponibilidad tecnológica.

Objetivo 2: Realizar un análisis teórico de los algoritmos de búsqueda empleados en entornos marítimos, centrándonos en sus capacidades de eficiencia y precisión. Esto incluirá una evaluación de su rendimiento en diferentes condiciones, considerando factores operativos relevantes como velocidad de respuesta y precisión en áreas de búsqueda reducidas.

Objetivo 3: Realizar un análisis comparativo del rendimiento de cada algoritmo en situaciones que simulen condiciones realistas de búsqueda marítima, evaluando factores clave como el tiempo de respuesta y la precisión de localización. Este análisis permitirá identificar la eficacia de los algoritmos bajo condiciones similares a las de una operación SAR.

Objetivo 4: Realizar un análisis operativo con base en la evaluación de resultados de los algoritmos, integrando la perspectiva de expertos en operaciones SAR. Este análisis práctico incluirá opiniones y experiencias de profesionales del sector para valorar el rendimiento de los algoritmos en contextos reales, identificando fortalezas, limitaciones y posibles mejoras adaptadas a las necesidades específicas de las misiones SAR.

Objetivo 5: Elaborar conclusiones basadas en los resultados obtenidos de las simulaciones y análisis comparativos. Estas conclusiones se enfocarán en identificar áreas de mejora y proponer líneas de investigación futura que fomenten el desarrollo continuo de herramientas tecnológicas y algoritmos de búsqueda más efectivos y seguros, adaptados a las necesidades y limitaciones reales de las operaciones SAR marítimas.

1.4 Estructura del documento

La presente memoria se compone de siete capítulos, los cuales se describen a continuación:

- **Capítulo 1:** Introducción y contexto de las operaciones SAR. Se ofrece el marco teórico necesario para comprender las operaciones de búsqueda y rescate (SAR), abordando su evolución histórica, los principios básicos que las sustentan y el papel que han desempeñado los algoritmos en su desarrollo. Además, se contextualiza el uso de técnicas basadas en inteligencia artificial (IA), que han impulsado mejoras significativas en eficiencia y seguridad.
- **Capítulo 2:** Herramientas y recursos empleados. Se describen las herramientas utilizadas en el desarrollo del trabajo, tanto a nivel de software como de entornos de programación y simulación. También se detallan los recursos que han permitido estructurar y ejecutar las pruebas de forma controlada y reproducible.
- **Capítulo 3:** Análisis teórico de algoritmos de búsqueda aplicados a SAR. Se presenta un estudio detallado de cuatro algoritmos de búsqueda: BFS, DFS, UCS y Wavefront Expansion, seleccionados por su aplicabilidad en entornos marítimos con aeronaves tripuladas. Para cada uno se explican sus características, funcionamiento, ventajas, limitaciones y su adecuación a escenarios SAR.
- **Capítulo 4:** Diseño operativo de los algoritmos. Se expone la estructura interna y lógica de cada algoritmo mediante diagramas de flujo y pseudocódigo. Este análisis busca profundizar en su mecánica de funcionamiento y evaluar su idoneidad para operaciones SAR aéreas en mar abierto, considerando las particularidades del entorno.
- **Capítulo 5:** Desarrollo de escenarios de simulación. Se construyen dos plantillas operativas que representan entornos marítimos realistas. Se simula una búsqueda desde un punto conocido sin obstáculos. Ambas permiten evaluar los algoritmos en condiciones estandarizadas y comparables.

- **Capítulo 6:** Evaluación de resultados y conclusiones. Se analizan los datos obtenidos de las simulaciones mediante métricas como tiempo de ejecución, nodos explorados, distancia recorrida y tasa de éxito. Los resultados permiten identificar el comportamiento de cada algoritmo y su viabilidad en misiones SAR reales. A partir de estos resultados se presentan las conclusiones del trabajo.
- **Capítulo 7:** Propuestas de mejora y líneas futuras. Este capítulo recoge diversas propuestas para ampliar y perfeccionar el trabajo realizado. Se plantea, entre otras ideas, el uso de inteligencia artificial para modelar y predecir trayectorias de deriva de objetivos en el mar, la inclusión de variables ambientales como viento, corrientes o remolinos, y la exploración de nuevos algoritmos más avanzados. Estas líneas futuras están orientadas a optimizar aún más las operaciones SAR y fomentar el desarrollo de sistemas más robustos y adaptativos.

Capítulo 2. Herramientas y Entorno de Desarrollo

En este capítulo se describen las herramientas, entornos y recursos utilizados para el desarrollo del Trabajo de Final de Título. La elección de estas herramientas ha permitido estructurar y validar cada uno de los algoritmos, así como reproducir los experimentos de forma controlada.

2.1 Documentación de referencia

La base teórica del proyecto ha sido desarrollada mediante el uso de manuales y documentación oficial relacionada con las operaciones SAR, en particular:

- Manuales IAMSAR (Manual Internacional de los Servicios Aeronáuticos y Marítimos de Búsqueda y Salvamento): empleados como fuente normativa para comprender los protocolos y estructuras comunes en misiones SAR internacionales. Se ha utilizado específicamente la documentación del Ejército del Aire y del Espacio, que establece las fases críticas del protocolo SAR, la normativa y protocolos y las rutas de búsqueda en operaciones SAR.
- Otras fuentes teóricas han incluido artículos científicos y documentación de algoritmos de búsqueda con IA.

2.2 Entorno de escritura y organización

Para la elaboración del Trabajo de Final de Título, así como la gestión y búsqueda de información, se han utilizado las siguientes herramientas:

- **Microsoft Word:** procesador de textos empleado para la redacción y organización estructural de la memoria.
- **Google Search:** buscador utilizado para la localización de fuentes bibliográficas, artículos técnicos y documentación complementaria.
- **Google Drive:** espacio de almacenamiento en la nube donde se ha el progreso del trabajo y los scripts de simulación. Su uso ha facilitado el acceso remoto y la gestión segura de versiones del código.
- **Canva:** herramienta online empleada para la creación de los diagramas de flujo y representaciones esquemáticas que explican el funcionamiento de los algoritmos. Su interfaz intuitiva ha permitido una elaboración clara y profesional de los esquemas incluidos en el trabajo.[16]

- La gestión de referencias se ha realizado manualmente directamente dentro del documento Word, citando y organizando cada fuente según las normas académicas.

2.3 Entorno de programación y simulación

Para el desarrollo de los algoritmos y sus respectivas simulaciones se ha empleado:

- **Google Colab:** plataforma en la nube basada en Jupyter Notebooks, que permite ejecutar código Python sin necesidad de instalación local. Su integración con Google Drive ha facilitado el desarrollo, el almacenamiento automático y la visualización de resultados.[17]
- **Python:** lenguaje de programación seleccionado por su versatilidad, claridad sintáctica y extensas de librerías especializadas en cálculos, simulación y visualización. Permite una rápida iteración del código y una fácil depuración de errores.[18]

2.4 Librerías y paquetes de Python utilizados

A continuación, se detallan las librerías y módulos de Python empleados en las diferentes fases del proyecto, incluyendo la construcción de plantillas, la ejecución de algoritmos de búsqueda y la visualización de resultados.

Librería	Función Principal
numpy	Manipulación de matrices y operaciones matemáticas básicas
matplotlib.pyplot	Creación de gráficos y visualización de trayectorias y mapas
matplotlib.patches	Representación de zonas, leyendas y elementos gráficos
matplotlib.colors	Personalización de colores mediante ListedColormap
random	Generación de valores aleatorios, útil para destinos y condiciones variables
Time	Medición del tiempo de ejecución de los algoritmos
collections.deque	Estructura de datos eficiente para colas en algoritmos BFS y DFS
math	Operaciones matemáticas adicionales como raíces cuadradas y distancias
os	Interacción con el sistema de archivos para gestión de datos o entorno
matplotlib.lines	Creación de líneas personalizadas en las visualizaciones
heapq	Implementación de colas de prioridad eficiente (usado en el algoritmo UCS)

Tabla 1. Librerías

Estas librerías han sido seleccionadas por su fiabilidad, compatibilidad con Google Colab y rendimiento en simulaciones sobre entornos bidimensionales representados mediante las plantillas.

Capítulo 3. Análisis teórico de los Algoritmos de búsqueda en operaciones SAR marítimas

Este capítulo presenta un análisis teórico detallado de cuatro algoritmos de búsqueda seleccionados por su aplicabilidad en operaciones de búsqueda y rescate (SAR) marítimas con aeronaves tripuladas. Dada la naturaleza incierta de estas misiones, donde la posición del objetivo suele ser desconocida, se han elegido enfoques de búsqueda no informada capaces de explorar el entorno de manera eficiente y sistemática. Los algoritmos analizados son: Búsqueda en Amplitud (BFS), Búsqueda en Profundidad (DFS), Búsqueda de Costo Uniforme (UCS) y Expansión por Ondas (Wavefront Expansion). Para cada uno, se detallan su funcionamiento, variables clave, ventajas, limitaciones y su relevancia operativa en escenarios SAR reales. El capítulo concluye con una comparativa general que sienta las bases para su posterior implementación y evaluación práctica en simulaciones.

3.1 Introducción al análisis de algoritmos de búsqueda

Tras una revisión crítica de las necesidades operativas en entornos SAR marítimos, este capítulo desarrolla un análisis teórico centrado en algoritmos de búsqueda especialmente adecuados para situaciones en las que la ubicación del objetivo es desconocida. Este tipo de escenarios, frecuentes en operaciones de rescate con aeronaves tripuladas, requiere enfoques de búsqueda sistemáticos y eficientes que garanticen una cobertura amplia del área afectada, minimizando el tiempo de respuesta y maximizando las probabilidades de localización.

Con este propósito, se han seleccionado cuatro algoritmos de búsqueda: Búsqueda en Amplitud (Breadth-First Search, BFS), Búsqueda en Profundidad (Depth-First Search, DFS), Búsqueda de Costo Uniforme (Uniform-Cost Search, UCS) y el algoritmo de Expansión por Ondas (Wavefront Expansion). Estos métodos han sido elegidos no solo por su solidez teórica, sino también por su potencial para ser implementados en entornos de simulación representativos de misiones SAR reales, y por su idoneidad para explorar entornos amplios sin requerir información previa sobre el punto de destino.

Cada algoritmo será analizado en profundidad, describiendo su modo de operación, principales variables, ventajas, desventajas y aplicabilidad a operaciones SAR marítimas desde el aire. Este enfoque teórico permitirá sentar las bases para su implementación en simulaciones, donde se comparará su desempeño en función de métricas operativas clave como cobertura del área, tiempo de búsqueda y adaptabilidad a condiciones cambiantes.

3.2 Características deseables en algoritmos SAR marítimos

Las operaciones de búsqueda y rescate (SAR) en entornos marítimos presentan una serie de desafíos operativos que condicionan fuertemente el diseño y la selección de los algoritmos de búsqueda. A diferencia de otros contextos más estructurados o delimitados, el medio marino es amplio, dinámico, carente de referencias físicas estables y sujeto a factores ambientales cambiantes como el viento, las corrientes o la visibilidad. En consecuencia, los algoritmos empleados en este tipo de misiones deben cumplir con una serie de características clave que garanticen su utilidad en condiciones reales.

A continuación, se enumeran y describen las propiedades más relevantes que deben presentar los algoritmos de búsqueda aplicados a entornos SAR marítimos con aeronaves tripuladas:

- **Exploración sistemática del entorno:** Dado que en la mayoría de los casos la ubicación exacta del objetivo es desconocida, los algoritmos deben ser capaces de explorar progresivamente grandes áreas, evitando zonas redundantes o ya inspeccionadas, y cubriendo de forma eficiente el espacio disponible.
- **Independencia del destino:** En muchas misiones SAR, no se dispone de una posición final conocida. Por tanto, los algoritmos deben funcionar sin necesidad de conocer la ubicación del objetivo, y ser capaces de detectar el mismo si aparece dentro del área de búsqueda durante la exploración.
- **Capacidad de adaptación:** Las condiciones en el entorno marítimo pueden variar drásticamente durante una operación. Los algoritmos deben poder adaptarse a cambios en las condiciones de búsqueda, tales como nuevas zonas prioritarias o restricciones en el área de operación.
- **Robustez ante incertidumbre:** Los algoritmos deben tolerar la falta de información exacta o precisa sobre la ubicación del objetivo, así como

posibles errores en los datos de entrada, como la posición del último punto conocido (Last Known Position, LKP).

- **Eficiencia operativa:** El tiempo es un factor crítico en las operaciones SAR. Por ello, los algoritmos deben ser lo suficientemente eficientes computacionalmente como para permitir una planificación o ejecución rápida, especialmente cuando se integran en sistemas de apoyo a la toma de decisiones en tiempo real.
- **Escalabilidad:** Dado que el área de búsqueda puede abarcar decenas o cientos de kilómetros cuadrados, los algoritmos deben mantener su rendimiento aunque se incremente considerablemente el tamaño del entorno o el número de unidades de búsqueda.
- **Simplicidad y aplicabilidad:** En un entorno operacional donde la carga cognitiva del personal SAR es alta, los algoritmos deben ser fácilmente interpretable por los operadores o integrables en sistemas automatizados. La facilidad de implementación y mantenimiento también resulta clave para su adopción práctica.

En resumen, los algoritmos de búsqueda aplicables a operaciones SAR marítimas deben equilibrar la exploración eficaz de grandes áreas, la tolerancia a la incertidumbre, la eficiencia en tiempo y recursos, y la posibilidad de ser aplicados en sistemas reales. Estos criterios han guiado la selección de los cuatro algoritmos analizados en este capítulo, cuya idoneidad será posteriormente evaluada mediante simulaciones prácticas.

3.3 Búsqueda en Amplitud (Breadth-First Search - BFS)

La Búsqueda en Amplitud (Breadth-First Search, BFS) es uno de los algoritmos más básicos y ampliamente conocidos dentro de la teoría de grafos y la inteligencia artificial. Fue desarrollado por primera vez en la década de 1950 como parte de los estudios iniciales sobre autómatas y resolución de problemas. Su lógica se basa en explorar sistemáticamente los nodos de un grafo o espacio de búsqueda partiendo desde un nodo inicial, avanzando nivel por nivel, y expandiendo primero todos los nodos vecinos antes de continuar con los de mayor profundidad.

3.3.1 Funcionamiento

El algoritmo BFS opera utilizando una estructura de datos de tipo cola (FIFO), lo que le permite gestionar los nodos por orden de llegada. A cada paso, se toma el nodo más antiguo de la cola (el más cercano al origen) y se expanden sus vecinos no explorados, que se añaden al final de la cola. Este proceso se repite hasta que se encuentra el objetivo o hasta que todos los nodos alcanzables han sido visitados.[19]

En el contexto de entornos discretizados, como los mapas usados en simulaciones SAR, el algoritmo explora las celdas vecinas a partir de una celda inicial, y continúa hasta que encuentra el objetivo o completa la cobertura del área.

3.3.2 Características principales

- Explora primero los nodos más cercanos al punto de partida.
- Garantiza la ruta más corta en términos de número de pasos si el coste de cada paso es uniforme.
- Es completo, es decir, encuentra una solución si existe alguna.
- Utiliza una cola como estructura de control para almacenar los nodos pendientes de exploración.

- No tiene en cuenta información sobre el destino, lo que lo convierte en un algoritmo no informado o ciego.

3.3.3 Ventajas

- Ideal para escenarios sin conocimiento del objetivo, como en SAR, donde no se sabe dónde está la persona o embarcación desaparecida.
- Garantiza la cobertura uniforme del entorno, asegurando que se exploran primero las zonas más cercanas, lo que puede ser útil cuando se asume que el objetivo está cerca del último punto conocido (LKP).
- Simplicidad de implementación y bajo requerimiento de parametrización.
- Buen punto de partida como algoritmo base para comparaciones con métodos más complejos.

3.3.4 Desventajas

- Uso intensivo de memoria, ya que almacena todos los nodos en memoria a medida que expande el espacio. Esto puede ser problemático en entornos grandes como el mar abierto.
- Poco eficiente en términos de tiempo si el objetivo está muy alejado del punto de partida o no existe.
- No prioriza ninguna zona del entorno ni se adapta a información externa, lo que lo vuelve ineficiente en situaciones donde existen datos adicionales que podrían guiar la búsqueda.

3.3.5 Aplicabilidad en SAR marítimo aéreo

En misiones SAR aéreas sobre el mar, BFS representa una estrategia básica pero funcional para la exploración de áreas de búsqueda cuando no se dispone de información adicional sobre la ubicación del objetivo. Su naturaleza exhaustiva y su capacidad para generar rutas de cobertura completa hacen que sea útil en fases iniciales de búsqueda, especialmente en las cercanías del último punto conocido.[20]

Aunque su escalabilidad es limitada en grandes áreas, su valor reside en establecer una línea base de comparación con algoritmos más sofisticados.

Además, puede resultar útil cuando se combinan varios métodos, por ejemplo, aplicando BFS a áreas de alta prioridad antes de pasar a algoritmos más selectivos.

3.4 Búsqueda en Profundidad (Depth-First Search - DFS)

La Búsqueda en Profundidad (DFS) es otro algoritmo fundamental en la teoría de grafos, con orígenes que se remontan a la década de 1950, asociado a problemas de exploración de laberintos y análisis de grafos. Fue formalizado en trabajos iniciales de la informática teórica y es una técnica clave en muchas aplicaciones de inteligencia artificial y optimización.[21]

3.4.1 Funcionamiento

- DFS comienza en el nodo inicial y explora tan profundo como sea posible a lo largo de cada rama antes de retroceder (backtracking).
- Utiliza una estructura de datos LIFO (pila) o recursividad para controlar la exploración.
- Marca los nodos visitados para evitar ciclos y repeticiones.

3.4.2 Características principales

- No garantiza encontrar la ruta más corta, pero puede ser más eficiente en memoria que BFS.
- Puede usarse para detectar ciclos y componentes conectados en grafos.
- Es un algoritmo no informado que explora exhaustivamente un camino antes de explorar otros.
- Requiere memoria proporcional a la profundidad máxima de la búsqueda.

3.4.3 Ventajas

- Bajo consumo de memoria comparado con BFS, especialmente en grafos muy amplios.
- Fácil de implementar mediante recursión.

- Útil cuando la solución se encuentra en niveles profundos o específicos del árbol de búsqueda.

3.4.4 Desventajas

- Puede quedar atrapado explorando ramas profundas irrelevantes, retrasando la localización del objetivo.
- No es completo en grafos infinitos o con ciclos sin gestión adecuada.
- No garantiza la ruta más corta ni la búsqueda óptima.

3.4.5 Aplicabilidad en SAR marítimo aéreo

DFS puede ser útil en escenarios donde se requiere explorar rutas o áreas específicas en profundidad, por ejemplo, en la inspección detallada de zonas con alta probabilidad basadas en datos previos o señales parciales. Sin embargo, su tendencia a explorar profundamente sin criterio heurístico puede limitar su eficiencia en búsquedas iniciales con alta incertidumbre.[22]

3.5 Búsqueda Uniforme de Costo (Uniform-Cost Search - UCS)

La Búsqueda Uniforme de Costo (UCS) es un algoritmo de búsqueda no informada diseñado para encontrar el camino de costo mínimo desde un nodo inicial hasta un objetivo en grafos ponderados. Fue formalizado dentro del campo de la inteligencia artificial en la década de 1970, siendo una generalización del algoritmo de Dijkstra para búsqueda de caminos óptimos.[23]

3.5.1 Funcionamiento

- UCS expande nodos en orden creciente de costo acumulado desde el nodo inicial.
- Utiliza una estructura de datos de prioridad (cola de prioridad) para seleccionar el nodo con el menor costo hasta el momento.
- Mantiene un registro de los costos más bajos conocidos para alcanzar cada nodo, actualizándolos conforme se encuentran rutas más económicas.

3.5.2 Características principales

- Es un algoritmo óptimo, que garantiza encontrar la ruta de menor costo cuando los costos son no negativos.
- Es completo, siempre que exista una solución finita.
- No utiliza heurísticas, por lo que es un método no informado.
- La eficiencia depende del tamaño del espacio de búsqueda y la distribución de los costos.

3.5.3 Ventajas

- Permite manejar costos de diferentes magnitudes, lo que es crucial en entornos reales donde distancias o tiempos varían.

- Proporciona rutas óptimas en cuanto a costo, no solo en número de pasos.
- Su estructura es simple y puede ser implementado con variantes eficientes.

3.5.4 Desventajas

- Puede ser computacionalmente costoso en espacios de búsqueda muy grandes.
- Consume memoria considerable para mantener la cola de prioridad y los costos asociados.
- No se adapta dinámicamente a cambios en el entorno durante la búsqueda.

3.5.5 Aplicabilidad en SAR marítimo aéreo

UCS es especialmente útil en operaciones SAR donde las zonas de búsqueda tienen costos asociados distintos, por ejemplo, tiempo de vuelo, consumo de combustible o riesgo operacional variable según la distancia o condiciones meteorológicas. Esto permite optimizar la ruta no solo para cubrir áreas, sino también para minimizar recursos y maximizar la seguridad. Su uso es pertinente cuando se conoce o se puede estimar un costo realista para cada segmento de la búsqueda.

3.6 Wavefront Expansion

El algoritmo Wavefront Expansion es una técnica basada en la propagación gradual de una “frontera” u “onda” desde un punto inicial hacia el espacio de búsqueda, similar a la expansión en capas o niveles. Es utilizado frecuentemente en la planificación de rutas y búsqueda en entornos discretos, especialmente para cobertura y navegación en áreas amplias.[24]

3.6.1 Funcionamiento

- Comienza en el nodo inicial asignándole un valor inicial (usualmente cero).
- Propaga valores incrementales a los nodos vecinos, creando “frentes” o “olas” que se expanden en todas direcciones.
- La expansión continúa hasta alcanzar el objetivo o cubrir toda el área deseada.
- Puede utilizarse para generar mapas de costos o distancias desde un punto origen.

3.6.2 Características principales

- Es un algoritmo exhaustivo que asegura la exploración completa del espacio dentro de las limitaciones del mapa.
- Funciona bien en entornos discretos con obstáculos claramente definidos.
- Genera un mapa de distancias o tiempos desde el punto inicial a cualquier otro nodo.

3.6.3 Ventajas

- Muy efectivo para tareas de cobertura completa, como mapeo o búsqueda en áreas limitadas.
- Simple de implementar y entender, basado en técnicas similares a búsqueda en anchura.

- Puede integrarse con otros algoritmos para planificación más avanzada.

3.6.4 Desventajas

- Su exhaustividad puede hacerlo lento y costoso en términos computacionales para espacios muy grandes.
- No es un método heurístico ni optimizado para encontrar la ruta más corta rápidamente.
- Requiere que el mapa y los obstáculos estén claramente definidos y discretizados.

3.6.5 Aplicabilidad en SAR marítimo aéreo

En operaciones SAR, Wavefront Expansion es útil para garantizar una búsqueda sistemática y completa de áreas definidas, especialmente cuando el punto inicial es conocido y se desea asegurar que no quede ninguna zona sin cubrir. Su aplicación puede optimizar la cobertura y reducir el riesgo de omisión en zonas críticas, aunque puede ser menos eficiente en búsquedas iniciales con alta incertidumbre sobre la posición del objetivo.

3.7 Justificación

La selección de los algoritmos Búsqueda en Amplitud (BFS), Búsqueda en Profundidad (DFS), Búsqueda Uniforme de Costo (UCS) y Wavefront Expansion para este estudio responde a criterios técnicos, prácticos y de aplicabilidad real en operaciones de búsqueda y rescate (SAR) aéreas en entornos marítimos, especialmente cuando el objetivo es desconocido y la incertidumbre es alta. Estos algoritmos representan enfoques fundamentales y complementarios para cubrir diferentes escenarios de búsqueda, balanceando exhaustividad, rapidez y complejidad.

A continuación se presentan las razones específicas para la inclusión de cada uno:

- **Búsqueda en Amplitud (BFS):** Algoritmo no informado que garantiza la exploración por niveles del espacio de búsqueda. Su exhaustividad y garantía de encontrar la ruta más corta en espacios discretos uniformes lo hacen adecuado para la búsqueda sistemática en zonas desconocidas, aunque su coste computacional puede ser elevado en mapas grandes.
- **Búsqueda en Profundidad (DFS):** Algoritmo que explora profundamente cada rama antes de retroceder, permitiendo un consumo de memoria reducido y una rápida exploración en ciertos casos. Aunque no garantiza encontrar la ruta más corta ni la solución óptima, puede ser útil en escenarios donde se requiere una exploración rápida de caminos específicos o cuando la memoria es limitada.
- **Búsqueda Uniforme de Costo (UCS):** Algoritmo informado que encuentra la ruta de menor coste acumulado, considerando variaciones en costos entre nodos o movimientos. Esto resulta especialmente valioso en operaciones SAR aéreas para modelar diferentes condiciones operativas, como zonas con diferentes velocidades o riesgos, optimizando la ruta basada en criterios reales.

- **Wavefront Expansion:** Algoritmo exhaustivo basado en la propagación gradual de una “onda” desde el punto inicial. Aunque es simple y menos eficiente para grandes espacios, su capacidad para asegurar la cobertura completa del área lo convierte en una herramienta útil para la planificación y supervisión de búsquedas exhaustivas en áreas limitadas.

Algoritmo	Motivo de aplicación para SAR aéreo
BFS	Garantiza la exploración sistemática y la ruta más corta en mapas discretos uniformes. Ideal para búsqueda en zonas desconocidas.
DFS	Explora profundamente con bajo uso de memoria, útil para escenarios con limitaciones de recursos o exploraciones rápidas.
UCS	Encuentra la ruta de menor coste considerando variaciones en el entorno, útil para optimizar rutas basadas en condiciones reales.
Wavefront Expansion	Proporciona cobertura completa asegurada en áreas delimitadas, facilitando búsquedas exhaustivas y supervisión del progreso.

Tabla 2. Justificación Aplicación SAR Algoritmo

3.8 Elección metodológica y marco de decisión

La selección definitiva de los algoritmos incluidos en este estudio surge de un análisis comparativo entre enfoques de búsqueda informada y no informada, atendiendo a su aplicabilidad realista en misiones de búsqueda y rescate (SAR) aéreas en mar abierto. Los algoritmos de búsqueda no informada, como BFS y DFS, destacan por no requerir información previa del entorno, lo que los hace especialmente útiles en contextos con incertidumbre elevada, como suele ocurrir cuando se desconoce la ubicación exacta del objetivo. En contraste, algoritmos de búsqueda informada como UCS incorporan funciones de coste que permiten decisiones más eficientes cuando se dispone de información parcial o criterios operativos específicos.

Esta diversidad metodológica permite estudiar distintas estrategias de exploración que, combinadas, cubren un amplio espectro de necesidades: desde la cobertura sistemática del terreno hasta la optimización de recursos en función del coste. Se han priorizado algoritmos que no sólo tienen un valor académico o teórico, sino que pueden implementarse y evaluarse en escenarios simulados que reflejan condiciones reales de operación SAR.

Aunque se exploraron inicialmente otros métodos más avanzados o alternativos, se descartaron por razones de aplicabilidad directa, disponibilidad de datos o dificultad de integración en entornos controlados para simulación. Así, la selección actual representa un equilibrio entre simplicidad estructural, diversidad funcional y relevancia operativa.

3.9 Tabla comparativa algoritmos

A continuación se presenta una comparación entre los cuatro algoritmos seleccionados, considerando criterios fundamentales para operaciones SAR aéreas en entornos marítimos:

Criterio	BFS	DFS	UCS	Wavefront Expansion
Calidad de la trayectoria	Buena (ruta más corta garantizada en grafos uniformes)	Variable (no garantiza la ruta más corta)	Alta (ruta óptima considerando costes)	Media (cobertura completa, pero no optimiza ruta)
Tiempo de cálculo estimado	Alto en mapas grandes	Bajo a medio	Medio a alto según complejidad	Alto (exhaustivo)
Capacidad de adaptación	Baja (no adaptativo)	Baja	Media (permite costes variables)	Muy baja (solo expansión fija)
Uso de memoria	Alto (cola con todos los nodos por nivel)	Bajo (pila o recursión)	Variable según coste y estructura	Alto (almacena mapa completo)
Realismo de la ruta	Medio	Bajo	Alto (costos realistas)	Bajo (basado en expansión uniforme)

Facilidad de implementación	Alta	Muy alta	Media	Alta
Escalabilidad a mapas grandes	Limitada	Buena	Limitada a media	Limitada
Aplicabilidad SAR	Alta (búsqueda completa y garantizada)	Media (exploración rápida)	Alta (optimización realista)	Media (cobertura sistemática)
Ventaja destacada	Garantiza ruta más corta en entornos discretos uniformes	Bajo uso de memoria y simple	Considera costes y optimiza ruta	Cobertura completa asegurada
Limitación principal	Alto consumo de memoria y tiempo	No garantiza solución óptima	Costoso en tiempo para grandes mapas	No optimiza ruta, alto coste computacional

Tabla 3. Tabla comparativa Algoritmos

3.10 Conclusión

Los algoritmos presentados en este capítulo representan estrategias fundamentales para abordar la complejidad de las operaciones SAR aéreas en entornos marítimos con alta incertidumbre. BFS ofrece una búsqueda sistemática y garantiza la ruta más corta en espacios discretos uniformes, ideal para zonas donde el objetivo es completamente desconocido. DFS permite exploraciones rápidas y con bajo consumo de memoria, útil en ciertos escenarios específicos, aunque no garantiza la optimización. Por su parte, UCS aporta un enfoque más realista al considerar costes variables, optimizando rutas en función de condiciones operativas, lo cual puede incrementar la eficiencia y seguridad en la misión. Finalmente, Wavefront Expansion asegura una cobertura completa y controlada del área de búsqueda, facilitando la supervisión y exhaustividad en operaciones críticas.

La elección del algoritmo más adecuado dependerá de las características concretas de cada misión SAR, la disponibilidad de recursos computacionales y la precisión requerida. En los capítulos posteriores, se procederá a la implementación práctica en Python, seguida de simulaciones que permitirán evaluar su desempeño en escenarios realistas y ajustar su aplicación a las necesidades operativas.

Capítulo 4. Diseño conceptual y representación algorítmica de los métodos de búsqueda

Este capítulo detalla el diseño conceptual y la estructura operativa de los cuatro algoritmos seleccionados para el estudio: búsqueda en amplitud (BFS), búsqueda en profundidad (DFS), búsqueda uniforme de costo (UCS) y Wavefront Expansion. El objetivo es proporcionar una comprensión clara y profunda de su funcionamiento interno antes de su implementación práctica.

Para lograrlo, utilizaremos diagramas de flujo y pseudocódigos visuales que ilustrarán paso a paso la lógica que sigue cada método en la exploración del espacio de búsqueda. Se destacarán las particularidades que los hacen idóneos o limitados para su aplicación en operaciones de búsqueda y rescate (SAR) aéreas en mar abierto.

Este análisis previo es crucial para los procesos que rigen la generación de trayectorias y la exploración del entorno, sentando las bases para el desarrollo y la simulación que se abordarán en los capítulos posteriores. Asimismo, se examinarán consideraciones específicas que condicionan la aplicabilidad de cada algoritmo en escenarios reales, donde la incertidumbre y la ausencia de información precisa sobre la ubicación del objetivo son factores determinantes.

4.1 Búsqueda en Amplitud (Breadth-First Search - BFS)

La Búsqueda en Amplitud (BFS) es un algoritmo fundamental y ampliamente utilizado para la exploración de espacios de búsqueda discretos. Su característica principal es la exploración sistemática por niveles o “capas” del grafo o cuadrícula, comenzando desde el nodo inicial y expandiendo simultáneamente todos los nodos vecinos antes de avanzar a los nodos del siguiente nivel.

Este método garantiza encontrar la solución con el menor número de pasos o nodos visitados, siempre que el coste entre nodos sea uniforme. Esto lo convierte en una opción viable para operaciones SAR cuando la prioridad es asegurar una búsqueda exhaustiva y ordenada sin conocimiento previo de la ubicación del objetivo.

En el contexto de las operaciones aéreas de búsqueda y rescate en mar abierto, BFS permite un barrido progresivo y sistemático del área de búsqueda, facilitando una cobertura completa y minimizando la probabilidad de omitir zonas. Sin embargo, su principal limitación radica en el alto consumo de memoria y tiempo cuando el espacio de búsqueda es muy amplio o complejo, lo que podría afectar la rapidez en situaciones críticas.

A continuación, se presenta el diagrama de flujo que representa la lógica de operación del algoritmo BFS, seguido de una explicación detallada de cada etapa.

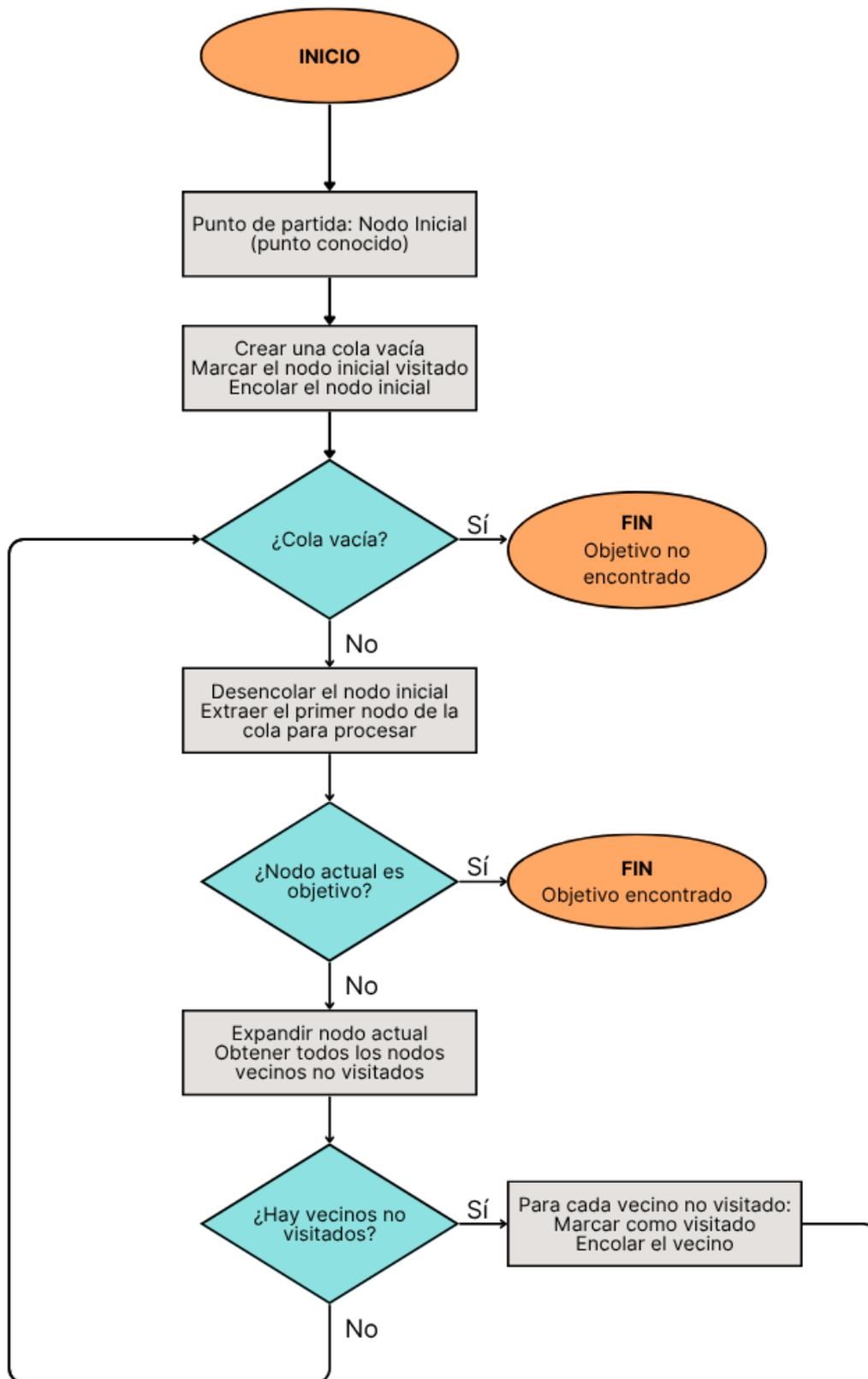


Figura 6. Diagrama de Flujo Algoritmo BFS

4.1.1 Pseudocódigo de BFS

Para una comprensión más formal de su funcionamiento, se presenta el pseudocódigo del algoritmo BFS.

ALGORITMO BFS(mapa, punto_inicio, punto_objetivo, centro_ZAP, radio_ZAP):

1. SI punto_inicio NO ES VÁLIDO en mapa:
2. RETORNAR NULO, LISTA_VACÍA, FALSO

3. CREAR COLA: Q
4. CREAR CONJUNTO: VISITADOS
5. CREAR DICCIONARIO: PADRES (para reconstruir la ruta)
6. CREAR LISTA: SECUENCIA_EXPLORACION

7. AÑADIR (punto_inicio, [punto_inicio]) a Q
8. AÑADIR punto_inicio a VISITADOS

9. MIENTRAS Q NO ESTÉ VACÍA:
10. (current_y, current_x), camino_actual = SACAR_DE_COLA(Q)
11. current_pos = (current_y, current_x)
12. AÑADIR current_pos a SECUENCIA_EXPLORACION

13. SI current_pos ES IGUAL a punto_objetivo:
14. RETORNAR camino_actual, SECUENCIA_EXPLORACION, VERDADERO

15. PARA CADA (dy, dx) en DIRECCIONES (arriba, abajo, izquierda, derecha):
16. neighbor_y = current_y + dy
17. neighbor_x = current_x + dx
18. neighbor_pos = (neighbor_y, neighbor_x)

19. SI ES_VALIDO(neighbor_pos, mapa) Y
20. DISTANCIA(centro_ZAP, neighbor_pos) <= radio_ZAP Y
21. neighbor_pos NO ESTÁ en VISITADOS:
22. AÑADIR neighbor_pos a VISITADOS
23. AÑADIR (neighbor_pos, camino_actual + [neighbor_pos])
a Q

24. RETORNAR NULO, SECUENCIA_EXPLORACION, FALSO

4.2 Búsqueda en Profundidad (Depth-First Search - DFS)

La Búsqueda en Profundidad (DFS) es un algoritmo que, a diferencia de BFS, explora tan profundamente como sea posible a lo largo de cada rama antes de retroceder. Comienza en el nodo raíz y se mueve por un camino hasta que alcanza un nodo que no tiene hijos no visitados o el objetivo. En ese punto, retrocede (backtracking) y explora otra rama.

Este enfoque de "todo o nada" en la exploración de una rama puede ser eficiente en términos de memoria si la profundidad de la solución no es muy grande, ya que solo necesita almacenar el camino actual. Sin embargo, no garantiza encontrar la ruta más corta, ya que puede encontrar una solución subóptima si la primera rama que explora conduce a un objetivo distante mientras que una rama no explorada más superficial contenía una solución más cercana.

En el contexto SAR, DFS podría ser útil en escenarios donde la rapidez para encontrar cualquier rastro es la prioridad, y el espacio de búsqueda es muy profundo pero estrecho. Por ejemplo, si se sospecha que un objetivo ha seguido una trayectoria lineal. No obstante, su incapacidad para garantizar la ruta más corta y el riesgo de quedar "atrapado" en una rama muy larga (lo que podría llevar a un tiempo de búsqueda excesivo si el objetivo está en otra rama o no existe en la rama actual) lo hacen menos predecible para una cobertura sistemática.

A continuación, se presenta el diagrama de flujo que representa la lógica de operación del algoritmo DFS, seguido de una explicación detallada de cada etapa.

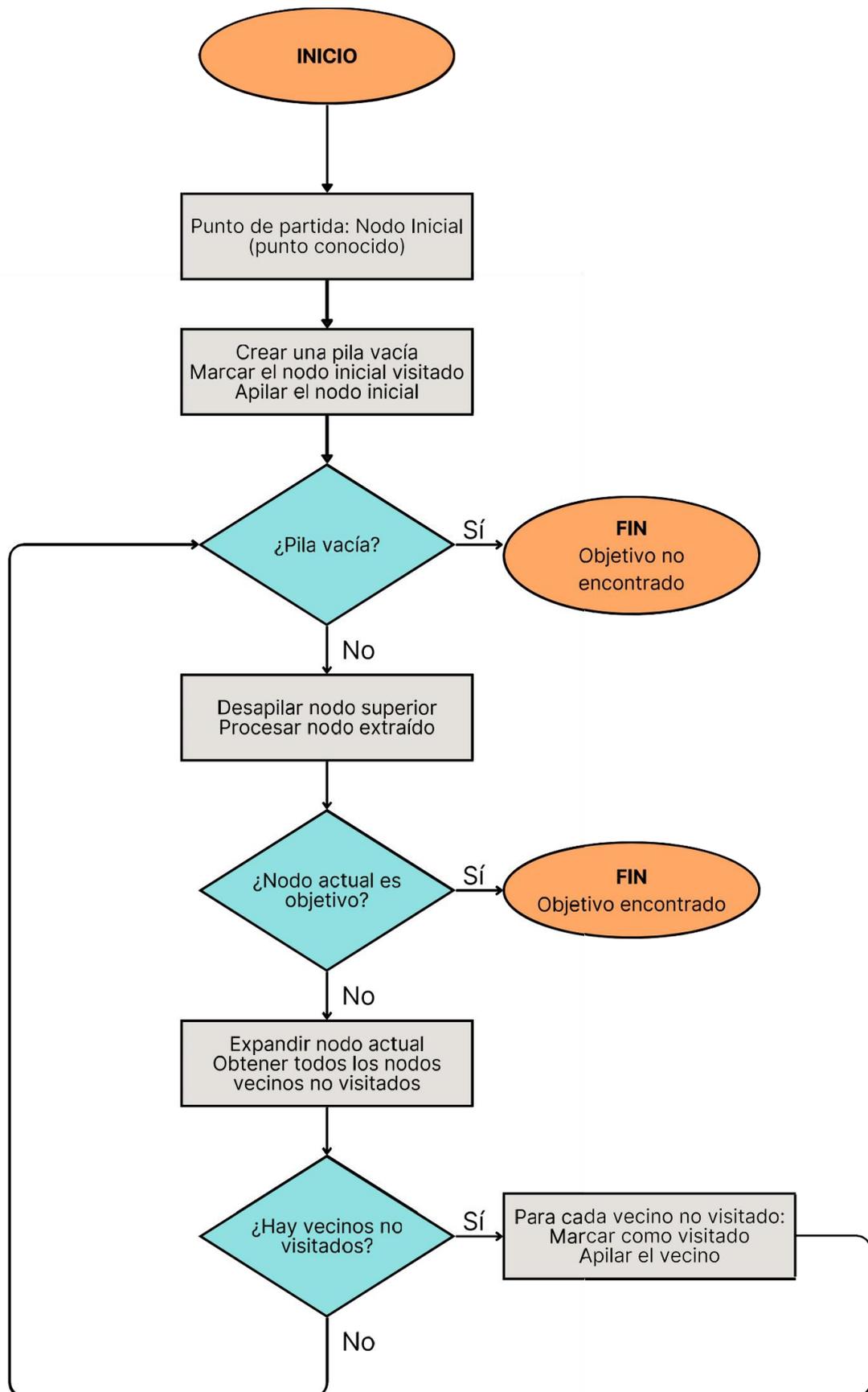


Figura 7. Diagrama de Flujo Algoritmo DFS

4.2.1 Pseudocódigo de DFS

A continuación, se presenta el pseudocódigo del algoritmo DFS para una comprensión más estructurada.

ALGORITMO DFS(mapa, punto_inicio, punto_objetivo, centro_ZAP, radio_ZAP):

1. SI punto_inicio NO ES VÁLIDO en mapa:
2. RETORNAR NULO, LISTA_VACÍA, FALSO

3. CREAR PILA: P
4. CREAR CONJUNTO: VISITADOS
5. CREAR LISTA: SECUENCIA_EXPLORACION

6. AÑADIR (punto_inicio, [punto_inicio]) a P
7. AÑADIR punto_inicio a VISITADOS

8. MIENTRAS P NO ESTÉ VACÍA:
9. (current_y, current_x), camino_actual = SACAR_DE_PILA(P)
10. current_pos = (current_y, current_x)
11. AÑADIR current_pos a SECUENCIA_EXPLORACION (si no está ya para evitar duplicados en la visualización)

12. SI current_pos ES IGUAL a punto_objetivo:
13. RETORNAR camino_actual, SECUENCIA_EXPLORACION, VERDADERO

14. CREAR LISTA: VECINOS_PARA_PILAR (para mantener orden y explorar correctamente)

15. PARA CADA (dy, dx) en DIRECCIONES (arriba, abajo, izquierda, derecha):

16. neighbor_y = current_y + dy

17. neighbor_x = current_x + dx

18. neighbor_pos = (neighbor_y, neighbor_x)

19. SI ES_VALIDO(neighbor_pos, mapa) Y

20. DISTANCIA(centro_ZAP, neighbor_pos) <= radio_ZAP Y

21. neighbor_pos NO ESTÁ en VISITADOS:

22. AÑADIR neighbor_pos a VISITADOS

23. AÑADIR (neighbor_pos, camino_actual + [neighbor_pos])
a VECINOS_PARA_PILAR

24. PARA CADA vecino en VECINOS_PARA_PILAR en ORDEN INVERSO
(para mantener el orden de exploración):

25. AÑADIR vecino a P

26. RETORNAR NULO, SECUENCIA_EXPLORACION, FALSO

4.3 Búsqueda Uniforme de Costo (Uniform Cost Search - UCS)

La Búsqueda Uniforme de Costo (UCS) es un algoritmo de búsqueda informado que expande los nodos en orden de su costo acumulado desde el nodo inicial. A diferencia de BFS, que prioriza la profundidad, UCS prioriza el costo total más bajo para llegar a un nodo. Utiliza una cola de prioridad para gestionar los nodos a expandir, garantizando que siempre se seleccione el nodo con el menor costo hasta el momento.

Este algoritmo es óptimo para encontrar la ruta de menor costo cuando los costos de las aristas (pasos entre nodos) son variables y no uniformes. Si los costos de las aristas son uniformes (como en nuestro modelo de cuadrícula donde cada paso cuesta 1), UCS se comporta de manera idéntica a BFS en términos de la ruta encontrada, aunque puede diferir en el orden de exploración de nodos con el mismo costo.

En el contexto SAR, UCS ofrece una ventaja si el costo de la exploración varía entre diferentes puntos del mapa (por ejemplo, zonas con más turbulencia, zonas con visibilidad reducida que requieren más tiempo de vuelo, o zonas de mayor riesgo). En nuestro escenario actual de mar abierto y cuadrícula uniforme, donde cada paso tiene un costo de 1, UCS se comporta como un BFS. Sin embargo, su capacidad para manejar costos variables lo hace conceptualmente superior para escenarios más complejos. Su principal limitación es que no utiliza información heurística sobre la distancia al objetivo, lo que puede hacerlo menos eficiente que algoritmos como A* en espacios de búsqueda muy grandes con costos variables.

A continuación, se presenta el diagrama de flujo que representa la lógica de operación del algoritmo UCS, seguido de una explicación detallada de cada etapa.

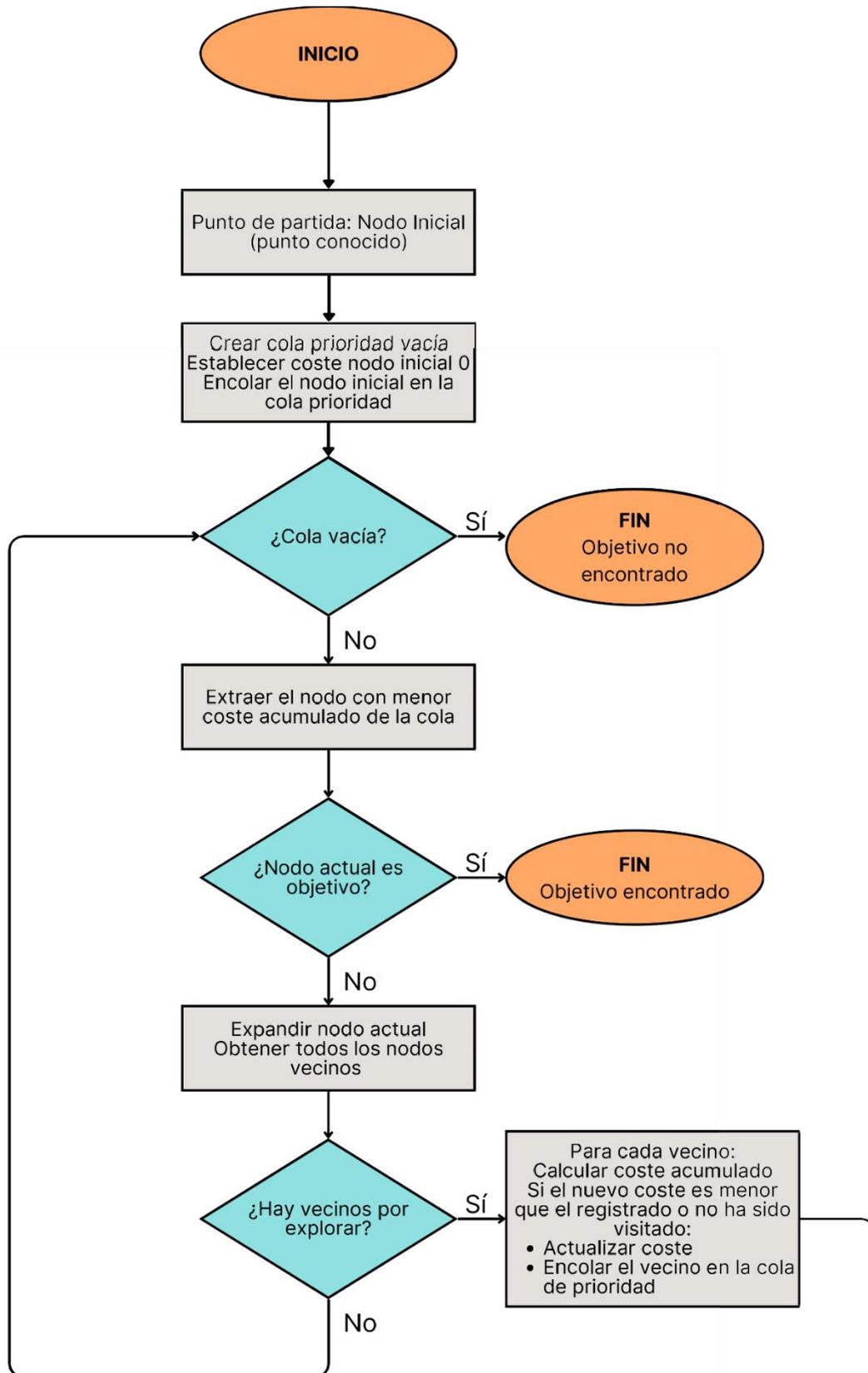


Figura 8. Diagrama de Flujo Algoritmo UCS

4.3.1 Pseudocódigo de UCS

Para una representación estructurada, se incluye el pseudocódigo de UCS.

ALGORITMO UCS(mapa, punto_inicio, punto_objetivo, centro_ZAP, radio_ZAP):

1. SI punto_inicio NO ES VÁLIDO en mapa:
2. RETORNAR NULO, LISTA_VACÍA, FALSO

3. CREAR COLA_PRIORIDAD: PQ (almacena (costo, nodo, camino))
4. CREAR DICCIONARIO: COSTOS (almacena el costo mínimo para llegar a cada nodo)
5. CREAR LISTA: SECUENCIA_EXPLORACION

6. AÑADIR (0, punto_inicio, [punto_inicio]) a PQ
7. ESTABLECER COSTOS[punto_inicio] = 0

8. MIENTRAS PQ NO ESTÉ VACÍA:
9. current_cost, (current_y, current_x), camino_actual = SACAR_MIN_DE_COLA_PRIORIDAD(PQ)
10. current_pos = (current_y, current_x)

11. SI current_cost > COSTOS[current_pos]: (ya encontramos un camino más barato)
12. CONTINUAR

13. AÑADIR current_pos a SECUENCIA_EXPLORACION
14. SI current_pos ES IGUAL a punto_objetivo:
15. RETORNAR camino_actual, SECUENCIA_EXPLORACION, VERDADERO
16. PARA CADA (dy, dx) en DIRECCIONES (arriba, abajo, izquierda, derecha):
17. neighbor_y = current_y + dy
18. neighbor_x = current_x + dx
19. neighbor_pos = (neighbor_y, neighbor_x)
20. costo_a_vecino = 1 (costo uniforme por paso)
21. new_cost = current_cost + costo_a_vecino
22. SI ES_VALIDO(neighbor_pos, mapa) Y
23. DISTANCIA(centro_ZAP, neighbor_pos) <= radio_ZAP Y
24. new_cost < OBTENER_COSTO(COSTOS, neighbor_pos, INFINITO):
25. ESTABLECER COSTOS[neighbor_pos] = new_cost
26. AÑADIR (new_cost, neighbor_pos, camino_actual + [neighbor_pos]) a PQ
27. RETORNAR NULO, SECUENCIA_EXPLORACION, FALSO

4.4 Algoritmo Wavefront Expansion

El algoritmo Wavefront Expansion, también conocido como Expansión por Ondas, es un método de búsqueda que se diferencia de los anteriores en su enfoque. En lugar de buscar directamente desde el inicio hacia el objetivo, Wavefront propaga una "onda" de información de costo desde el objetivo hacia el resto del espacio de búsqueda. Una vez que el mapa de costos (o "mapa de ondas") se ha inundado, la ruta óptima desde cualquier punto de inicio al objetivo se puede reconstruir simplemente siguiendo el gradiente de costos decrecientes.

Este algoritmo es particularmente efectivo para planificación de rutas en entornos estáticos donde el objetivo es fijo y se necesitan encontrar rutas desde múltiples puntos de partida, o cuando se desea una visualización clara de la "propagación" de la información de distancia. Garantiza encontrar la ruta más corta (óptima) en cuadrículas con costos uniformes, similar a BFS, pero lo hace en dos fases: una fase de inundación (propagación) y una fase de reconstrucción de la ruta.

En el contexto de operaciones SAR, Wavefront Expansion ofrece una perspectiva interesante. La fase de inundación proporciona un mapa de "cercanía" al objetivo desde cualquier punto dentro de la Zona de Alta Probabilidad (ZAP), lo cual podría ser útil para la coordinación de múltiples unidades de búsqueda. Sin embargo, su desventaja es la necesidad de construir y almacenar el mapa de ondas completo, lo que puede requerir una memoria significativa para mapas muy grandes. La fase de inundación puede ser costosa si el espacio de búsqueda es muy extenso y el objetivo está muy alejado.

A continuación, se presenta el diagrama de flujo que representa la lógica de operación del algoritmo Wavefront Expansion, seguido de una explicación detallada de cada etapa.

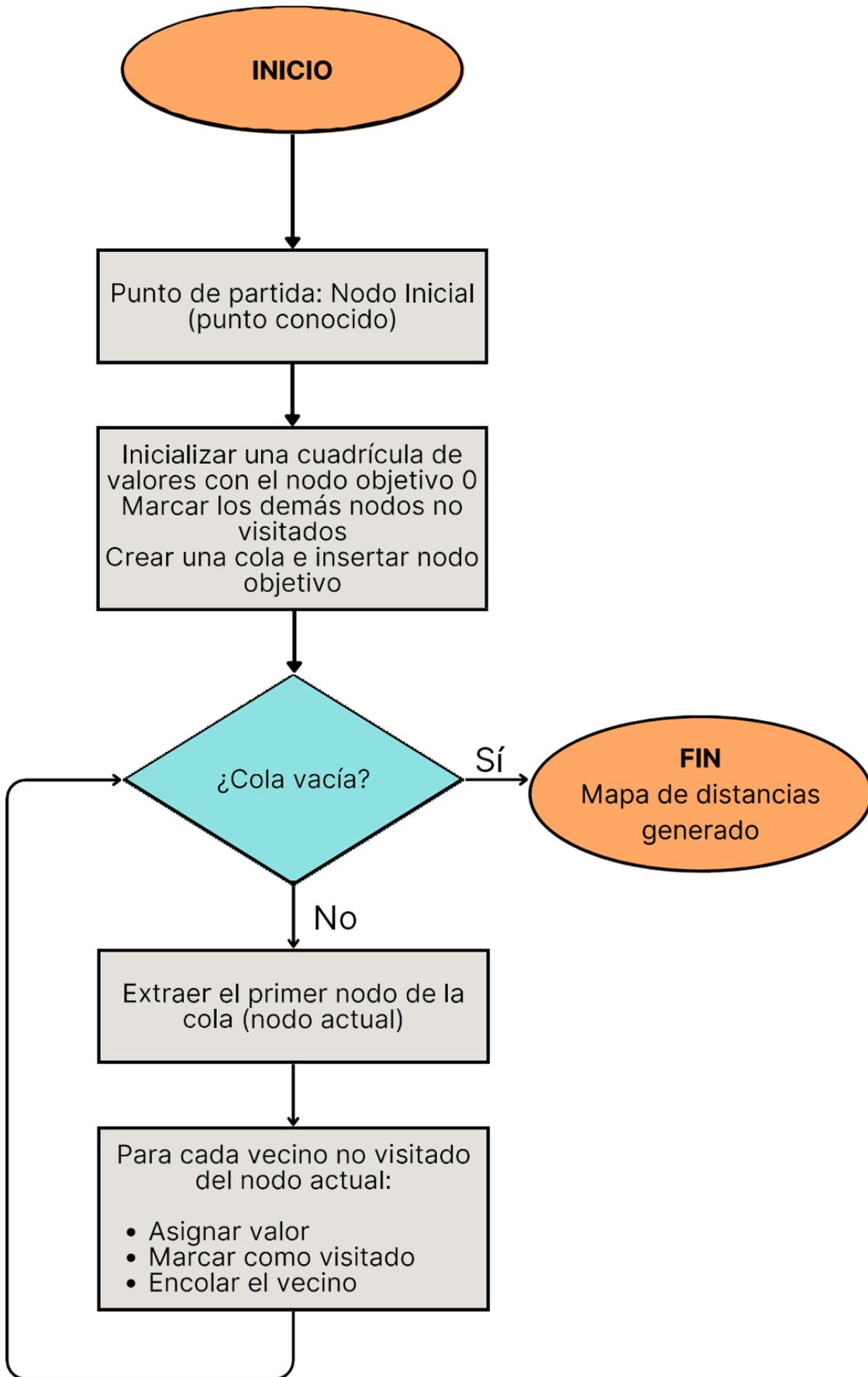


Figura 9. Diagrama de Flujo Algoritmo Wavefront Expansion

4.4.1 Pseudocódigo de Wavefront Expansion

A continuación, se detalla el pseudocódigo del algoritmo Wavefront Expansion.

ALGORITMO WAVEFRONT_EXPANSION(mapa, punto_inicio, punto_objetivo, centro_ZAP, radio_ZAP):

1. SI punto_objetivo NO ES VÁLIDO en mapa:
2. RETORNAR NULO, LISTA_VACÍA, FALSO, NULO

3. CREAR MATRIZ: WAVEFRONT_MAP (dimensiones de mapa, inicializar con INFINITO)

4. CREAR COLA: Q

5. CREAR LISTA: SECUENCIA_EXPANSION_NODOS (para registrar la "onda")

6. ESTABLECER WAVEFRONT_MAP[punto_objetivo.y, punto_objetivo.x] = \emptyset

7. AÑADIR (punto_objetivo, \emptyset) a Q

8. --- FASE 1: INUNDACIÓN (PROPAGACIÓN DE LA ONDA) ---

9. MIENTRAS Q NO ESTÉ VACÍA:
10. (current_y, current_x), current_cost = SACAR_DE_COLA(Q)
11. current_pos = (current_y, current_x)

12. SI current_cost ES IGUAL a WAVEFRONT_MAP[current_y, current_x]:
13. AÑADIR current_pos a SECUENCIA_EXPANSION_NODOS

```

14.     PARA CADA (dy, dx) en DIRECCIONES (arriba, abajo,
izquierda, derecha):

15.     neighbor_y = current_y + dy

16.     neighbor_x = current_x + dx

17.     neighbor_pos = (neighbor_y, neighbor_x)

18.     costo_a_vecino = 1

19.     SI ES_VALIDO(neighbor_pos, mapa) Y

20.         DISTANCIA(centro_ZAP, neighbor_pos) <= radio_ZAP Y

21.         WAVEFRONT_MAP[neighbor_y, neighbor_x] ES INFINITO:

22.         new_cost = current_cost + costo_a_vecino

23.         ESTABLECER WAVEFRONT_MAP[neighbor_y, neighbor_x] =
new_cost

24.         AÑADIR (neighbor_pos, new_cost) a Q

25. --- FASE 2: RECONSTRUCCIÓN DE LA RUTA ---

26.     SI WAVEFRONT_MAP[punto_inicio.y, punto_inicio.x] ES
INFINITO:

27.         RETORNAR NULO, SECUENCIA_EXPANSION_NODOS, FALSO,
WAVEFRONT_MAP

28.     CREAR LISTA: RUTA_OPTIMA

29.     current_path_pos = punto_inicio

30.     AÑADIR current_path_pos a RUTA_OPTIMA

```

```

31. objetivo_encontrado = FALSO

32. MIENTRAS current_path_pos NO ES IGUAL a punto_objetivo:

33.     min_cost = INFINITO

34.     next_pos = NULO

35.     PARA CADA (dy, dx) en DIRECCIONES (arriba, abajo,
izquierda, derecha):

36.         neighbor_y = current_path_pos.y + dy

37.         neighbor_x = current_path_pos.x + dx

38.         neighbor_pos = (neighbor_y, neighbor_x)

39.         SI neighbor_pos ES VÁLIDO en mapa Y

40.             WAVEFRONT_MAP[neighbor_y, neighbor_x] < min_cost:

41.                 min_cost = WAVEFRONT_MAP[neighbor_y, neighbor_x]

42.                 next_pos = neighbor_pos

43.     SI next_pos ES NULO:

44.         ROMPER BUCLE (no se encontró un camino válido)

45.     current_path_pos = next_pos

46.     AÑADIR current_path_pos a RUTA_OPTIMA

47.     SI current_path_pos ES IGUAL a punto_objetivo:

```

```
48.     objetivo_encontrado = VERDADERO

49.     ROMPER BUCLE

50.     RETORNAR     RUTA_OPTIMA,     SECUENCIA_EXPANSION_NODOS,
objetivo_encontrado, WAVEFRONT_MAP
```

4.5 Consideraciones de aplicabilidad en operaciones SAR aéreas

En el contexto de operaciones de búsqueda y rescate (SAR) con aeronaves en mar abierto, la selección del algoritmo adecuado depende en gran medida de las características particulares del entorno y de las capacidades computacionales a bordo. A continuación, se analiza cómo las propiedades intrínsecas de cada algoritmo se traducen en ventajas y limitaciones prácticas para este tipo de misiones.

4.5.1 Búsqueda en Amplitud (BFS)

- **Ventaja:** Es idónea para lograr una cobertura sistemática y exhaustiva del área de búsqueda. Garantiza encontrar el objetivo más cercano en términos del número de movimientos, lo que resulta fundamental en situaciones donde el tiempo de localización es crítico.
- **Limitación:** Requiere una gran cantidad de memoria y tiempo de procesamiento cuando se aplica sobre áreas de búsqueda muy amplias, como el mar abierto. Esto puede suponer un problema si los sistemas embarcados de la aeronave tienen recursos limitados.

4.5.2 Búsqueda en Profundidad (DFS)

- **Ventaja:** Presenta una eficiencia en el uso de memoria, siendo adecuada en escenarios donde se desea explorar rápidamente una dirección concreta, como podría ser el seguimiento de una corriente marina o una estimación lineal de deriva.
- **Limitación:** No garantiza la obtención de la ruta más corta y puede resultar ineficaz en la cobertura del área. Existe el riesgo de que el algoritmo se adentre en zonas alejadas del objetivo y omita otras más relevantes dentro de la zona de búsqueda.

4.5.3 Búsqueda de Coste Uniforma (UCS)

- Ventaja: Permite encontrar rutas óptimas en términos de coste acumulado, lo cual es especialmente útil cuando desplazarse entre diferentes zonas implica costes variables (por ejemplo, consumo de combustible, turbulencias o limitaciones sensoriales). Resulta muy útil cuando estos costes no son homogéneos en el espacio.
- Limitación: Al no emplear heurísticas, puede ser más lento que otros algoritmos como A* en entornos complejos. En casos con costes uniformes, su comportamiento es similar al de la búsqueda en amplitud, pero con mayor carga computacional.

4.5.4 Wavefront Expansion

- Ventaja: Genera un mapa de distancias desde el objetivo hacia todas las celdas, lo que permite no solo encontrar la ruta más corta, sino también facilitar la coordinación entre varias aeronaves SAR. Este tipo de propagación sistemática es especialmente valiosa en estrategias cooperativas.
- Limitación: Tiene un alto consumo de memoria, ya que mantiene una matriz completa de distancias. Además, la fase de propagación de la “onda” puede ser computacionalmente costosa si el objetivo se encuentra lejos del origen o si el área a cubrir es extensa.

Algoritmo	Idoneidad en SAR aéreo	Ventajas clave	Limitaciones Principales
BFS	Alta	Cobertura exhaustiva y sistemática	Requiere mucha memoria y procesamiento
DFS	Media-Baja	Bajo consumo de memoria, útil para trayectorias lineales	Puede ignorar zonas relevantes
UCS	Alta (si hay costes)	Ruta óptima en entornos con costes variables	Lento sin heurísticas, similar a BFS si los costes son fijos
Wavefront Expansion	Alta	Ideal para coordinación de múltiples aeronaves	Alta memoria. Propagación costosa

Tabla 4. Consideraciones Operaciones SAR

4.6 Impacto de la incertidumbre y la ausencia de información precisa

Una de las principales dificultades en las misiones SAR aéreas en mar abierto es la incertidumbre en la localización del objetivo. Generalmente no se dispone de coordenadas exactas, sino de una Zona de Alta Probabilidad (ZAP) basada en estimaciones de deriva, últimos avistamientos, condiciones meteorológicas o trayectorias posibles.

Esta situación afecta directamente a la elección y comportamiento de los algoritmos:

- Imposibilidad de aplicar heurísticas precisas: Al desconocer el destino concreto, no es posible emplear heurísticas efectivas como en A^* , ya que no hay una dirección clara hacia la que orientar la búsqueda. Se requiere, por tanto, una exploración amplia y no dirigida.
- Cobertura y sistematicidad como prioridad: En lugar de optimizar la ruta hacia un objetivo conocido, lo fundamental es maximizar la cobertura de la ZAP de forma eficaz. Algoritmos como la búsqueda en amplitud o la expansión por frentes de onda, que realizan una exploración estructurada y garantizan el hallazgo del objetivo si se encuentra dentro del área, resultan especialmente valiosos en estos escenarios.
- Riesgos con algoritmos menos exhaustivos: La búsqueda en profundidad, al centrarse en una sola rama a la vez y no garantizar cobertura global, puede ser arriesgada. Podría ignorar sectores completos donde se encuentra el objetivo, sobre todo si no se ajusta bien la estrategia de retroceso o reinicio.

En conclusión, en entornos con incertidumbre alta, los algoritmos que priorizan la exploración completa, estructurada y sin depender de una heurística hacia el objetivo son los más adecuados. No basta con encontrar el camino más corto, sino que es vital explorar bien y rápido dentro de una ZAP extensa para maximizar las posibilidades de éxito.

Para visualizar cómo afecta la incertidumbre a la elección de algoritmos en SAR, se propone el siguiente esquema:

4.6.1 Influencia del tipo de incertidumbre sobre la selección del algoritmo SAR

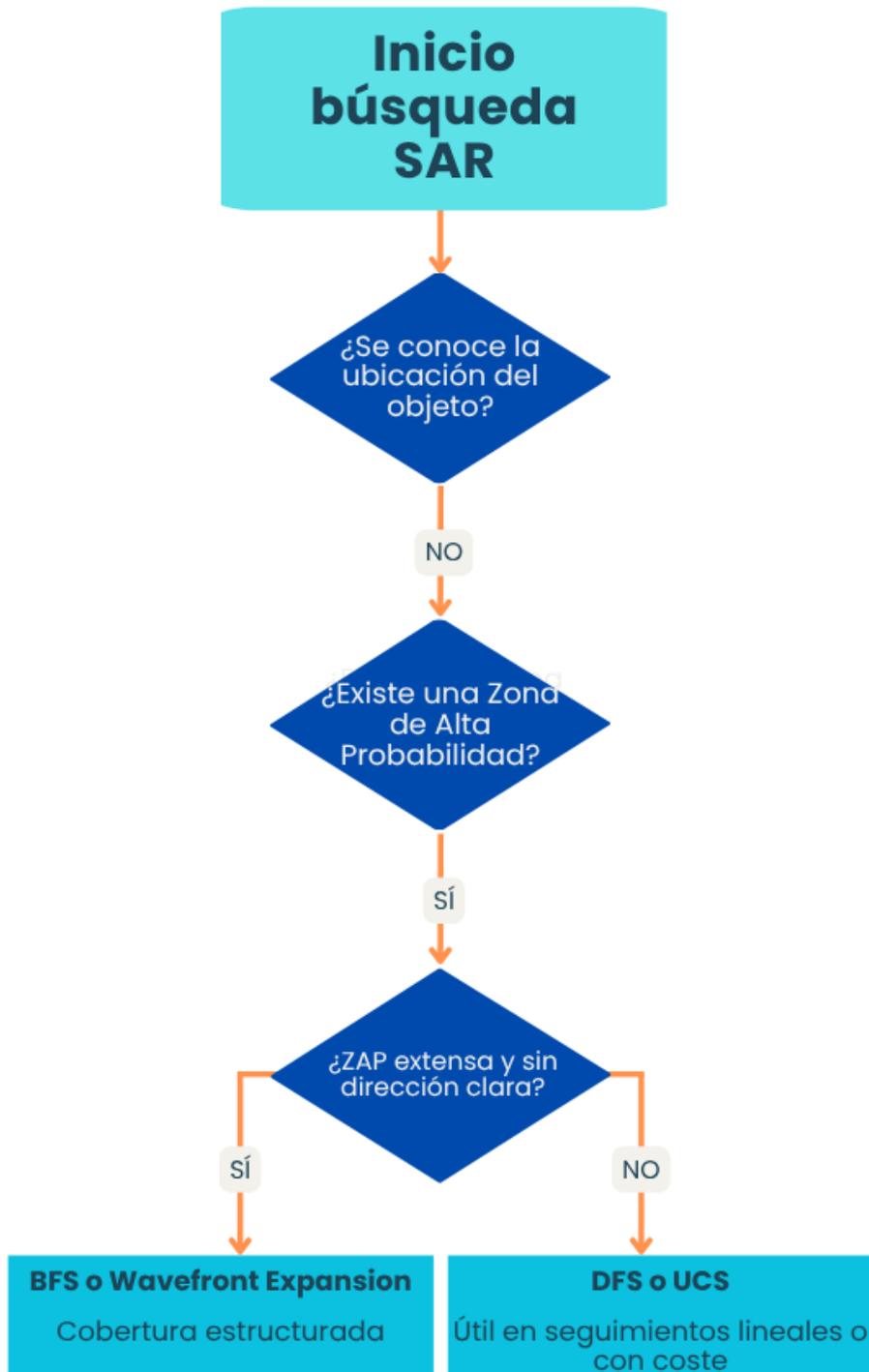


Figura 10. Esquema selección del algoritmo SAR

4.7 Conclusión

Este capítulo ha ofrecido una representación estructurada y visual del funcionamiento interno de los cuatro algoritmos seleccionados para el estudio: Búsqueda en Amplitud (BFS), Búsqueda en Profundidad (DFS), Búsqueda Uniforme de Costo (UCS) y Wavefront Expansion. A través de diagramas de flujo y descripciones detalladas, se ha logrado ilustrar con claridad la lógica operativa que guía a cada uno de estos métodos en la exploración del espacio de búsqueda.

Cada algoritmo presenta fortalezas y limitaciones específicas en relación con su aplicabilidad a operaciones de búsqueda y rescate (SAR) en mar abierto mediante medios aéreos. Mientras que BFS y Wavefront Expansion destacan por su exhaustividad y por garantizar trayectorias óptimas en entornos de coste uniforme, DFS aporta eficiencia en memoria a costa de menor previsibilidad en la calidad de la solución. Por su parte, UCS introduce el concepto de coste acumulado como criterio de expansión, siendo especialmente útil en escenarios con costes variables, aunque en el presente estudio se comporta de forma similar a BFS debido al uso de cuadrículas con costes homogéneos.

Esta similitud entre UCS y BFS en entornos simulados no debe interpretarse como una equivalencia práctica. En situaciones reales de operaciones SAR, los medios aéreos deben considerar factores como consumo de combustible, condiciones meteorológicas, zonas restringidas y tiempo de vuelo, que introducen costes variables en el trayecto. En estos casos, UCS puede adaptarse con mayor precisión a las condiciones operativas reales, seleccionando rutas que no solo sean cortas en distancia, sino también eficientes en términos de recursos y seguridad. Este matiz operativo refuerza la importancia de comprender el entorno de aplicación para seleccionar adecuadamente el algoritmo más eficaz.

Esta fase de diseño conceptual y representación algorítmica constituye un paso esencial antes de abordar la implementación práctica. Conocer la lógica y el comportamiento esperado de cada algoritmo permite una posterior simulación

más controlada, precisa y orientada al análisis comparativo. Asimismo, este conocimiento previo facilita la identificación de posibles limitaciones operativas y de las condiciones bajo las cuales cada algoritmo puede ofrecer un mejor rendimiento.

En los siguientes capítulos se aplicarán estas representaciones al entorno simulado, permitiendo evaluar el desempeño real de cada algoritmo en situaciones similares a las de una operación SAR real, con el objetivo de determinar su eficacia práctica y utilidad operativa.

En el siguiente capítulo se abordará la implementación de estos algoritmos en Python, aplicada a escenarios de búsqueda simulados, lo que permitirá validar y contrastar los comportamientos previstos con datos experimentales y situaciones de incertidumbre realistas.

Capítulo 5. Diseño de escenarios de simulación para evaluación de algoritmos SAR

Este capítulo presenta el diseño y desarrollo de dos escenarios de simulación específicos para la evaluación de algoritmos de búsqueda en operaciones SAR marítimas con aeronaves tripuladas. Dado que las condiciones reales de estas misiones se caracterizan por entornos extensos, obstáculos ocasionales y una alta incertidumbre respecto a la ubicación del objetivo, se han creado dos plantillas diferenciadas que representan situaciones operativamente relevantes: una con expansión de búsqueda desde un punto conocido (Plantilla 1) y otra que incorpora obstáculos estructurales en forma de “T” (Plantilla 2), simulando la complejidad geográfica de zonas marítimas con islas, arrecifes o áreas restringidas. Ambas plantillas utilizan una escala homogénea, un punto de partida fijo (LKP) y objetivos móviles definidos mediante parámetros probabilísticos. El capítulo describe en detalle la lógica de construcción de cada escenario, su correspondencia con situaciones reales y su utilidad para estandarizar las condiciones de prueba. Finalmente, se ofrece una comparativa estructurada entre ambas plantillas, estableciendo su valor estratégico en la validación objetiva de algoritmos SAR antes de su implementación en misiones simuladas.

5.1 Introducción a los escenarios de simulación

Tras haber analizado en profundidad las características y fundamentos teóricos de los algoritmos seleccionados para operaciones SAR marítimas con aeronaves tripuladas, este capítulo se centra en el diseño de dos plantillas o escenarios base que permiten evaluar de forma controlada y repetible el rendimiento de dichos algoritmos en situaciones realistas. La construcción de estas plantillas responde a la necesidad metodológica de comparar los algoritmos bajo condiciones equivalentes, garantizando que cualquier diferencia observada en los resultados se deba a la lógica interna del algoritmo y no a variaciones en el entorno de búsqueda.

Las plantillas se desarrollan como mapas bidimensionales con una escala aproximada en kilómetros, donde se definen un punto inicial fijo (correspondiente al último punto conocido, o Last Known Position, LKP) y un conjunto de posiciones destino variables que simulan posibles ubicaciones del objetivo. En una de las plantillas se ha optado por una disposición completamente abierta, representando un entorno despejado sin obstáculos, mientras que la otra incorpora una estructura en forma de “T” para emular la presencia de accidentes geográficos, como archipiélagos, zonas costeras fragmentadas o áreas restringidas por condiciones meteorológicas o normativas.

Además de facilitar la comparación directa entre algoritmos, estas plantillas permiten analizar cómo influyen las características del entorno en la eficacia de cada método de búsqueda, algo especialmente relevante en contextos SAR donde el tiempo de respuesta y la optimización de la ruta pueden significar la diferencia entre el éxito o el fracaso de la misión. A lo largo del capítulo se describen en detalle las decisiones de diseño tomadas para cada plantilla, su funcionalidad operativa y su aplicabilidad en simulaciones posteriores.

5.2 Plantilla 1: Zona de búsqueda con expansión controlada

Esta primera plantilla ha sido diseñada para representar un escenario de búsqueda marítima en el que la zona de exploración se expande progresivamente desde un punto de referencia inicial, conocido como Last Known Point (LKP). El propósito de esta configuración es simular condiciones realistas que suelen encontrarse en operaciones SAR (Search and Rescue), especialmente aquellas donde se desconoce la ubicación actual del objetivo y se deben aplicar modelos de deriva para predecir su posible posición. Esta expansión controlada sigue un patrón típicamente utilizado en búsqueda de personas u objetos a la deriva, en el que se asume que la incertidumbre espacial aumenta con el tiempo transcurrido desde el último contacto.

5.2.1 Objetivo y lógica del diseño

El mapa simulado tiene unas dimensiones de 50 km de ancho por 20 km de alto, está estructurado como una cuadrícula donde cada celda representa 1 km². Esta estructura discreta permite una modelización precisa de trayectorias y costes para los algoritmos de búsqueda.

A partir del LKP situado en la coordenada (10, 3), se despliega una zona de búsqueda focalizada que se amplía en dirección este (hacia la derecha en el mapa), siguiendo una lógica de expansión por capas temporales. Esta expansión simula la evolución esperada del área de posible ubicación del objetivo a medida que pasan las horas desde el último avistamiento.

La zona azul clara representa el área de búsqueda activa y está libre de obstáculos lo que permite a los algoritmos evaluar el terreno sin restricciones de navegación. Dentro del área de búsqueda se han definido regiones específicas denominadas ZAP (Zonas de Alta Probabilidad). Estas zonas representan áreas restringidas y concretas dentro de la plantilla donde se generan aleatoriamente los objetivos a buscar. Su inclusión tiene un propósito metodológico: evitar la dispersión total del objetivo por toda la plantilla, centrar el análisis en regiones

clave, y observar cómo se comportan los algoritmos frente a entornos parcialmente acotados. Así, se garantiza que las trayectorias propuestas por los algoritmos puedan evaluarse de manera más eficiente y objetiva respecto a si logran o no alcanzar estas zonas. A los lados superior e inferior de la plantilla se han incluido franjas de color marrón que simbolizan zonas excluidas, no porque representen obstáculos físicos, sino por tratarse de áreas descartadas operativamente (por ejemplo, zonas donde la deriva es improbable, áreas ya cubiertas o geográficamente restringidas). Aunque no influyen directamente en el patrón de expansión, estas franjas permiten acotar el área y mantener la búsqueda centrada en un eje preferente.

Este enfoque centrado y dinámico permite evaluar cómo responden los algoritmos cuando deben adaptarse a un escenario de incertidumbre creciente, manteniendo un equilibrio entre cobertura eficaz y eficiencia computacional en un entorno abierto pero condicionado por la evolución temporal.

5.2.2 Representación gráfica y codificación

La codificación y representación visual se ha desarrollado cuidadosamente para simular elementos propios de una misión SAR, tanto operativos como estratégicos:

- **Zona marítima explorable (azul claro):** área libre de obstáculos donde se lleva a cabo la búsqueda.
- **Zona prohibida (marrón):** representan áreas no navegables o excluidas del patrón de búsqueda por decisión táctica u operativa.
- **Last Known Point (rojo):** marcado como un punto circular rojo en la posición (10,3) sirve como origen de la expansión.
- **Líneas de expansión temporal (negras):** representan la expansión del área de búsqueda a intervalos regulares (cada 5 km hacia la derecha), simulando una hora adicional transcurrida desde el último contacto ($t+1h$, $t+2h$, ..., $t+10h$).
- **Zonas de Alta Probabilidad (ZAP) (verde):** áreas circulares verdes distribuidas en la cuadrícula, son el foco principal de la búsqueda en cada simulación.

5.2.3 Funcionalidad específica de las Zonas de Alta Probabilidad (ZAP)

Las Zonas de Alta Probabilidad (ZAP) son elementos claves del diseño y del funcionamiento de las pruebas. En cada simulación, la aeronave se inicia en el LKP y se desplaza directamente hacia el centro de una ZAP específica. Esta maniobra de aproximación permite concentrar el análisis y la evaluación del algoritmo dentro de un espacio delimitado, evitando que el algoritmo tenga que explorar toda la plantilla y optimizando el uso de recursos computacionales y operativos.

Una vez alcanzado el centro de la ZAP, el algoritmo de búsqueda correspondiente comienza a ejecutarse dentro de los límites circulares de esa zona. En otras palabras, la búsqueda está restringida a esa área delimitada, y el objetivo se encuentra ubicado aleatoriamente dentro de ella. Esto simula escenarios SAR reales en los que se cuenta con un área probable estimada mediante modelos de deriva u otras técnicas, y la búsqueda debe ser eficiente dentro de ese sector.

5.2.4 Consideraciones metodológicas importantes

Es relevante aclarar que, aunque la aeronave se desplaza directamente desde el LKP al centro de la ZAP, en ningún caso se permite que el objetivo sea detectado durante ese trayecto, incluso si la trayectoria pasa directamente por encima del objetivo. Esta limitación ha sido introducida de forma deliberada en el diseño de la simulación, con el fin de aislar y analizar exclusivamente el rendimiento del algoritmo de búsqueda dentro del área ZAP una vez comienza su ejecución formal.

Esta decisión, aunque se aleja de una situación operativa real (donde el objetivo podría ser visualizado o detectado en ruta), permite:

- Centrar el análisis en la capacidad del algoritmo para recorrer eficientemente una zona delimitada con un objetivo oculto.
- Eliminar la posibilidad de detección aleatoria o prematura que afecte la medición de rendimiento del algoritmo.

- Garantizar que todas las detecciones se produzcan bajo las mismas condiciones de búsqueda, es decir, dentro del área operativa designada y bajo control del algoritmo.

De este modo, el comportamiento y eficiencia de cada algoritmo SAR puede evaluarse de forma objetiva y reproducible, sin interferencias externas o detecciones fortuitas que desvirtúen los resultados comparativos.

5.2.5 Cobertura, puntos ciegos y costos

En esta plantilla no existen obstáculos ni celdas inaccesibles dentro del área de búsqueda, por lo que no hay puntos ciegos que impidan la cobertura total. El algoritmo debe, idealmente, recorrer toda la ZAP para garantizar la localización del objetivo.

Cada celda dentro de la ZAP tiene un costo unitario uniforme para el desplazamiento, garantizando que la optimización del recorrido se base únicamente en la distancia y no en costes diferenciados. Esto permite evaluar la capacidad del algoritmo para generar trayectorias óptimas o eficientes en tiempo y distancia dentro del área delimitada.

5.2.6 Variabilidad y replicabilidad de la simulación

En cada ejecución, la aeronave inicia en el LKP y se desplaza al centro de una ZAP predefinida.

El objetivo se genera aleatoriamente dentro del área delimitada por esa ZAP.

Se repiten múltiples simulaciones variando la posición del objetivo dentro de la zona para obtener resultados estadísticamente significativos.

Todos los algoritmos se ejecutan bajo estas mismas condiciones para garantizar una comparación equitativa

5.2.7 Ventajas y relevancia de la estrategia ZAP

- Focalización del análisis: Al delimitar la búsqueda dentro de zonas específicas, se evita la dispersión del objetivo en todo el mapa y se facilita una evaluación comparativa más precisa de los algoritmos.

- Simulación realista: Representa la práctica habitual en SAR, donde las áreas de búsqueda se basan en pronósticos probabilísticos (deriva, viento, corrientes) y no se explora la totalidad del área marítima.
- Optimización computacional: Restringir el área de búsqueda a una ZAP reduce el tiempo de cálculo y mejora la eficiencia de la simulación.
- Flexibilidad: Los radios de las ZAP son variables y pueden configurarse para simular diferentes tamaños de áreas prioritarias, lo que permite analizar el comportamiento de los algoritmos ante escenarios más o menos complejos.

5.2.8 Cono de deriva y efecto Coriolis

La plantilla incorpora, de forma conceptual, el cono de deriva, una herramienta habitual en planificación SAR para representar el área dentro de la cual es probable encontrar al objeto o persona desaparecida. Este cono se amplía con el tiempo, siguiendo una geometría orientada por el desplazamiento que inducen el viento, las corrientes marinas y efectos geofísicos.

Entre estos factores destaca el efecto Coriolis, una fuerza aparente producida por la rotación de la Tierra que desvía los movimientos hacia la derecha en el hemisferio norte y hacia la izquierda en el sur. Este fenómeno puede afectar el desplazamiento de objetos flotantes, naufragos, embarcaciones menores o incluso vertidos de combustible, y su influencia se ha considerado teóricamente al diseñar el patrón de expansión. Aunque la plantilla actual muestra una expansión simétrica hacia el este, en condiciones reales esta simetría puede romperse debido al efecto Coriolis, que tiende a desviar los movimientos hacia el noreste en el hemisferio norte. Esta observación permite entender mejor el contexto de uso de la plantilla y su posible evolución futura hacia modelos más realistas de deriva.[25]

5.2.9 Caso real: hundimiento del buque Oleg Naydenov

La importancia de integrar factores como la deriva o el efecto Coriolis se reflejó claramente en el caso del buque pesquero ruso Oleg Naydenov, hundido al sur de Gran Canaria en abril de 2015 tras un incendio. Aunque fue remolcado mar adentro antes de hundirse, el fuel que transportaba emergió días después y se desplazó hacia el suroeste de Gran Canaria y el sureste de Tenerife, afectando áreas costeras de alto valor ecológico. Este movimiento fue el resultado directo de la acción combinada de las corrientes oceánicas locales y del efecto Coriolis, subrayando la necesidad de considerar estos fenómenos en la planificación operativa.[26]

5.2.10 Utilidad de la plantilla

La plantilla garantiza condiciones homogéneas para todos los algoritmos evaluados, partiendo desde un mismo punto inicial y enfrentando un entorno con restricciones simétricas. Este diseño favorece la equidad de las pruebas, la replicabilidad de los resultados y facilita la comparación objetiva entre algoritmos. Además, la claridad visual del diseño y la representación simplificada de la dinámica de deriva hacen que también sea útil para la divulgación de resultados o la formación de equipos operativos no técnicos.

5.2.11 Resultado

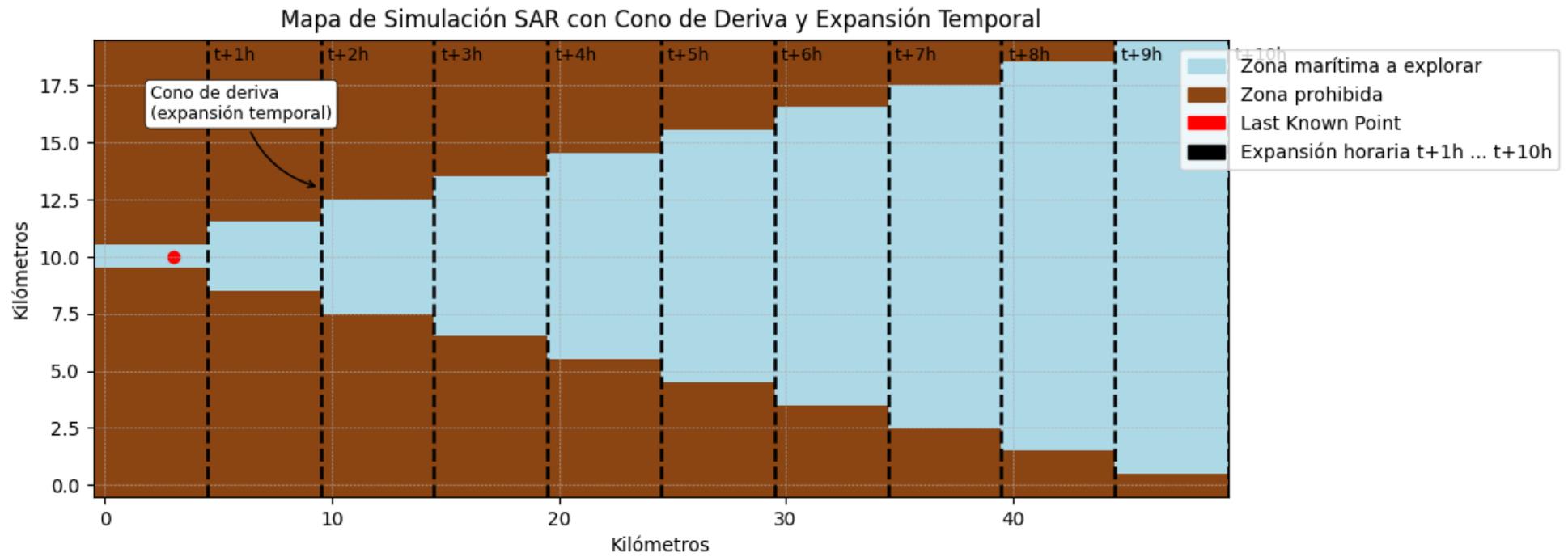


Figura 11. Resultado Plantilla 1

5.3 Plantilla 2: Entorno geográfico simulado en forma de “T”

Esta segunda plantilla de simulación ha sido diseñada con el objetivo de representar entornos fragmentados por grandes obstáculos geográficos, como masas de tierra insular. . A diferencia de la plantilla 1, que simula un entorno abierto y homogéneo, esta configuración en forma de “T” introduce zonas no transitables que condicionan la planificación de trayectorias. La forma de “T” reproduce un corredor marítimo delimitado por dos grandes masas de tierra a izquierda y derecha, reflejando con fidelidad ciertas zonas reales del Atlántico entre islas, como el canal entre Gran Canaria y Tenerife.

Este entorno permite analizar cómo los algoritmos SAR se comportan cuando deben redirigir sus trayectorias debido a obstáculos significativos, y en qué medida logran adaptar su comportamiento ante la fragmentación del espacio operativo.

5.3.1 Objetivo y lógica del diseño

El propósito de esta plantilla es evaluar el rendimiento de los algoritmos en situaciones donde el espacio de búsqueda se encuentra parcialmente bloqueado por grandes masas de tierra. Este tipo de configuración resulta especialmente relevante en operaciones SAR realizadas entre islas, donde las aeronaves deben navegar por canales estrechos, ajustando sus trayectorias para evitar zonas no transitables como montañas, núcleos urbanos, o áreas de exclusión aérea o marítima.

La disposición en forma de “T” busca representar un entorno más fragmentado, donde la navegación directa hacia el objetivo no siempre es posible, obligando a los algoritmos a explorar rutas alternativas y optimizar su comportamiento ante restricciones físicas significativas.

5.3.2 Representación gráfica y codificación

La plantilla se presenta sobre un mapa bidimensional con una codificación visual estandarizada, consistente con la empleada en la plantilla 1, lo que permite una lectura unificada de los entornos:

- **Zona marítima explorable (azul claro):** espacio navegable donde las aeronaves pueden realizar operaciones SAR.
- **Zonas prohibidas (marrón):** dos bloques que simulan islas o masas de tierra no transitables para aeronaves SAR. Están ubicados en los extremos superior e inferior izquierdo del mapa, formando visualmente una “T”.
- **Last Known Point – LKP (rojo):** indica la última posición conocida del objetivo, ubicada en la coordenada (10, 3), marcada con un punto circular rojo.
- **Líneas de expansión temporal (negras):** igual que en la plantilla 1, representan la progresiva expansión de la zona de búsqueda a intervalos de 5 km hacia la derecha ($t+1h$, $t+2h$, ..., $t+10h$), simulando el incremento de incertidumbre a lo largo del tiempo desde el último contacto.

5.3.3 Funcionalidad específica de las Zonas de Alta Probabilidad (ZAP)

En esta plantilla, las Zonas de Alta Probabilidad mantienen su función como área prioritaria de búsqueda. La principal diferencia con respecto a la plantilla 1 radica en su forma geométrica en “T”, que representa un caso donde la probabilidad de presencia del objeto se distribuye en una zona más amplia horizontalmente, manteniendo la progresión temporal hacia la derecha mediante líneas equidistantes. Esto permite observar si los algoritmos siguen patrones coherentes de expansión, y si se ajustan adecuadamente a la estructura del espacio sin necesidad de obstáculos que les fueren desviaciones.

5.3.4 Consideraciones metodológicas importantes

Aunque el área de búsqueda de esta plantilla presenta una configuración distinta en forma de “T”, el enfoque metodológico para evaluar los algoritmos SAR mantiene ciertos principios básicos. En particular, la búsqueda se desarrolla exclusivamente dentro de la ZAP delimitada, evitando que el objetivo pueda ser detectado fuera de este espacio, incluso si la trayectoria de la aeronave lo cruza directamente.

Esta restricción permite centrar la evaluación en el desempeño de los algoritmos frente a una distribución espacial distinta, donde la superficie total explorada se mantiene constante, pero la forma y disposición de la zona introduce nuevos desafíos en la planificación y el recorrido. Así, se asegura que la comparación entre algoritmos sea objetiva y que las diferencias en los resultados reflejen su habilidad para adaptarse a esta nueva configuración, sin interferencias externas ni detecciones anticipadas.

5.3.5 Cobertura, puntos ciegos y costos

En esta plantilla no existen obstáculos ni celdas inaccesibles dentro del entorno general de búsqueda, por lo que la superficie total puede ser recorrida sin restricciones físicas. Sin embargo, al igual que en la plantilla anterior, la evaluación del rendimiento de los algoritmos se centra únicamente en la ZAP, una subregión prioritaria que cambia de posición en cada ejecución de prueba para simular distintas hipótesis de deriva.

Cada celda dentro de la ZAP mantiene un coste unitario y homogéneo de desplazamiento, lo que garantiza que la planificación de trayectorias dependa exclusivamente de la eficiencia espacial del algoritmo, sin influencias externas como celdas penalizadas o diferenciadas por tipo de terreno.

No obstante, la geometría en forma de “T” del entorno influye indirectamente en la distribución del grid, particularmente en las transiciones entre los segmentos verticales y horizontales. Estas transiciones pueden generar pequeñas irregularidades en los márgenes o solapamientos parciales de celdas, según la forma en que se posiciona la ZAP en cada ejecución. Aunque no constituyen puntos ciegos propiamente dichos, estas asimetrías estructurales pueden afectar

sutilmente cómo algunos algoritmos priorizan su avance o distribuyen la búsqueda, sobre todo si siguen patrones de expansión direccional o heurísticas sensibles a la orientación del entorno.

Este tipo de detalles resulta especialmente útil para analizar si los algoritmos son robustos frente a entornos no rectangulares o configuraciones menos simétricas, incluso en ausencia de obstáculos.

5.3.6 Variabilidad y replicabilidad de la simulación

A pesar de que el entorno en esta plantilla representa una geometría diferente respecto a la anterior, la lógica de posicionamiento de las ZAP se mantiene: esta se ubica de forma aleatoria en cada simulación, dentro de los límites accesibles del mapa. Esto permite generar múltiples escenarios con configuraciones distintas sin modificar la superficie total del entorno ni introducir factores aleatorios fuera del control experimental.

Gracias a esta estrategia, se conserva la replicabilidad del experimento, permitiendo que los algoritmos sean evaluados bajo condiciones comparables. Además, la variabilidad posicional de las ZAP simula situaciones realistas de búsqueda en las que las zonas de mayor probabilidad cambian en función del tiempo, las corrientes o la deriva estimada. Esta característica permite observar el comportamiento del algoritmo ante múltiples distribuciones espaciales dentro de un entorno no convencional, como el de esta plantilla en “T”.

5.3.7 Ventajas y relevancia de la estrategia ZAP

- Focalización del análisis: Permite aislar el rendimiento del algoritmo dentro de un área específica de alta prioridad, lo que mejora la comparabilidad entre pruebas y evita interferencias de exploración fuera de dicha zona.
- Adaptación a escenarios reales: Representa una práctica habitual en operaciones SAR, donde se acotan áreas con mayor probabilidad de hallar al objetivo según modelos de deriva, sin necesidad de cubrir todo el entorno.
- Aprovechamiento computacional: Al centrar la búsqueda en zonas más pequeñas, se reduce el tiempo de procesamiento de los

algoritmos y se favorece su evaluación en múltiples escenarios sin necesidad de modificar todo el entorno.

- Exploración de entornos complejos: En esta plantilla, la combinación entre geometría no rectangular y posicionamiento variable de las ZAP permite analizar cómo cada algoritmo adapta su estrategia en áreas de formas más irregulares.

5.3.8 Caso real: caída del helicóptero Super Puma del SAR (2014)

En marzo de 2014, un helicóptero Super Puma del Servicio de Búsqueda y Rescate del Ejército del Aire español cayó al mar durante un ejercicio nocturno a unos 40 kilómetros al suroeste de Gran Canaria. La operación SAR se desarrolló en una zona marítima con gran profundidad y alta complejidad logística, ya que implicó la coordinación de aeronaves, barcos y submarinos. Aunque el accidente ocurrió relativamente cerca de la costa, la configuración del entorno y las condiciones del mar obligaron a redefinir múltiples veces las rutas de búsqueda, esquivando áreas no transitables y adaptándose a zonas restringidas. Este caso ilustra de forma clara la necesidad de contar con algoritmos capaces de gestionar espacios fragmentados, optimizando trayectorias indirectas en situaciones de alta urgencia y limitaciones físicas del entorno.[27]

5.3.9 Utilidad de la plantilla

Esta plantilla ha sido diseñada para evaluar el desempeño de algoritmos de búsqueda en un entorno cuya geometría se aleja de las formas rectangulares tradicionales. Su configuración en “T” introduce un desafío adicional a nivel de planificación de trayectorias, sin necesidad de incluir obstáculos físicos.

Es especialmente útil para analizar cómo los algoritmos:

- Se adaptan a zonas de búsqueda con distribuciones espaciales irregulares.
- Organizan el recorrido en áreas no convexas, donde es necesario cubrir ramificaciones o tramos separados de la zona principal.

- Mantienen su eficiencia en contextos donde la búsqueda no puede resolverse mediante movimientos lineales simples.

A diferencia de la plantilla 1, esta estructura obliga al algoritmo a gestionar rutas con más cambios de dirección y a reajustar su estrategia en función de la forma, no de la presencia de obstáculos.

Además, su representación gráfica estandarizada facilita la interpretación visual de los recorridos y permite realizar pruebas replicables en condiciones controladas, manteniendo la comparabilidad con otros entornos diseñados para el estudio SAR.

5.3.10 Resultado

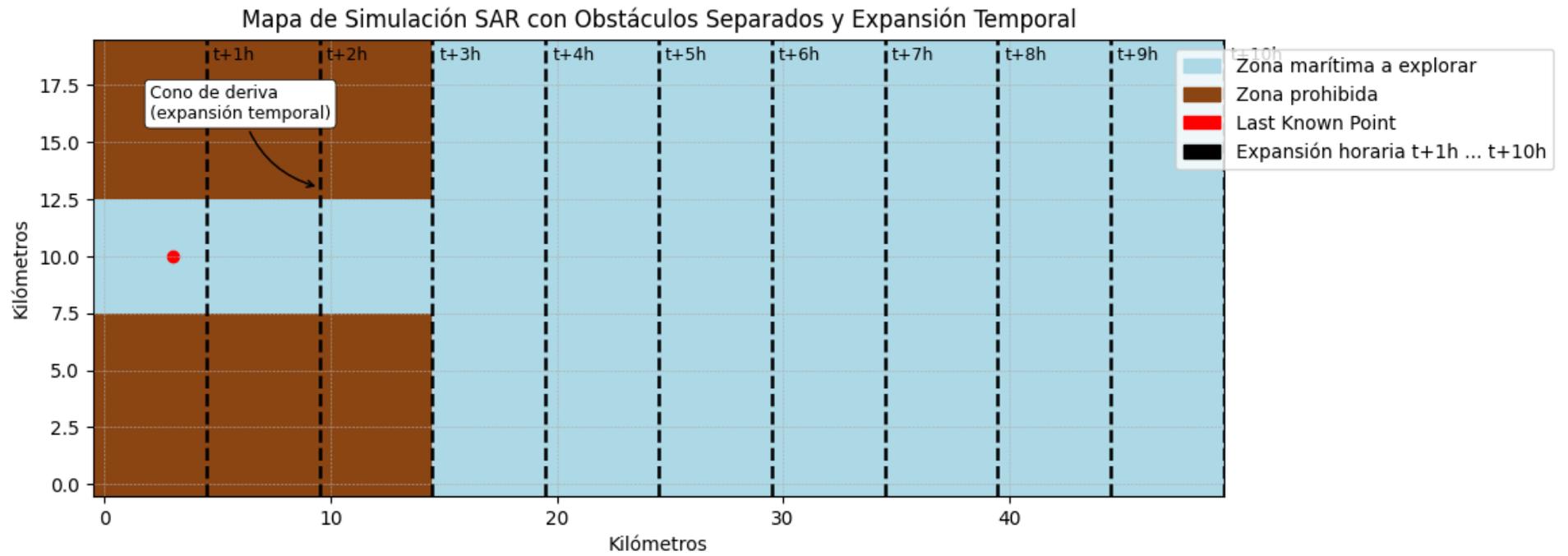


Figura 12. Resultado Plantilla 2

5.4 Consideración sobre el alcance de la búsqueda y la relevancia del diseño

Aunque en todos los escenarios la búsqueda se restringe intencionadamente a zonas de alta probabilidad (ZAP) delimitadas dentro del mapa, el diseño general de cada plantilla sigue teniendo un papel fundamental en la validez y utilidad de las simulaciones.

En primer lugar, la geometría global de la plantilla condiciona tanto la ubicación de las zonas ZAP como la forma en que el algoritmo accede a ellas y organiza su planificación interna. Por ejemplo, en una plantilla con forma de "T", una ZAP situada en el extremo de un brazo obliga al algoritmo a adaptarse a una distribución espacial más compleja, aunque no explore la totalidad del mapa.

En segundo lugar, esta metodología refleja con fidelidad las prácticas operativas reales en misiones SAR, donde no se inspecciona todo el entorno marítimo, sino que se delimitan áreas de búsqueda prioritarias basadas en probabilidades estimadas. En este contexto, la plantilla representa el marco geográfico dentro del cual se ubican y analizan dichas zonas, actuando como soporte espacial realista y controlado.

Así, aunque el área efectivamente explorada por el algoritmo no cubra toda la plantilla, su diseño sigue siendo clave para definir restricciones, trayectorias, accesos y configuraciones que afectan directamente a la planificación. Esto permite mantener la diversidad espacial, la complejidad operativa y el control experimental necesario para una evaluación rigurosa.

5.5 Comparativa de las plantillas de simulación

Aunque ambas plantillas representan entornos simplificados, han sido diseñadas intencionadamente para evaluar aspectos diferentes y complementarios del comportamiento de los algoritmos de planificación de trayectorias en SAR.

- Plantilla 1 reproduce un entorno abierto en el que la zona de búsqueda se amplía de forma progresiva a medida que se avanza desde el punto de partida. La geometría irregular y el ensanchamiento gradual del espacio libre permiten observar cómo cada algoritmo se adapta a trayectorias no lineales y reorganiza su dirección para optimizar el recorrido en espacios amplios. Este escenario es especialmente útil para analizar algoritmos que deben cubrir zonas de deriva extensas, donde la ruta óptima no siempre es evidente desde el inicio.
- Plantilla 2, en contraste, presenta un entorno fragmentado, delimitado por grandes obstáculos intransitables que simulan formaciones geográficas como islas o masas rocosas. Esta configuración reproduce condiciones típicas de escenarios interinsulares o litorales, donde la aeronave debe adaptarse constantemente a restricciones severas. La búsqueda se ve obligada a seguir rutas indirectas, analizar bifurcaciones y responder a configuraciones en las que la solución óptima requiere mayor procesamiento espacial.

Ambas plantillas integran ZAP ubicadas dentro de un entorno más amplio, lo que permite introducir variabilidad en la disposición del espacio de búsqueda sin perder el control sobre las condiciones experimentales. Esta combinación ofrece una visión integral del rendimiento de los algoritmos ante dos grandes retos: la expansión dinámica del área de búsqueda y la navegación en espacios restringidos. En conjunto, permiten identificar fortalezas y limitaciones específicas de cada algoritmo bajo condiciones contrastadas pero representativas.

5.6 Conclusión

En este capítulo se han definido y justificado las dos plantillas base sobre las que se desarrollarán las simulaciones del capítulo siguiente. Cada una ha sido concebida para evaluar dimensiones distintas del rendimiento de los algoritmos SAR: desde su eficiencia en entornos abiertos y dinámicos hasta su capacidad para planificar trayectorias en escenarios restringidos y fragmentados.

A pesar de que la búsqueda se concentra en zonas delimitadas de alta probabilidad, el diseño completo de las plantillas juega un papel esencial, ya que estructura el espacio operativo, condiciona el acceso a las zonas ZAP y refleja la diversidad geográfica presente en misiones reales.

Gracias a esta estrategia de diseño, se garantiza una evaluación metodológicamente robusta, controlada y relevante, que permite analizar tanto la calidad de las trayectorias como la adaptabilidad de los algoritmos en situaciones distintas. Con ello, se establece una base sólida para el análisis cuantitativo y cualitativo del rendimiento computacional y operativo de los algoritmos seleccionados en el capítulo siguiente.

Capítulo 6. Simulaciones y análisis de resultados

En este capítulo se presentan y analizan los resultados obtenidos a partir de las simulaciones realizadas con los cuatro algoritmos seleccionados: búsqueda en amplitud (BFS), búsqueda en profundidad (DFS), búsqueda uniforme de costo (UCS) y Wavefront Expansion. Para ello, se han desarrollado dos plantillas operativas representativas de entornos marítimos reales y, sobre cada una de ellas, se han ejecutado 50 simulaciones por algoritmo, con un total de 400 pruebas. A través de un conjunto de métricas estandarizadas, como el tiempo de ejecución, los nodos explorados, la distancia recorrida y la tasa de éxito en la localización del objetivo, se realiza un análisis detallado del comportamiento de cada algoritmo. Asimismo, se comparan los resultados de forma global para identificar fortalezas, debilidades y posibles aplicaciones reales en misiones SAR aéreas. El capítulo concluye con una reflexión crítica sobre el rendimiento observado, coherente con los objetivos de esta investigación.

6.1 Introducción a las simulaciones

El presente capítulo tiene como finalidad analizar y comparar el rendimiento práctico de cuatro algoritmos de búsqueda aplicados a operaciones de búsqueda y rescate marítimo (SAR) mediante simulaciones sobre escenarios controlados. Los algoritmos evaluados han sido: Búsqueda en amplitud (BFS), Búsqueda en profundidad (DFS), Búsqueda uniforme de costo (UCS) y Expansión por frente de onda (Wavefront Expansion).

La experimentación se ha desarrollado sobre dos plantillas distintas, diseñadas para simular condiciones representativas de un entorno marítimo operativo. Cada plantilla presenta un diseño específico de zonas prohibidas, áreas de exploración potencial y posición inicial del último punto conocido (LKP), integrando un modelo de zona de alta probabilidad (ZAP) a partir del cual se estructura la búsqueda. En cada escenario se han simulado 50 ejecuciones por algoritmo, sumando un total de 400 simulaciones. Esta cantidad de iteraciones garantiza una muestra estadísticamente significativa que permite comparar el rendimiento medio de cada algoritmo y evitar conclusiones sesgadas por simulaciones atípicas.

La simulación de cada ejecución se ha realizado bajo condiciones controladas y homogéneas, permitiendo comparar métricas clave como:

- Tiempo de resolución de la búsqueda (en segundos).
- Número total de nodos visitados, desde el LKP hasta la ZAP y dentro de ella.
- Número de nodos explorados únicos en la ZAP, representando el coste real de exploración.
- Longitud total de la ruta de exploración en kilómetros.
- Tasa de éxito en la localización del objetivo, indicando si el objetivo fue encontrado o no en cada ejecución.

Adicionalmente, en cada simulación se ha calculado y registrado el área total de la zona de alta probabilidad (ZAP) en kilómetros cuadrados (km²). Esta medida no solo permite contextualizar el esfuerzo de exploración dentro de cada ejecución, sino también valorar la eficiencia relativa del algoritmo respecto al tamaño del espacio prioritario. Así, se facilita una interpretación más completa del rendimiento, especialmente en cuanto a cobertura efectiva y coste de exploración en zonas operativas de distintas dimensiones.

En función de estos parámetros, se busca evaluar la eficiencia, cobertura y comportamiento exploratorio de cada algoritmo, considerando tanto sus capacidades técnicas como su aplicabilidad real a operaciones SAR. Además, el análisis tiene en cuenta las características propias de cada plantilla, ya que su estructura y diseño influyen directamente en la estrategia de búsqueda desplegada.

En las siguientes secciones se presentarán los resultados obtenidos por cada algoritmo, diferenciando ambas plantillas. Posteriormente se realizará una comparación cruzada entre ellos, destacando ventajas, limitaciones y posibles aplicaciones dentro de un contexto SAR realista.

6.1.1 Consideración sobre el tiempo de ejecución

En el contexto de las simulaciones, el parámetro de tiempo de ejecución hace referencia al tiempo computacional que cada algoritmo necesita para resolver el escenario de búsqueda, desde el punto de partida (LKP) hasta el hallazgo del objetivo.

Este tiempo no puede extrapolarse de forma directa a una operación SAR real, ya que no contempla variables cruciales como la velocidad de desplazamiento del medio aéreo, condiciones meteorológicas, toma de decisiones humana, retrasos operativos ni comunicación entre unidades SAR.

No obstante, el tiempo de ejecución resulta útil como indicador de la eficiencia algorítmica relativa: permite comparar la rapidez de procesamiento entre algoritmos, lo que en un sistema embarcado o autónomo podría influir en la capacidad de reacción o en la gestión de recursos computacionales en tiempo real.

6.2 Análisis de Resultados en la Plantilla 1

La Plantilla 1 representa un entorno marítimo estructurado como una cuadrícula, en el que se diferencian claramente dos tipos de regiones: las zonas permitidas y las zonas prohibidas. En este contexto:

- La Zona de Alta Probabilidad (ZAP) representa el área donde se estima con mayor certeza la posible ubicación del objetivo. Está definida en la parte noreste del mapa y sirve como principal región a explorar.
- Las zonas prohibidas están distribuidas de forma escalonada, simulando restricciones dinámicas como el efecto Coriolis, patrones de deriva, o zonas de exclusión por condiciones meteorológicas adversas. Estas regiones no representan obstáculos físicos, sino áreas que no pueden ser sobrevoladas, obligando a los algoritmos a trazar rutas que las rodeen o eviten.
- El Last Known Position (LKP) se sitúa en la parte suroeste del mapa y se mantiene constante en todas las simulaciones, mientras que la posición del objetivo varía en cada ejecución, siempre dentro de la ZAP.

Este diseño permite observar de manera clara cómo diferentes algoritmos responden ante un entorno estructurado con restricciones espaciales impuestas por zonas de exclusión, en ausencia de obstáculos físicos o topográficos.

6.2.1 Características observables gracias al diseño de la plantilla

La configuración específica de esta plantilla permite analizar de forma precisa los siguientes aspectos clave:

- Capacidad de adaptación de los algoritmos a restricciones espaciales dinámicas, dado que no existen obstáculos físicos, pero sí zonas claramente delimitadas como permitidas o prohibidas.
- Evaluación de la respuesta ante una ZAP que varía en cada ejecución, tanto en posición como en tamaño, lo que obliga a los algoritmos a reajustar su estrategia de búsqueda en cada simulación. Este detalle añade realismo y complejidad, ya que simula situaciones operativas donde la estimación de la posición del objetivo cambia con el tiempo.
- Efecto de la geometría de la zona prohibida en forma de escalera, que representa fenómenos como el efecto Coriolis, corrientes marinas u otras influencias naturales que limitan el acceso a ciertas zonas. Se observa si los algoritmos eligen bordear por la parte superior, inferior, o si intentan aproximaciones directas que luego deben corregir.
- Diferenciación entre algoritmos que tienden a optimizar la distancia recorrida frente a aquellos que buscan una cobertura más amplia, especialmente cuando la ubicación del objetivo es incierta dentro de la ZAP.
- Consumo de recursos computacionales en presencia de zonas excluidas: se analiza si los algoritmos pierden tiempo y nodos intentando explorar regiones no válidas o si logran concentrar su búsqueda eficazmente dentro del área prioritaria.

Esta plantilla, con sus restricciones espaciales y una ZAP variable, se ha diseñado precisamente para poner a prueba la robustez adaptativa de cada algoritmo, ofreciendo un entorno controlado pero con variabilidad suficiente para obtener resultados comparables y representativos.

6.2.2 Condiciones experimentales

Cada algoritmo se ha ejecutado 50 veces bajo condiciones idénticas, con las siguientes particularidades:

- Punto de inicio (LKP) fijo en la esquina suroeste del mapa.
- Destino variable, aleatorio dentro de los límites de la ZAP.
- Misma disposición de zonas permitidas y prohibidas en cada prueba.
- Se han registrado las siguientes métricas por ejecución:
 - Tiempo de ejecución (segundos).
 - Número de nodos visitados.
 - Número de nodos explorados dentro de la ZAP.
 - Longitud de la ruta generada (kilómetros).
 - Porcentaje de éxito en la localización del objetivo.
 - Área exacta de la ZAP en cada simulación individual, aunque este valor no se ha utilizado para cálculos globales.

Aunque en cada simulación se calcula y muestra el área exacta de la ZAP, este valor no se ha utilizado como media general. Esto se debe a que no siempre se recorre completamente la ZAP en la búsqueda: algunos algoritmos localizan el objetivo antes de explorarla por completo, lo que introduce variabilidad. No obstante, el dato se conserva visible en las simulaciones por si se requiere un análisis adicional o específico sobre cobertura o focalización.

6.2.3 Resultados obtenidos en la Plantilla 1

Algoritmo	Tiempo (s)	Nodos Visitados	Nodos ZAP	Longitud Ruta (km)	% Éxito
BFS	0.0002	67.68	34.98	140.3054	100%
DFS	0.0003	88.54	54.86	66.2862	100%
UCS	0.0003	67.10	34.34	145.0089	100%
WAVEFRONT EXPANSION	0.0010	146.12	115.96	457.6912	100%

Tabla 5. Resultados Plantilla 1

6.2.4 Análisis técnico de los resultados

Búsqueda en Amplitud (BFS)

- Se muestra eficiente en términos computacionales (67.68 nodos visitados) y tiene un tiempo medio muy bajo.
- Su ruta promedio (140.31 km) es razonable, aunque no la óptima.
- La exploración progresiva en capas le permite adaptarse bien a entornos donde no se conoce la dirección del objetivo.
- El diseño del mapa, al no incluir obstáculos pero sí zonas prohibidas, no afecta en exceso a su lógica de expansión ordenada.

Búsqueda en Profundidad (DFS)

- Aunque explora más nodos, especialmente dentro de la ZAP, consigue la ruta más corta (66.29 km).

- Su comportamiento es más arriesgado: profundiza por un camino sin asegurar que es el óptimo.
- En esta plantilla, al no haber obstáculos físicos y tener zonas de exclusión claras, su estrategia le permite encontrar rutas muy eficaces si la rama explorada es favorable.
- En misiones SAR donde se busca eficiencia en la distancia recorrida, DFS puede resultar útil si se combina con criterios de orientación inicial.

Búsqueda de Costo Uniforme (UCS)

- Uno de los algoritmos más equilibrados: pocos nodos explorados y consumo de tiempo contenido.
- Aun así, genera una ruta más larga que DFS y BFS, lo que indica que su priorización por coste no le brinda ventaja en este entorno con pesos uniformes.
- Sería más efectivo en mapas con penalizaciones dinámicas (corrientes, viento, peligros), donde sí se beneficiaría de su lógica de coste acumulado.

Wavefront Expansion

- Alto coste computacional (146.12 nodos visitados, 0.00108 s por ejecución) y una ruta extremadamente larga (457.69 km).
- Su estrategia expansiva global resulta ineficiente en este entorno, donde el objetivo se encuentra en una región específica y no es necesario cubrir toda la cuadrícula.
- Sin embargo, ofrece alta cobertura, lo que podría aprovecharse si la ubicación del objetivo fuera altamente incierta o desconocida.

6.2.5 Conclusiones específicas de la Plantilla 1

- Todos los algoritmos localizaron el objetivo en el 100% de las ejecuciones.
- El entorno diseñado permite aislar y analizar cómo cada algoritmo responde frente a zonas no accesibles sin influencias de obstáculos físicos.
- El diseño escalonado de la zona prohibida simula condiciones reales de deriva o restricciones ambientales, forzando a los algoritmos a tomar decisiones de desvío y planificación de ruta.
- DFS demuestra ser el más eficiente en distancia, a pesar de requerir mayor exploración en nodos.
- UCS es el más equilibrado, aunque en este caso no aporta beneficios claros respecto a BFS.
- Wavefront no es recomendable en entornos donde el objetivo está acotado a una región y se dispone de un punto de partida fijo.

6.2.6 Resultados Gráficos Obtenidos Plantilla 1

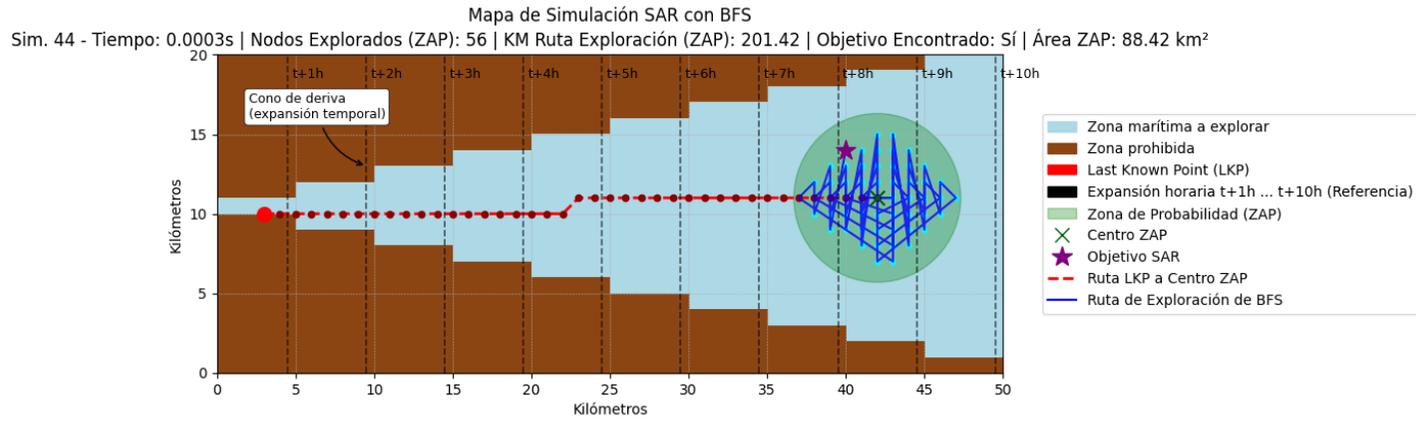


Figura 13. Resultado simulación 44 con BFS Plantilla 1

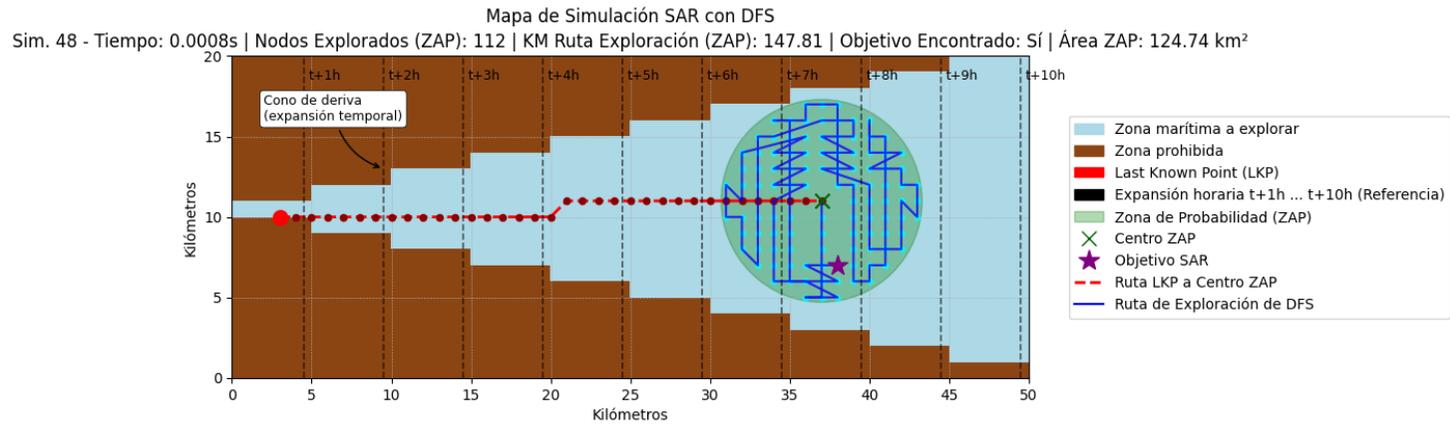


Figura 14. Resultado simulación 48 con DFS Plantilla 1

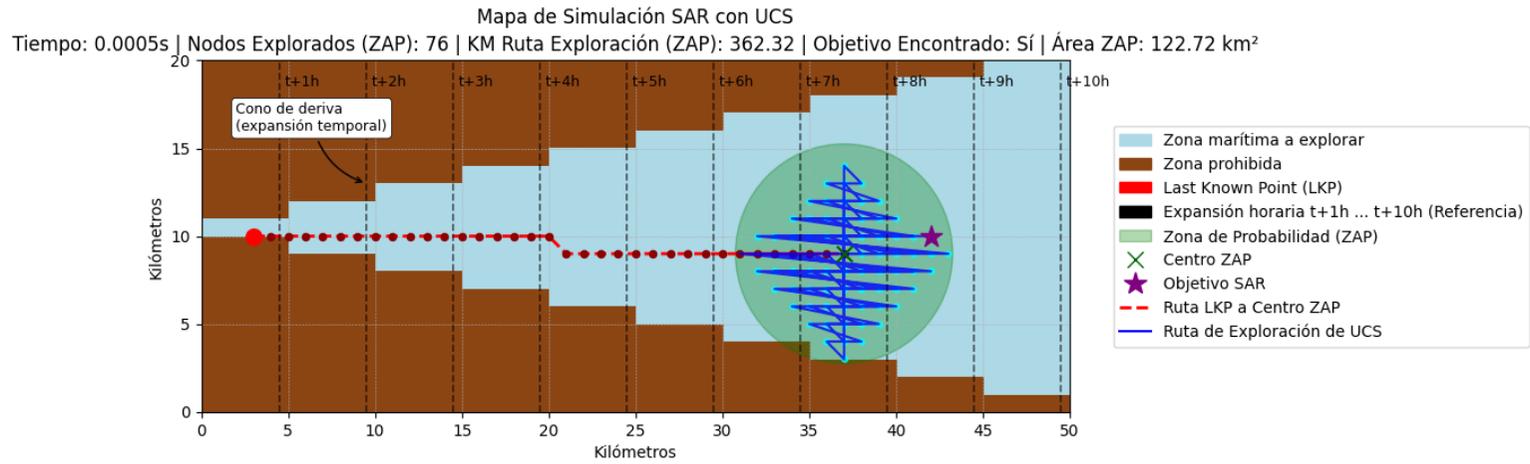


Figura 15. Resultado simulación 11 con UCS Plantilla 1

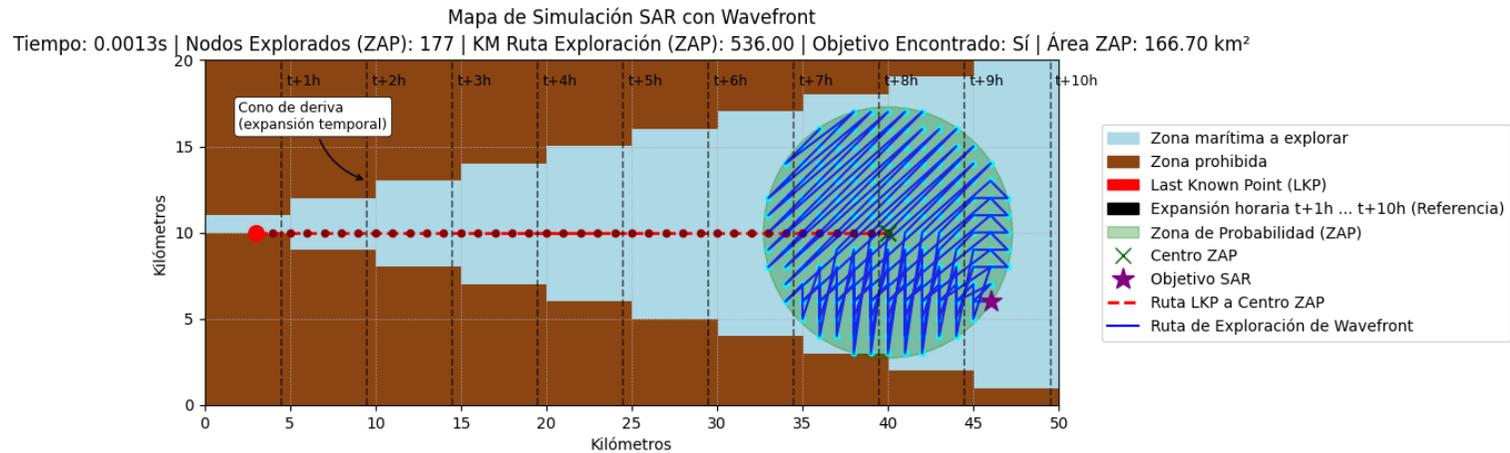


Figura 16. Resultado simulación 38 con Wavefront Expansion Plantilla 1

6.3 Análisis de Resultados en la Plantilla 2

La segunda plantilla presenta un entorno distinto respecto a la anterior. Si bien mantiene el concepto de zonas permitidas y zonas prohibidas (sin obstáculos físicos como tal), su configuración espacial cambia significativamente, especialmente en relación con la estructura de la zona prohibida, que en este caso está más centralizada y presenta una forma más compacta y densa, lo que condiciona directamente las rutas de aproximación hacia la ZAP.

Esta plantilla fue diseñada para:

- Aumentar la dificultad operativa al reducir las rutas válidas desde el punto de partida a la ZAP.
- Obligar a los algoritmos a gestionar mejor los espacios restringidos, especialmente en situaciones donde la ZAP se encuentra parcialmente rodeada por zona prohibida.
- Evaluar la consistencia del rendimiento de los algoritmos frente a un cambio de escenario con características topológicas distintas.

Tal como en la plantilla 1, la ZAP (Zona de Alta Probabilidad) en esta plantilla no es fija: varía tanto en posición como en tamaño en cada ejecución. Esto significa que los algoritmos deben ser suficientemente flexibles y adaptativos para encontrar soluciones eficientes en un espacio de búsqueda cambiante.

Además, durante cada simulación, se extrae el área concreta de la ZAP alcanzada, aunque no se incluye como métrica comparativa global, ya que en muchos casos los algoritmos encuentran el objetivo antes de cubrir la ZAP completa. No obstante, estos valores están disponibles para análisis cualitativos más detallados si se desea explorar el comportamiento específico de cada algoritmo.

6.3.1 Características observables gracias al diseño de la plantilla

Esta segunda plantilla permite estudiar con claridad aspectos técnicos distintos a los de la primera:

- Capacidad de evitar zonas prohibidas más densas y estratégicamente posicionadas, lo cual limita las opciones de ruta válidas y genera cuellos de botella.
- Evaluación del impacto de un entorno más cerrado y menos accesible, obligando a los algoritmos a gestionar mejor el equilibrio entre exploración y eficiencia.
- Mayor sensibilidad al coste de exploración. Al tener caminos más reducidos y dirigidos, los algoritmos que tienden a una búsqueda expansiva pueden verse penalizados por recorrer áreas innecesarias.
- Detección de patrones en rutas preferidas por cada algoritmo, sobre todo al comparar con la plantilla anterior.
- Medición del coste computacional al gestionar entornos menos permisivos espacialmente.

Este entorno, más exigente, es útil para analizar la escalabilidad y resistencia de los algoritmos cuando las condiciones del entorno se vuelven más complejas, simulando misiones SAR en áreas con más restricciones operativas o influencia de factores ambientales que limitan las rutas seguras.

6.3.2 Condiciones experimentales

Las condiciones experimentales aplicadas en esta segunda plantilla han sido idénticas a las empleadas en la plantilla 1, con el fin de garantizar la comparabilidad de los resultados:

- Se han realizado 50 simulaciones independientes por algoritmo.
- En cada ejecución, el punto de inicio (LKP) ha permanecido fijo, mientras que la posición y el tamaño de la ZAP han variado de forma aleatoria, respetando la lógica del diseño de búsqueda.
- La zona prohibida está delimitada en el mapa como espacio no transitable, lo que condiciona las rutas pero no representa obstáculos físicos, sino áreas vetadas de exploración.
- Las métricas observadas en cada simulación fueron:
 - Tiempo de ejecución promedio.
 - Número de nodos visitados desde el LKP hasta la ZAP y dentro de la ZAP.
 - Número de nodos únicos explorados dentro de la ZAP.
 - Longitud del trayecto de búsqueda realizado dentro de la ZAP (en kilómetros).
 - Total de objetivos encontrados.

Adicionalmente, aunque el área concreta de la ZAP explorada se obtiene en cada ejecución, no se ha considerado como una métrica final agregada, ya que en muchos casos los algoritmos localizan el objetivo antes de recorrerla completamente, lo cual distorsionaría cualquier promedio de cobertura real. Sin embargo, este valor se muestra en los resultados individuales para posibles análisis posteriores más detallados.

6.3.3 Resultados obtenidos en la Plantilla 2

Algoritmo	Tiempo (s)	Nodos Visitados	Nodos ZAP	Longitud Ruta (km)	% Éxito
BFS	0.0003	76.56	51.38	247.8176	100%
DFS	0.0008	104.66	79.52	95.0294	100%
UCS	0.0004	74.50	49.20	249.4343	100%
WAVEFRONT EXPANSION	0.0013	180.08	157.64	700.4983	100%

Tabla 6. Resultados Plantilla 2

6.3.4 Análisis técnico de resultados

Búsqueda en Amplitud (BFS)

- Se mantiene como una opción fiable y estable: número reducido de nodos visitados (76.56) y un tiempo de ejecución muy bajo (0.0003 s).
- La ruta promedio en la ZAP fue de 247.82 km, notablemente superior a la registrada en la plantilla 1.
- La estructura del mapa 2, con distribución más amplia de zonas prohibidas, obliga a BFS a desviar sus trayectorias, manteniendo sin embargo su lógica ordenada y sistemática.
- Su rendimiento sigue siendo adecuado cuando se requiere una exploración exhaustiva y simétrica del entorno.

Búsqueda en Profundidad (DFS)

- Aumenta considerablemente la cantidad de nodos explorados (79.52), con una lógica más desordenada.
- Aun así, ofrece la ruta más corta (95.03 km), superando ampliamente a todos los demás algoritmos en eficiencia de desplazamiento.
- Esta eficiencia se debe a la posibilidad de encontrar el objetivo en ramificaciones tempranas, evitando coberturas innecesarias.
- Su comportamiento arriesgado puede ser ventajoso en entornos donde el objetivo está contenido en un área accesible sin ambigüedad.

Búsqueda de Costo Uniforme (UCS)

- Ofrece resultados casi idénticos a BFS en cuanto a nodos visitados (74.50) y explorados (49.20), con tiempos también bajos.
- La distancia recorrida (249.43 km) muestra que, en un entorno con pesos uniformes y sin obstáculos físicos, no obtiene ventajas adicionales respecto a BFS.
- Su lógica basada en el coste podría ser más útil si se introdujeran condiciones dinámicas como penalizaciones, riesgo o viento.

Wavefront Expansion

- El algoritmo más costoso computacionalmente en este entorno: 180.08 nodos visitados y 700.50 km de ruta en ZAP.
- Su estrategia expansiva garantiza una cobertura total, pero a costa de un recorrido innecesariamente extenso y tiempo de ejecución mayor (0.0013 s).
- Aunque no es eficiente, asegura el éxito en búsquedas con alta incertidumbre o cuando se desconoce por completo la posición del objetivo.

6.3.5 Conclusiones específicas de la Plantilla 2

Todos los algoritmos localizaron el objetivo en el 100% de las ejecuciones, confirmando la efectividad general de los métodos en este entorno simulado.

El diseño de esta plantilla incorpora una zona prohibida más amplia y dispersa, obligando a los algoritmos a realizar desvíos adicionales sin contar con obstáculos físicos definidos.

A diferencia de la plantilla anterior, donde las zonas restringidas formaban un patrón escalonado, aquí su disposición exige un mayor esfuerzo de planificación, especialmente para estrategias que intentan cubrir exhaustivamente el entorno.

DFS vuelve a ser el más eficiente en cuanto a distancia recorrida, a pesar de explorar más nodos. Esto lo reafirma como un algoritmo útil en misiones SAR donde se prioriza la rapidez en desplazamiento sobre la cobertura completa.

UCS y BFS se muestran de nuevo prácticamente equivalentes, sin que UCS aporte beneficios adicionales en un entorno sin costes diferenciados.

Wavefront sigue siendo poco recomendable cuando la ubicación del objetivo está acotada y el entorno es limitado en extensión, aunque su alta cobertura podría ser útil en escenarios con incertidumbre extrema o patrones de búsqueda de área amplia.

En cada simulación se obtiene también el área exacta de la ZAP explorada, aunque no se incluye como métrica final ya que en muchas ocasiones el objetivo se encuentra antes de cubrir toda la zona, desviando el promedio y disminuyendo su valor como indicador comparativo directo. Aun así, esta información se conserva por si se desea realizar un análisis más detallado en futuras fases.

6.3.6 Resultados Gráficos Obtenidos Plantilla 2

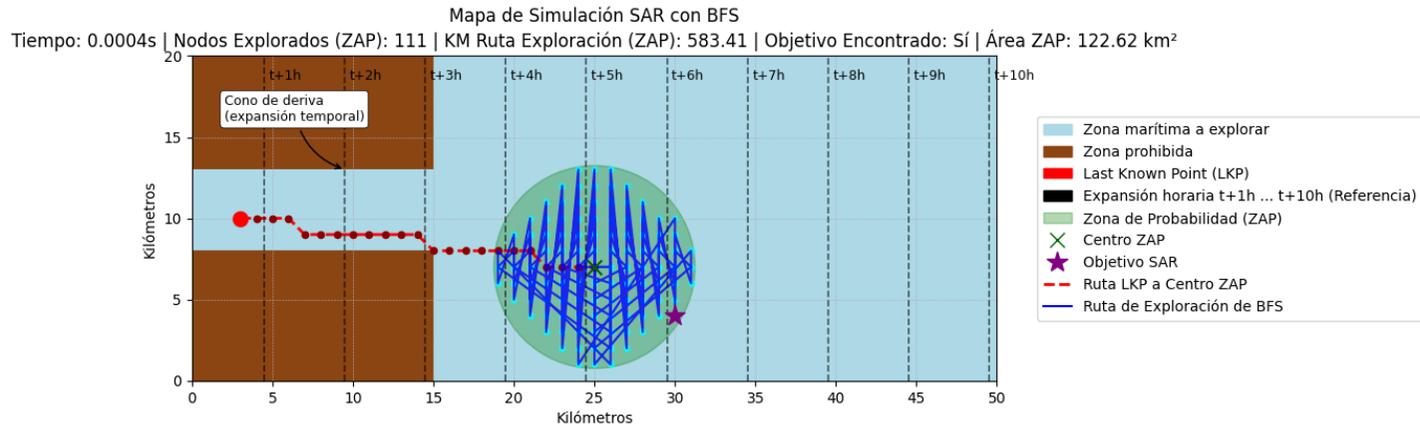


Figura 17. Resultado simulación 43 con BFS Plantilla 2

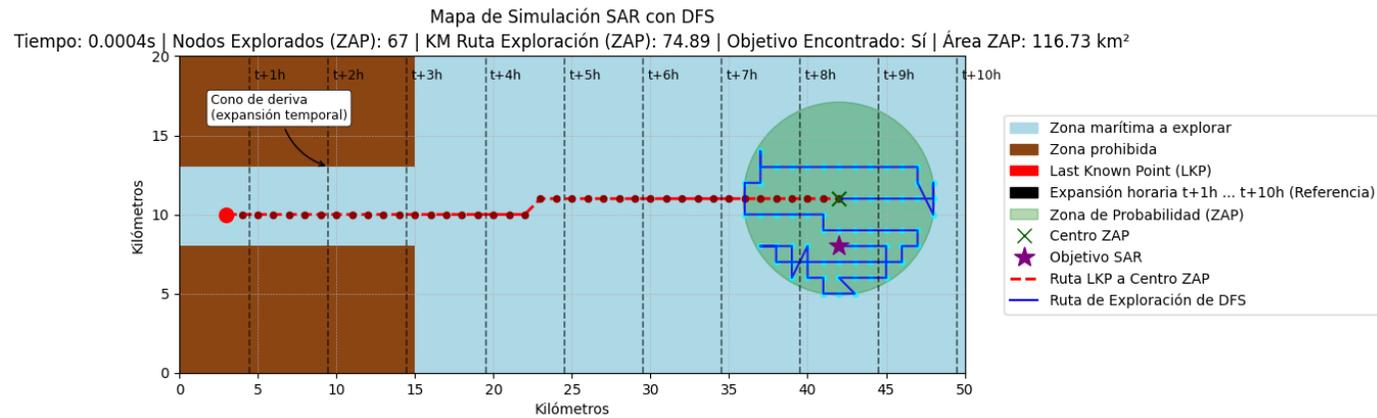


Figura 18. Resultados simulación 28 con DFS Plantilla 2

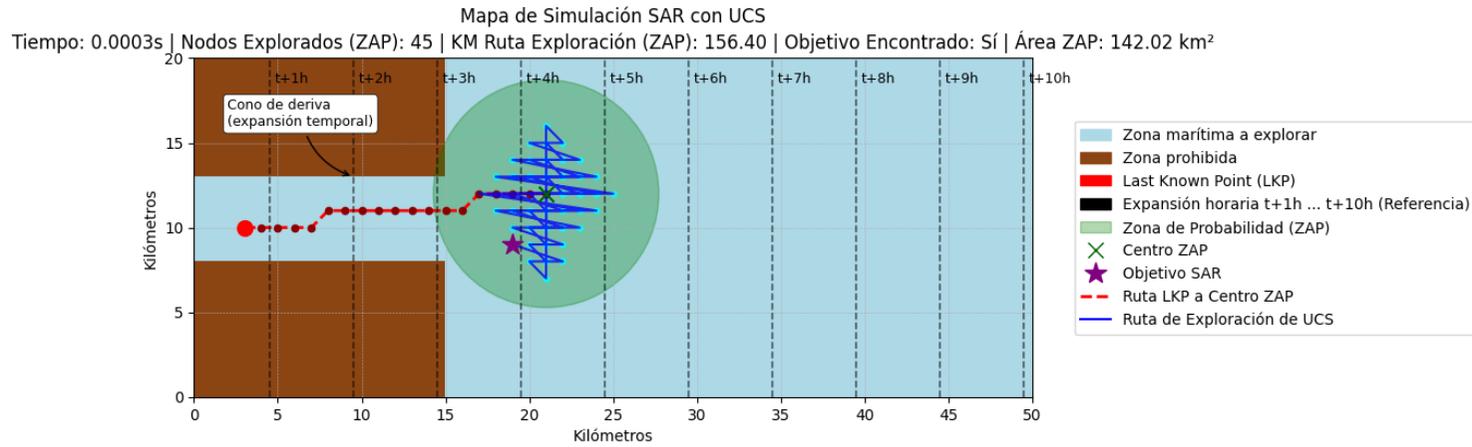


Figura 19. Resultados simulación 26 con UCS Plantilla 2

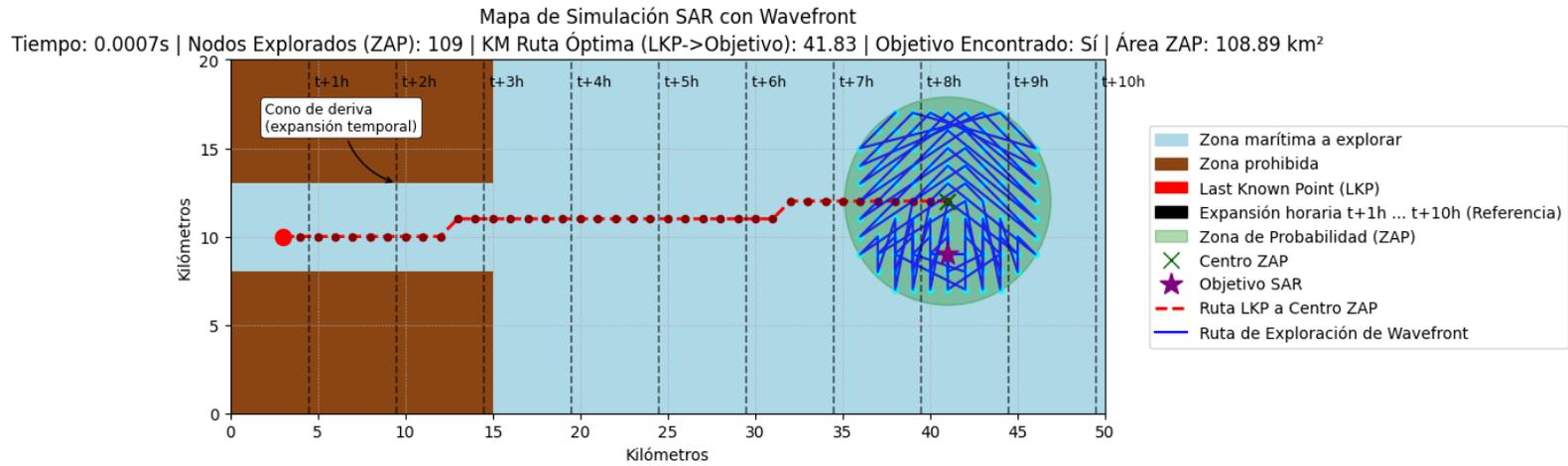


Figura 20. Resultados simulación 12 con Wavefront Expansion Plantilla 2

6.4 Conclusiones generales del análisis experimental

El análisis conjunto de las simulaciones realizadas sobre las dos plantillas diseñadas permite extraer conclusiones globales sobre el rendimiento y comportamiento de cada algoritmo en condiciones variadas, simulando entornos SAR reales. A lo largo de los ensayos, se han medido métricas clave como el número de nodos explorados, la distancia recorrida en zona de búsqueda (ZAP), el tiempo de ejecución y la tasa de éxito en la localización del objetivo. Este apartado organiza las conclusiones más destacadas en distintos ejes de observación.

6.4.1 Rendimiento computacional y eficiencia

En términos de tiempo de ejecución, todos los algoritmos han mostrado rendimientos óptimos en ambas plantillas, resolviendo los escenarios en milisegundos. Sin embargo, se observan diferencias significativas en el número de nodos visitados, una métrica más sensible a la eficiencia interna del algoritmo.

- **BFS** ha sido uno de los más consistentes, con tiempos de ejecución muy bajos (0.0003–0.0004 s) y una cantidad de nodos visitados relativamente contenida, lo que lo convierte en un algoritmo eficaz para misiones con limitaciones de recursos computacionales.
- **DFS**, pese a tener un consumo de tiempo algo superior, se mantiene dentro de márgenes eficientes, aunque con mayor variabilidad en la cantidad de nodos explorados.
- **UCS** muestra una eficiencia parecida a BFS en cuanto a tiempos, pero sin explotar plenamente su lógica de optimización de coste, dado que las simulaciones no aplican penalizaciones o pesos diferenciados.
- **Wavefront Expansion** destaca como el algoritmo con mayor carga computacional, tanto por número de nodos explorados como por el tiempo de ejecución (hasta 0.0013 s). Su naturaleza expansiva completa lo

vuelve poco eficiente en términos computacionales cuando el objetivo se encuentra en una región acotada.

En general, BFS y UCS presentan la mejor relación entre coste computacional y estabilidad de resultados.

6.4.2 Distancia recorrida y optimización de ruta

Uno de los factores clave en SAR aéreo es la distancia de exploración en zona ZAP, ya que está directamente ligada al consumo de combustible, tiempo de misión y fatiga de la tripulación.

- **DFS** destaca como el algoritmo que, en ambas plantillas, ha generado las rutas más cortas en promedio dentro de la ZAP (66.29 km en Plantilla 1, 95.03 km en Plantilla 2). Su búsqueda en profundidad le permite alcanzar rápidamente el objetivo si tiene suerte en la primera rama, lo que reduce significativamente la distancia recorrida.
- **BFS** y **UCS** mantienen distancias medias aceptables (140.31 y 247.82 km respectivamente), lo cual refleja una exploración estructurada sin ser completamente exhaustiva.
- **Wavefront Expansion**, en cambio, genera rutas extremadamente largas (457.69 y hasta 700.49 km), lo que resulta poco realista para operaciones aéreas con recursos limitados.

En misiones donde se busca minimizar la distancia, DFS puede ser muy eficiente, aunque con la contrapartida de mayor exploración de nodos si el objetivo no está en la primera rama.

6.4.3 Cobertura, exploración y garantías

En contextos donde la ubicación del objetivo es incierta y no se puede asumir una dirección concreta desde el punto de inicio, la capacidad de cobertura del algoritmo se vuelve esencial.

- **Wavefront Expansion** garantiza una exploración completa y uniforme del entorno, lo que puede ser positivo en misiones donde se desconoce la zona de alta probabilidad (ZAP) o esta abarca una superficie amplia.

- **BFS**, con su expansión en capas, también ofrece una cobertura ordenada, siendo más balanceado que DFS, aunque menos exhaustivo que Wavefront.
- **DFS**, por su comportamiento determinista, no asegura una cobertura sistemática. En entornos donde se requiere revisar toda la zona, su comportamiento puede dejar regiones sin explorar si se encuentra el objetivo antes.
- **UCS**, en entornos con costes uniformes, se comporta de forma muy parecida a BFS, con una cobertura razonable aunque no completa.

La elección del algoritmo dependerá del nivel de certeza sobre la ubicación del objetivo. Si esta es baja, los algoritmos con cobertura más amplia (Wavefront o BFS) resultan más adecuados.

6.4.4 Sensibilidad al entorno: cómo afecta el diseño del mapa

Las simulaciones se realizaron sobre dos plantillas distintas, sin obstáculos físicos, pero con zonas delimitadas de prohibición y zonas de alta probabilidad (ZAP) que varían de tamaño y posición en cada ejecución.

- En Plantilla 1, con una zona prohibida en forma escalonada, se fuerza a los algoritmos a adaptarse a restricciones geométricas, simulando corrientes o áreas inaccesibles por condiciones externas. Aquí, DFS se adapta muy bien al encontrar rutas eficientes dentro de las limitaciones, mientras que Wavefront se ve penalizado por su incapacidad para priorizar rutas útiles.
- En Plantilla 2, más abierta y simétrica, los algoritmos tienen mayor libertad de movimiento. Aun así, el diseño permite evaluar la eficiencia cuando no existen caminos forzados, y en este entorno BFS y UCS mejoran sus resultados respecto a la anterior.

El comportamiento de los algoritmos está fuertemente influenciado por la topología de las zonas prohibidas y la posición aleatoria de la ZAP, lo que demuestra la importancia de adaptar la estrategia SAR al escenario específico.

6.4.5 Estabilidad de resultados y adaptabilidad

Todos los algoritmos han alcanzado el 100% de localización del objetivo en las 50 ejecuciones realizadas por plantilla, lo que indica un diseño de entorno funcional y un comportamiento robusto en términos de éxito de misión.

Sin embargo, la estabilidad en las métricas varía:

- **BFS y UCS** muestran resultados estables, con pequeñas variaciones entre ejecuciones, lo que facilita la previsibilidad operativa.
- **DFS**, aunque efectivo en distancia, tiene mayor dispersión en el número de nodos explorados.
- **Wavefront**, aunque ofrece consistencia en cobertura, lo hace a costa de consumir más recursos en cada ejecución.

Estos resultados sugieren que BFS y UCS serían más adecuados para entornos operativos donde se requiere estabilidad, mientras que DFS podría utilizarse como estrategia secundaria o en entornos acotados con alta probabilidad de éxito rápido.

6.4.6 Aplicación operativa de los algoritmos SAR

Las diferencias observadas entre los algoritmos no son únicamente técnicas o computacionales; implican consecuencias directas en la planificación y ejecución de misiones reales de búsqueda y rescate. En función del tipo de entorno, los recursos disponibles y el nivel de incertidumbre sobre la localización del objetivo, cada algoritmo puede ofrecer ventajas o presentar limitaciones operativas específicas:

- **BFS (Búsqueda en Amplitud)**

Proporciona una cobertura sistemática de toda la Zona de Alta Probabilidad (ZAP), lo que resulta útil cuando se busca garantizar que ningún área quede sin explorar. Su rendimiento es sólido en entornos donde el objetivo puede estar en cualquier lugar, pero su consumo de recursos puede ser elevado si se necesita minimizar la distancia recorrida o el tiempo total de misión.

- **DFS (Búsqueda en Profundidad)**

Adecuada para localizar objetivos que se encuentren en la dirección inicialmente explorada, ya que prioriza la profundidad antes que la cobertura. Puede ser útil en seguimientos lineales (como derivas conocidas), pero su falta de cobertura uniforme es una desventaja si no se dispone de información inicial fiable.

- **UCS (Búsqueda de Coste Uniforme)**

Equilibra de forma eficiente la cobertura y el coste acumulado de desplazamiento. Su aplicación resulta especialmente útil en entornos abiertos (como zonas marítimas amplias), donde los costes de movimiento son homogéneos. También es una alternativa viable cuando se deben optimizar los recursos sin sacrificar la cobertura.

- **Wavefront Expansion**

Destaca por su capacidad de cobertura máxima, expandiéndose de forma global desde el punto de partida. Es especialmente adecuada en escenarios con alta incertidumbre y prioridad crítica en la localización del objetivo, aunque su demanda de tiempo y recursos puede ser prohibitiva en contextos con limitaciones operativas.

6.5 Comparación con métodos SAR clásicos

Los algoritmos de búsqueda implementados y analizados en este estudio (BFS, DFS, UCS y Wavefront Expansion) ofrecen una visión alternativa a las metodologías clásicas de búsqueda y rescate (SAR), como el cuadrado expansivo, la búsqueda por sectores, la búsqueda a lo largo de la derrota y el barrido paralelo. Estas técnicas tradicionales, ampliamente utilizadas por fuerzas de rescate marítimo y aéreo, han demostrado eficacia operativa en numerosos escenarios reales. Sin embargo, su naturaleza manual, sistemática y basada en reglas fijas limita su adaptabilidad a situaciones altamente dinámicas o con incertidumbre espacial significativa.

A continuación, se presentan las principales diferencias entre ambos enfoques:

6.5.1 Flexibilidad y adaptabilidad

Los métodos clásicos están diseñados para escenarios con hipótesis iniciales claras y están optimizados para tripulaciones humanas, priorizando rutas visuales, cobertura garantizada y seguridad operacional.

En contraste, los algoritmos como DFS o UCS permiten adaptar dinámicamente la trayectoria de búsqueda en función de nuevas prioridades (coste, profundidad, distancia), ajustándose a zonas complejas o restricciones cambiantes.

En situaciones de alta incertidumbre o sin dirección definida, los métodos clásicos pierden eficiencia, mientras que BFS y Wavefront Expansion garantizan cobertura total del área, aunque a mayor coste computacional o de recorrido.

6.5.2 Eficiencia de cobertura

El barrido paralelo y el cuadrado expansivo aseguran una cobertura secuencial o radial del área de búsqueda, pero son sensibles a errores de localización inicial o a condiciones ambientales adversas (corrientes, vientos).

Los algoritmos computacionales garantizan cobertura estructurada sin necesidad de líneas de referencia externas, y pueden adaptarse a patrones de deriva modelados matemáticamente.

Además, pueden ejecutarse miles de veces con ligeras variaciones (Monte Carlo, simulaciones estocásticas) para evaluar escenarios diversos de forma automatizada, lo que es inviable en métodos manuales.

6.5.3 Coste computacional vs operacional

Mientras que los métodos clásicos requieren medios SAR reales (helicópteros, aviones, embarcaciones) para ejecutarse, los algoritmos pueden ejecutarse en tierra antes del despliegue, optimizando las rutas antes del inicio operativo real, lo que reduce tiempo, coste y riesgo para la tripulación.

Incluso algoritmos más exigentes como Wavefront Expansion presentan tiempos de ejecución inferiores a un segundo, lo que permite usarlos en operaciones en tiempo real como apoyo a la toma de decisiones.

6.5.4 Limitaciones operativas

Los métodos clásicos son de más fácil comprensión y planificación para operadores humanos, especialmente en condiciones de estrés y tiempo limitado.

Los algoritmos, aunque precisos, requieren traducción a trayectorias operativas, validación en entornos reales, e interfaces que los integren a sistemas de mando SAR.

Además, algunos algoritmos como DFS pueden ser poco fiables si no se introducen restricciones o filtros de exploración racional, ya que podrían explorar trayectorias subóptimas sin sentido táctico.

6.5.5 Potencial de integración

Ambos enfoques no son necesariamente excluyentes. Un planteamiento mixto puede ser el más adecuado en operaciones reales:

- Utilizar algoritmos computacionales para identificar ZAPs, predecir derivas o seleccionar rutas iniciales óptimas.
- Aplicar métodos clásicos para ejecutar operativamente la búsqueda con personal entrenado, en patrones conocidos y bajo supervisión directa.

6.5.6 Conclusión

El presente capítulo ha permitido evaluar, comparar y contextualizar distintos algoritmos de búsqueda aplicados a operaciones SAR mediante simulaciones controladas. La aplicación sistemática de métricas objetivas ha puesto de relieve tanto las fortalezas como las limitaciones de cada técnica en función del tipo de entorno y objetivo planteado.

Los algoritmos de búsqueda computacional moderna, como BFS, DFS, UCS y Wavefront Expansion, ofrecen herramientas potentes para optimizar la planificación de rutas en condiciones de incertidumbre, con resultados que en muchos casos superan a los métodos clásicos en términos de eficiencia, cobertura o flexibilidad. Sin embargo, su implementación operativa requiere una transición progresiva: es necesario asegurar la compatibilidad con protocolos existentes, interfaces comprensibles por los equipos SAR y una validación empírica mediante datos reales o entornos de entrenamiento simulados con mayor realismo.

La complementariedad entre enfoques clásicos y algoritmos avanzados se perfila como una vía estratégica para mejorar la capacidad de respuesta en misiones críticas. A través de esta convergencia, se abre la posibilidad de diseñar sistemas de búsqueda más adaptativos, robustos y orientados a maximizar la probabilidad de éxito bajo restricciones reales.

El siguiente capítulo explora las principales líneas de investigación que podrían guiar futuras mejoras, tanto en el plano algorítmico como en el diseño del entorno simulado y su aplicación operativa.

6.6 Visualización de los resultados obtenidos en cada una de las simulaciones

Con el fin de complementar el contenido teórico de este Trabajo de Final de Título y proporcionar una herramienta interactiva que permita reproducir y analizar los resultados obtenidos, se ha habilitado un espacio en Google Drive que da acceso directo a un entorno de trabajo en Google Colab.

En dicho entorno se encuentran integrados los códigos desarrollados para la generación de dos plantillas de búsqueda (escenarios), así como las simulaciones ejecutadas con cada uno de los algoritmos seleccionados. Estos códigos han sido implementados en Python y estructurados en cuadernos de Colab, lo que permite su ejecución sin necesidad de instalaciones locales adicionales, directamente desde el navegador.

Cada cuaderno permite ejecutar paso a paso los algoritmos de búsqueda aplicados a las dos plantillas de prueba diseñadas. Al ejecutar las celdas correspondientes, se generan automáticamente las 50 visualizaciones de resultados para cada algoritmo, lo que facilita la observación de su comportamiento, rendimiento y patrones de búsqueda bajo condiciones homogéneas. Esta organización permite comparar con claridad el desempeño de cada enfoque en ambos escenarios definidos.

El propósito de este recurso es ofrecer al lector una herramienta interactiva y accesible que no solo permita verificar los resultados presentados en esta memoria, sino también experimentar con modificaciones en los parámetros para futuras investigaciones o extensiones del trabajo.

Puede accederse al contenido completo y ejecutar los códigos directamente desde el siguiente enlace:

<https://colab.research.google.com/drive/1xYTHSNV-zVnMCWRD39ejE4uqpW4ogrH6?usp=sharing>

Además del entorno interactivo disponible en Google Colab, se ha depositado una copia completa del proyecto en el repositorio científico Zenodo, con el objetivo de facilitar su descarga directa y garantizar la conservación del trabajo desarrollado durante este Trabajo de Final de Título.

El repositorio incluye únicamente los códigos fuente en Python correspondientes a la simulación de los algoritmos en las dos plantillas diseñadas. Estos archivos están organizados en cuadernos en formato .ipynb, compatibles con entornos como Google Colab o Jupyter Notebook, lo que permite su ejecución tanto en línea como de forma local.

Este recurso está especialmente pensado para quienes deseen explorar el funcionamiento interno de los algoritmos, revisar o reutilizar el código, o continuar con trabajos futuros en esta línea.

El repositorio puede consultarse y descargarse a través del siguiente enlace:

<https://zenodo.org/records/16045110?preview=1&token=eyJhbGciOiJIUzUxMiJ9.eyJpZCI6ImQwYWI1ZDY0LTQwN2MtNDAwMS1iZDQ0LTJmZGY2ZDM0YWw0NiIsImRhdGEiOiJ9.4f84Kh7vk6Rvs2E9eHALXTrbdp0Xddst89Q-AhWJizfpKR8sWmdxtElofTZkzNyTckGLfXxKpAsSs15LVmLVjQ>

Capítulo 7. Líneas futuras

En este capítulo se presentan distintas propuestas de mejora y posibles ampliaciones del presente Trabajo de Final de Título, orientadas a servir como punto de partida para futuras investigaciones o aplicaciones prácticas. Estas líneas futuras buscan optimizar las operaciones de búsqueda y rescate (SAR), incrementando la seguridad de la tripulación, mejorando el uso de los recursos disponibles y explorando nuevas funcionalidades que podrían incorporarse al sistema. Todo ello con el propósito de favorecer una evolución tecnológica constante y una mejor adaptación a los escenarios operativos actuales y emergentes.

7.1 Posibles ampliaciones

Una de las líneas futuras más relevantes para continuar el desarrollo de este trabajo se centra en la incorporación de modelos avanzados de deriva de objetos en el mar, utilizando herramientas de inteligencia artificial. En operaciones reales de búsqueda y rescate, es muy poco probable que el objetivo permanezca en el punto de desaparición. Las corrientes marinas, el viento en superficie, las mareas e incluso fenómenos menos predecibles como remolinos o turbulencias oceánicas pueden modificar significativamente la posición del objetivo con el paso del tiempo. Por ello, uno de los siguientes pasos naturales sería implementar algoritmos que puedan predecir la trayectoria de deriva de un naufrago, embarcación o aeronave caída al mar, combinando datos oceanográficos, meteorológicos y de comportamiento del objetivo.

Este tipo de predicciones se puede abordar mediante técnicas como redes neuronales recurrentes (RNN), modelos de predicción bayesiana o aprendizaje supervisado entrenado con datos históricos reales de rescates, simulaciones oceanográficas o registros de deriva. El uso de estos modelos permitiría estimar con mayor precisión el área de incertidumbre, es decir, la región en la que es más probable que se encuentre el objetivo, optimizando así la planificación de las trayectorias de búsqueda.

Otra línea de desarrollo consistiría en entrenar los algoritmos de búsqueda con inteligencia artificial, especialmente mediante aprendizaje por refuerzo o aprendizaje supervisado, de forma que los agentes puedan ir adaptando su comportamiento a medida que aprenden de distintos escenarios. En este trabajo se han utilizado algoritmos clásicos de búsqueda en grafos, principalmente de tipo no informados y deterministas. Estos algoritmos han demostrado ser eficaces en entornos discretos y estructurados como las plantillas utilizadas, pero presentan ciertas limitaciones cuando se enfrentan a condiciones dinámicas o altamente inciertas, lo que abre nuevas posibilidades de mejora mediante técnicas más avanzadas o adaptativas.

También se propone como línea futura la evaluación de nuevos algoritmos o metaheurísticas que no han sido explorados en esta fase del proyecto. Algunos

de ellos pueden incluir algoritmos evolutivos, planificación multiagente, algoritmos híbridos o búsqueda basada en teoría de juegos para escenarios cooperativos entre aeronaves. Estas aproximaciones podrían resultar especialmente útiles en situaciones en las que intervienen varios medios aéreos o marítimos coordinados entre sí, o cuando es necesario tomar decisiones bajo incertidumbre elevada.

Además, se considera relevante mejorar el entorno de simulación, incorporando escenarios dinámicos que contemplen variaciones meteorológicas, cambios en las condiciones del mar o la aparición de nuevos datos durante la misión. Incluir estas variables ayudaría a validar la robustez de los algoritmos propuestos en condiciones más cercanas a la realidad operativa. Incluso podría integrarse un sistema que permita cargar datos reales o semirrealistas desde fuentes abiertas (como Copernicus Marine Environment Monitoring Service) para simular condiciones oceánicas auténticas.

Finalmente, otra posible extensión del trabajo sería la validación experimental de los algoritmos propuestos, ya sea mediante colaboración con organismos de rescate o a través de simuladores profesionales. La inclusión de datos reales permitiría evaluar no solo la eficiencia teórica del algoritmo, sino su impacto operativo y su aplicabilidad práctica. Incluso podría considerarse el desarrollo de una herramienta de apoyo a la toma de decisiones que, integrando los algoritmos desarrollados, proporcione recomendaciones visuales o trayectorias óptimas a los operadores de misión.

Presupuesto

En esta sección se presentarán los costes relacionados con el desarrollo de este Trabajo de Final de Título correspondientes a cada una de las etapas del proceso. El objetivo es ofrecer una visión detallada y precisa de los recursos económicos empleados, facilitando así una comprensión global del presupuesto requerido para llevar a cabo y finalizar el proyecto.

Desglose del Presupuesto

El presupuesto estará dividido en los siguientes apartados:

- Materiales y recursos utilizados.
- Trabajo tarificado por tiempo empleado.
- Material fungible.
- Costes asociados a la elaboración y edición de la documentación
- Tasas correspondientes al visado por parte del COITT (Colegio Oficial de Ingenieros Técnicos de Telecomunicación).
- Gastos de administrativos, de gestión y envío.
- Aplicación de los impuestos pertinentes según la normativa vigente.

Materiales y recursos utilizados

Para la realización del Trabajo de Final de Título (TFT), se han empleado diversos recursos tanto de hardware como de software, los cuales pueden implicar costes asociados en función de su naturaleza, ya sea con o sin licencia. En cuanto a los recursos hardware, se han utilizado los siguientes dispositivos personales: un ordenador portátil, una Tablet y un teléfono móvil. Estos equipos han sido empleados principalmente para la redacción y edición de la memoria, la elaboración de elementos gráficos y el diseño del cartel correspondiente. Asimismo, han sido fundamentales para la realización de videoconferencias y llamadas de seguimiento a lo largo del desarrollo del trabajo.

Respecto a los recursos software, no se han contemplado gastos, ya que se ha recurrido exclusivamente a herramientas en sus versiones gratuitas. Para la redacción de la memoria se ha utilizado el paquete Microsoft Office, cuya licencia ha sido proporcionada por la universidad al tratarse de un estudiante, lo que ha permitido el uso de este recurso sin coste adicional.

Para calcular el coste de amortización de los equipos hardware empleados, se ha considerado su valor de adquisición, así como una estimación del valor residual que conservarán al finalizar su vida útil. En este caso, se ha establecido una vida útil estimada de cinco años para todos los dispositivos utilizados.

Hardware Recurso	Valor de adquisición (€)	Valor residual (€)	Vida útil estimada (meses)	Tiempo usado (meses)	Coste mensual (€)	Importe final (€)
Huawei MateBook D (i7, modelo 2018)	999,00 €	199,00 €	60	6	13,33 €	79,98 €
Xiaomi Poco X3 Pro	269,00 €	69,00 €	60	6	3,33 €	19,98 €
Xiaomi Pad 6S Pro + lápiz	679,00 €	99,00 €	60	6	9,67 €	58,02 €
Conexión a Internet	—	—	—	6	21,00 €	126,00 €
Total						283,98 €

Tabla 7. Materiales y Recursos

El cálculo del coste de amortización se ha realizado considerando el valor de adquisición y el valor residual de cada equipo, aplicando una amortización lineal según la ecuación correspondiente. Este procedimiento se ha aplicado de forma individual a cada recurso hardware utilizado, con el fin de obtener una estimación precisa del coste asociado al desarrollo del Trabajo de Final de Título.

$$\text{Coste de amortización} = \frac{(\text{Valor de adquisición} - \text{Valor residual})}{\text{Años de vida útil}}$$

Ecuación 1. Coste de amortización

La amortización total se detalla en la Tabla, donde se reflejan los valores iniciales, el valor residual, la vida útil estimada y el tiempo real de uso durante el proyecto. En dicha tabla también se incluyen los costes mensuales y el importe final asociado a cada recurso. El importe total resultante asciende a **DOSCIENTOS OCHENTA Y TRES EUROS CON NOVENTA Y OCHO CÉNTIMOS (283,98 €)**.

Trabajo tarificado por tiempo empleado

Este apartado se centra en el cálculo de los gastos de mano de obra, determinados en función del salario por hora correspondiente a un Ingeniero Técnico de Telecomunicaciones. Para ello, nos hemos basado en las directrices y recomendaciones establecidas por el Colegio Oficial de Ingenieros Técnicos de Telecomunicación (COITT), que define el método para calcular el importe a percibir por las horas trabajadas.

$$H = C_t \times 74,88 \times H_n + 96,72 \times H_e$$

Ecuación 2. Trabajo tarificado por tiempo empleado

Donde:

- C_t indica un factor de corrección en función del número de horas trabajadas.
- H_n indica las horas trabajadas dentro de una jornada laboral.
- H_e indica las horas trabajadas fuera de la jornada laboral.

De acuerdo con lo establecido en el proyecto docente de la asignatura de Trabajo de Fin de Título de la Universidad de Las Palmas de Gran Canaria, se obtiene un total de 12 créditos ECTS en la asignatura, lo que resulta en un total de 300 horas. Por tanto, según lo regulado por el COITT, el coeficiente C_t tendrá el siguiente valor.

Horas empleadas H_n	Factor de corrección C_t
180 – 360	0,6

Tabla 8. Factor de corrección

Por tanto, el resultado de la ecuación viene dado por:

$$H = 0,6 \times 74,88 \times 300 + 96,72 \times 0 = 13.478,40 \text{ €}$$

La suma total por los honorarios por tiempo dedicado asciende a **TRECE MIL CUATROCIENTOS SETENTA Y OCHO EUROS CON CUARENTA CÉNTIMOS (13.478,40 €)**.

Material fungible

No se ha utilizado ningún tipo de material fungible durante el desarrollo del presente Trabajo de Final de Título. Todas las tareas se han llevado a cabo utilizando herramientas digitales y recursos electrónicos, sin necesidad de materiales consumibles físicos.

Por lo tanto, el coste asociado a este apartado es de **CERO EUROS (0,00 €)**.

Costes asociados a la elaboración y edición de la documentación

La cantidad asociada a los costes de la documentación del proyecto se calculan según la siguiente ecuación:

$$R = 0,07 \times P \times C_n$$

Ecuación 3. Costes elaboración y edición

Donde:

- R honorarios por la redacción del TFT.
- P indica el total del presupuesto estimado hasta ahora.
- C_n es el coeficiente de ponderación en función del presupuesto.

El valor de P se obtiene sumando los costes asociados al trabajo tarificado por el tiempo empleado y a la amortización del inmovilizado material, que incluye tanto hardware como software.

Concepto	Coste €
Amortización del material	283,98
Trabajo tarificado por tiempo empleado	13.478,40
TOTAL	13.762,38 €

Tabla 9. Costes elaboración y edición

Como el coeficiente de ponderación, C_n , para presupuestos menores de 30.050€ viene definido por el COITT con un valor de 1.0, el coste derivado de la redacción del TFT será igual a:

$$R = 0,07 \times 13.762,38 \times 1 = 963,37€$$

El coste asociado a la elaboración y edición de la documentación del TFT asciende a **NOVECIENTOS SESENTA Y TRES EUROS CON TREINTA Y SIETE CÉNTIMOS (963,37€)**.

Tasas correspondientes al visado por parte del COITT

A continuación, se calcularán los costes asociados al visado del COITT en base a la siguiente ecuación:

$$V = 0,006 \times P \times C_v$$

Ecuación 4. Derechos de visado del COITT

Donde:

- P es el presupuesto actual del TFT.
- C_v es el coeficiente reductor en función del presupuesto del trabajo.

El presupuesto hasta el momento asciende a 14.725,75€. El coeficiente es 1.0, por lo que los costes del visado serán:

$$V = 0,006 \times 14.725,75 = 88,35€$$

Los costes por derechos del visado del presupuesto ascienden a **OCHENTA Y OCHO EUROS CON TREINTA Y CINCO CÉNTIMOS (88,35€)**.

Gastos administrativos, de gestión y envío

Los gastos administrativos asociados a la tramitación del Trabajo de Final de Título incluyen la gestión documental, así como el envío y procesamiento telemático de la documentación final. De acuerdo con los costes estipulados para estos trámites, se establece un importe de **SEIS EUROS (6,00 €)** por documento visado electrónicamente. Este importe cubre los gastos relacionados con la presentación oficial del proyecto ante los organismos correspondientes y la gestión administrativa requerida para su validación y registro.

Aplicación de los impuestos pertinentes según la normativa vigente

En resumen, el gasto global del presente Trabajo de Final de Título asciende a **QUINCE MIL DOSCIENTOS SESENTA Y CINCO EUROS CON CUATRO CÉNTIMOS (15.265,04 €)**. Teniendo en cuenta, el régimen fiscal en vigencia en Canarias se debe incorporar un Impuesto General Indirecto Canario (IGIC) del 7%. Por tanto, el resultado final establece que:

Recurso	Coste €
Materiales y Recursos utilizados	283,98
Trabajo tarificado por tiempo empleado	13.478,40
Material fungible	0
Costes asociados a la elaboración y edición de la documentación	963,37
Tasas correspondientes al visado por parte del COITT	88,35
Gastos administrativos, de gestión y envío	6
Total sin IGIC	14.820,10
Aplicación de impuestos (7%)	1.037,41
TOTAL	15.857,51

Tabla 10. Aplicación de impuestos

El presupuesto total del Trabajo de Final de Título de **Análisis de algoritmos para la generación de trayectorias SAR con IA** asciende a **QUINCE MIL OCHOCIENTOS CINCUENTA Y SIETE EUROS CON CINCUENTA Y UN CÉNTIMOS (15.857,51€)**.

Bibliografía

- [1] HISPAVIACIÓN, AVIACIÓN, “Aniversario del ejército del aire. El águila despliega sus alas”. Accedido el 20 de marzo de 2025. [En línea]. Disponible: <https://www.hispaviacion.es/aniversario-del-ejercito-del-aire-el-aguila-despliega-sus-alas/>
- [2] THE FIGHTER COMMUNITY, “Grupo 82 de Rescate Aéreo (SAR) del Ala 46”. Accedido el 20 de marzo de 2025. [En línea]. Disponible: <https://www.thefightercommunity.com/index.php/2019/07/23/grupo-82-de-rescate-aereo-sar-del-ala-46/>
- [3] DEFENSA, “El ejército del Aire español recibe su primer Super Puma H215”. Accedido el 20 de marzo de 2025. [En línea]. Disponible: <https://www.defensa.com/espana/ejercito-aire-espanol-recibe-primer-super-puma-h215>
- [4] AERONAVES MILITARES ESPAÑOLA, “Base de datos y noticias de aeronaves de nuestras Fuerzas Armadas y Guardia Civil – CN-235”. Accedido el 20 de marzo de 2025. [En línea]. Disponible: <https://aeronavesmilitaresespanolas.com/cn-235-ejercito-del-aire/cn-235-saer/>
- [5] EJÉRCITO DEL AIRE Y DEL ESPACIO, DEFENSA, “Mando Aéreo de Canarias (MACAN)”. Accedido el 20 de marzo de 2025. [En línea]. Disponible: https://ejercitodelaireydelespacio.defensa.gob.es/EA/ejercitodelaire/es/Unidades/Unidad/d5cdb673-9f2c-11ee-b1b1-005056bf91c5/?resourceId=d5cdb673-9f2c-11ee-b1b1-005056bf91c5&path=/sites/internet.es/.content/unidad/unidad_00075.xml#gsc.tab=0

- [6] MANUAL IAMSAR. “Volumen II – Coordinación de las misiones”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://higieneyseguridadlaboralcvs.wordpress.com/wp-content/uploads/2012/09/imasar-2010-vol-2.pdf>
- [7] ITAÉREA, AERONAUTICAL BUSINESS SCHOOL, “La gestión de Operaciones de Búsqueda y Salvamento (SAR)”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://www.itaerea.es/gestion-operaciones-busqueda-salvamento-sar#tipos-de-sar>
- [8] MANUAL IAMSAR, MANUAL INTERNACIONAL DE LOS SERVICIOS AERONÁUTICOS Y MARÍTIMOS DE BÚSQUEDA Y SALVAMENTO. “Sección 3 – Coordinación en el lugar del siniestro - Configuraciones de búsqueda”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://es.scribd.com/document/690353352/IAMSAR-Manual-Vol-III-Medios-Moviles-2022-espanol>
- [9] ACADEMIA-LAB. “Shakey el robot”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: https://academia-lab.com/enciclopedia/shakey-el-robot/?utm_source=chatgpt.com
- [10] ACCU. “The Rover Challenge: Mars Rover Evolution 1970-2020”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://accu-components.com/us/p/145-rover-challenge-19702020>
- [11] ROBOTNIK. “History of robots and robotics”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://robotnik.eu/history-of-robots-and-robotics/>
- [12] ALGORITMOS GENÉTICOS. “Historia de la Computación Evolutiva”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://www.unedourense.es/noticias/noticia/algoritmos-irRfx>
- [13] BAELDUNG. “Q-Learning vs. Deep Q-Learning vs. Deep Q-Network”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://www.baeldung.com/cs/q-learning-vs-deep-q-learning-vs-deep-q-network>

- [14] ANTONIO-RICHAUD. “Algoritmos de optimización para redes neuronales”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://antonio-richaud.com/biblioteca/archivo/Algoritmos-de-optimizacion-para-RN/Algoritmos-de-optimizacion-para-RN.pdf>
- [15] BLUE PRISM. “IA generativa vs. Aprendizaje automático”. Accedido el 21 de marzo de 2025. [En línea]. Disponible: <https://www.blueprism.com/es/resources/blog/generative-ai-vs-machine-learning/>
- [16] CANVA. “Canva para estudiantes”. Accedido el 30 de marzo de 2025. [En línea]. Disponible: https://www.canva.com/es_mx/educacion/estudiantes/
- [17] COLAB-GOOGLE. “Google Colaboratory”. Accedido el 30 de marzo de 2025. [En línea]. Disponible: <https://colab.google/>
- [18] PYTHON. “Python”. Accedido el 30 de marzo de 2025. [En línea]. Disponible: <https://www.python.org/>
- [19] DATACAMP. “Búsqueda en profundidad en Python: Guía con ejemplos”. Accedido el 2 de abril de 2025. [En línea]. Disponible: <https://www.datacamp.com/es/tutorial/breadth-first-search-in-python>
- [20] ASIMOV. “Algoritmo BFS en Python”. Accedido el 2 de abril de 2025. [En línea]. Disponible: <https://asimov.cloud/blog/programacion-5/algoritmo-bfs-en-python-284>
- [21] DATACAMP. “Búsqueda en profundidad en Python: recorrer grafos y árboles”. Accedido el 2 de abril de 2025. [En línea]. Disponible: <https://www.datacamp.com/es/tutorial/depth-first-search-in-python>
- [22] ASIMOV. “Algoritmo DFS en Python”. Accedido el 2 de abril de 2025. [En línea]. Disponible: <https://asimov.cloud/blog/programacion-5/algoritmo-dfs-en-python-283>
- [23] GEEKSFORGEEEKS. “Búsqueda Uniforme de Costo”. Accedido el 2 de abril de 2025. [En línea]. Disponible: <https://www.geeksforgeeks.org/dsa/uniform-cost-search-dijkstra-for-large-graphs/>

- [24] WIKIPEDIA. “Wavefront expansion algorithm”. Accedido el 2 de abril de 2025. [En línea]. Disponible: https://en.wikipedia.org/wiki/Wavefront_expansion_algorithm
- [25] KILERMT. “Deriva Coriolis y Efecto Coriolis”. Accedido el 20 de abril de 2025. [En línea]. Disponible: <https://kilermt.com/balistica-parte-3-deriva-coriolis-y-efecto-coriolis/>
- [26] MINISTERIO DE FOMENTO. “Incendio en el Puerto de la Luz (Las Palmas de Gran Canaria) y posterior hundimiento del pesquero OLEG NAYDENOV, entre el 11 y el 15 de abril de 2015”. Accedido el 25 de abril de 2025. [En línea]. Disponible: https://www.transportes.gob.es/recursos_mfom/ic_01_2016_olegnaydenov_web.pdf
- [27] CANARIAS AHORA. “El helicóptero del SAR que se estrelló en Canarias se quedó a oscuras en el mar por un fallo en un avión de apoyo”. Accedido el 25 de abril de 2025. Disponible: https://www.eldiario.es/canariasahora/sociedad/helicoptero-sar-estrello-canarias-oscuras_1_4327439.html