

Arquitectura de software para
navegación autónoma y
coordinada de enjambres de
drones en labores de lucha contra
incendios forestales y urbanos

por

Ángel Madridano Carrasco

Tesis depositada en cumplimiento parcial de los
requisitos para el grado de Doctor en

Ingeniería Eléctrica, Electrónica y Automática

Universidad Carlos III de Madrid

Directores:

Arturo de la Escalera Hueso
David Martín Gómez

Tutor:

David Martín Gómez

Noviembre 2020

Esta tesis se distribuye bajo licencia “Creative Commons Reconocimiento - No Comercial - Sin Obra Derivada”.



Porque al final

todo llega

Agradecimientos

En primer lugar, agradecer a mis directores académicos, Arturo y David y, a Pablo en Drone Hopper, por acompañarme a lo largo de esta etapa que hoy está más cerca de culminarse. Gracias por la ayuda prestada, por hacerme crecer, en el ámbito profesional y en el personal. Gracias por confiar en mí y permitir que, junto a Drone Hopper, haya podido cerrar una etapa tan importante.

Aunque si algo me ha llevado hoy a escribir estas líneas es formar parte, durante 5 años, de una familia como el LSI, incluida esa nueva generación de *chavalitos*, que han hecho del B16 un lugar donde nunca sentirse solo. El principal culpable de ello es Fernando. Fue él quien me brindó la oportunidad de trabajar con gente tan buena como la que aquí trabaja y ha trabajado, porque echando la vista atrás, no puedo olvidarme de personas como Noelia, Ricardo o Dani con las que ha sido un verdadero placer trabajar y, que junto al resto, han contribuido a que cada día uno se sienta afortunado de formar parte de un grupo humano como este.

Dentro de esta gran familia hay otra persona que merece una mención aparte, no es otro que Abdulla, un amigo más que un compañero de trabajo. Sólo él sabe lo duro que, durante muchos momentos, ha sido esto y, sus palabras, su ayuda y su trabajo han hecho posible que las dudas se despejaron y continuara trabajando por llegar hoy a escribir estas líneas.

Al escribir estas líneas no puedo olvidarme de un grupo de personas que durante esta etapa han sido más que compañeros de trabajo, han sido amigos, aquellas personas con las que compartir las alegrías y penas del día a día, gente como Carlos, María, Irene, Fran o Jorge que me han ayudado y apoyado de manera especial a lo largo de esta dura etapa, la cual Jorge advirtió y yo no hice caso. Tampoco puedo terminar este párrafo sin agradecer a Sergio y Alejandro todo el trabajo de estos últimos meses, por hacer más fácil y llevadera la vuelta a la normalidad y por estar codo con codo en las últimas fases de este trabajo.

Por último, gracias a mi verdadera familia, a unos padres y hermanas que han sido un espejo dónde mirarse, a los que les debo y, sin los cuáles, no habría llegado hoy a escribir este trabajo. Y gracias a ti, Olga, por comprenderme, soportarme y ayudarme en estos duros meses y, sobre todo, por dar sentido a mi vida de la manera en la que lo has hecho.

Contenidos Publicados y Presentados

Algunas ideas, figuras y tablas usadas en esta tesis han aparecido previamente en las siguientes publicaciones:

Contenido Publicado

*Artículos en revista*¹

- Al-Kaff, A.; **Madridano, Á.**; Campos, S.; García, F.; Martín, D.; de la Escalera, A. Emergency Support Unmanned Aerial Vehicle for Forest Fire Surveillance. *Electronics* 2020, 9, 260. DOI:<https://doi.org/10.3390/electronics9020260> (1).

Este artículo ha sido incluido parcialmente en esta tesis, en los capítulos 2 y 3. La inclusión en la tesis de material de esta fuente se especifica en una nota de pie de página de cada capítulo en el que se produce una inclusión. El material de esta fuente incluido en la presente tesis no está señalado con medios y referencias tipográficas.

- **Madridano, Á.**; Al-Kaff, A.; Martín, D.; de la Escalera, A.A. 3D Trajectory Planning Method for UAVs Swarm in Building Emergencies. *Sensors* 2020, 20, 642. <https://doi.org/10.3390/s20030642> (2).

Este artículo ha sido incluido parcialmente en esta tesis, en los capítulos 2 y 3. La inclusión en la tesis de material de esta fuente se especifica en una nota de pie de página de cada capítulo en el que se produce una inclusión. El material de esta fuente incluido en la presente tesis no está señalado con medios y referencias tipográficas.

¹Los autores de los artículos se enumeran por contribución relativa. El nombre del autor de esta tesis está marcado en negrita para describir fácilmente su participación.

Artículos en congresos

- Sosa San Frutos R., Al Kaff A., Hussein A., **Madridano Á.**, Martín D., de la Escalera A. (2020) ROS-Based Architecture for Multiple Unmanned Vehicles (UXVs) Formation. In: Moreno-Díaz R., Pichler F., Quesada-Arencibia A. (eds) Computer Aided Systems Theory - EUROCAST 2019. EUROCAST 2019. Lecture Notes in Computer Science, vol 12013. Springer, Cham. https://doi.org/10.1007/978-3-030-45093-9_2 (3).

Este artículo ha sido incluido parcialmente en esta tesis, en el capítulo 3. La inclusión en la tesis de material de esta fuente se especifica en una nota de pie de página de cada capítulo en el que se produce una inclusión. El material de esta fuente incluido en la presente tesis no está señalado con medios y referencias tipográficas.

- **Madridano, Á.**, Al-Kaff, A., Martin, D., Escalera, A. D. L., & Armingol, J. M. (2019). Método de planificación de trayectorias múltiples para enjambre de UAVs. In XL Jornadas de Automática (pp. 741-748). Universidade da Coruña, Servizo de Publicacións. DOI: <http://hdl.handle.net/2183/23761> (4).

Este artículo ha sido incluido parcialmente en esta tesis, en el capítulo 3. La inclusión en la tesis de material de esta fuente se especifica en una nota de pie de página de cada capítulo en el que se produce una inclusión. El material de esta fuente incluido en la presente tesis no está señalado con medios y referencias tipográficas.

- **Á. Madridano**, A. Al-Kaff, D. M. Gómez and A. de la Escalera, "Multi-Path Planning Method for UAVs Swarm Purposes," 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES), Cairo, Egypt, 2019, pp. 1-6, doi: 10.1109/ICVES.2019.8906381. (5).

Este artículo ha sido incluido parcialmente en esta tesis, en el capítulo 3. La inclusión en la tesis de material de esta fuente se especifica en una nota de pie de página de cada capítulo en el que se produce una inclusión. El material de esta fuente incluido en la presente tesis no está señalado con medios y referencias tipográficas.

- **Madridano, Á.**, Al-Kaff, A., Flores, P., Martín, D., & de la Escalera, A. (2020, September). Obstacle Avoidance Manager for UAVs Swarm. In 2020 International Conference on Unmanned Aircraft Systems (ICUAS) (pp. 815-821). IEEE. DOI: 10.1109/ICUAS48674.2020.9213905 (6).

Este artículo ha sido incluido parcialmente en esta tesis, en el capítulo 4. La inclusión en la tesis de material de esta fuente se especifica en una nota de pie

de página de cada capítulo en el que se produce una inclusión. El material de esta fuente incluido en la presente tesis no está señalado con medios y referencias tipográficas.

Material de terceros

Material NO escrito o co-escrito por el autor de la tesis

- Las figuras 2.10, 2.11, 2.16, 2.18, 2.19, 2.20, 2.21 se han incluido con el permiso del titular de los derechos de autor, el IEEE.
- Las figuras 2.14, 2.22 no requieren un permiso especial para reutilizar todo o parte del artículo publicado por el MDPI, incluyendo figuras y tablas. En el caso de los artículos publicados bajo una licencia Creative Common CC BY de acceso abierto, cualquier parte del artículo puede ser reutilizada sin permiso siempre que el artículo original sea claramente citado. La reutilización de un artículo no implica el respaldo de los autores o de la MDPI.
- La figura 2.15 se ha incluido al concederse permiso sin costo alguno para utilizar el contenido de una tesis de maestría y/o tesis doctoral, con sujeción a las limitaciones de emplear solo un extracto o hasta 3 tablas o figuras ©2016, ©SAGE Publications.

Material escrito o co-escrito por el autor de la tesis

- Las figuras 3.5, 3.6, 3.7 (de (5)), 4.1, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11 (de (6)) se han incluido con el permiso del titular de los derechos de autor, el IEEE.
- Las figuras 2.9, 3.11, 3.20, 3.24, 3.27, 3.28 (de (2)) no requieren un permiso especial para reutilizar todo o parte del artículo publicado por el MDPI, incluyendo figuras y tablas. En el caso de los artículos publicados bajo una licencia Creative Common CC BY de acceso abierto, cualquier parte del artículo puede ser reutilizada sin permiso siempre que el artículo original sea claramente citado. La reutilización de un artículo no implica el respaldo de los autores o de la MDPI.

Otros Méritos de Investigación

Contenido adicional publicado

Artículos en revista

- Madridano, Á., Campos, S., Al-Kaff, A., García, F., Martín, D., & Escalera, A. (2020). Vehículo aéreo no tripulado para vigilancia y monitorización de incendios. *Revista Iberoamericana de Automática e Informática industrial*, 17(3), 254-263. doi:<https://doi.org/10.4995/riai.2020.11806> (7).
- Madridano, Á., Al-Kaff, A., Flores, P., Marín, J.C., Martín, D., & Escalera, A. (2018). SUAFF: Scalable Unmanned Aerial Fire Fighter - Concept and Development Prototype. *Revista Técnica Aeronáutica*, 2, 9-16.

Artículos en congresos

- Al-Kaff, A., Madridano, Á., Radwan, A., Moreno, F.M., Hussein, A., & Escalera, A. (2019). Heterogeneous Multiple Vehicles Cooperation Approach for Smart Roads. In 11th International Micro Air Vehicle Competition and Conference (IMAV2019), IMAV2019-28, 222-227. doi:[imavs.org/papers/2019/41396.pdf](https://doi.org/10.1109/ICUAS48674.2020.9213915) (8)
- A. Astudillo, A. Al-Kaff, Á. Madridano, F. García, D. Martín and A. de la Escalera, "Mono-LSDE: Lightweight Semantic-CNN for Depth Estimation from Monocular Aerial Images*," 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 2020, pp. 807-814, doi: 10.1109/ICUAS48674.2020.9213915 (9).
- M. Rodríguez, A. Al-Kaff, Á. Madridano, D. Martín and A. de la Escalera, "Wilderness Search and Rescue with Heterogeneous Multi-Robot Systems*," 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 2020, pp. 110-116, doi: 10.1109/ICUAS48674.2020.9213974 (10).

Otros méritos de investigación

- Best Student Paper Award en IEEE International Conference on Vehicular Electronics and Safety (ICVES) 2019 por el trabajo "Multi-Path Planning Method for UAVs Swarm Purposes" de Ángel Madridano, Abdulla Al-Kaff, David Martín Gómez y Arturo de la Escalera. El 6 de septiembre de 2019.
- Participación y tercera posición en la competición "Minesweepers: Towards a Landmine-Free World 2018." organizada por IEEE RAS Egypt Chapter and Global Company for Innovation & Entrepreneurship y desarrollado durante el International Conference on Intelligent Robots and Systems (IROS) 2018, en el Madrid Municipal Conference Centre de Madrid, 1-5 de octubre de 2018.

Prefacio

La realización de esta tesis doctoral se enmarca dentro de la convocatoria de ayudas para la realización de doctorados industriales convocados por la Comunidad de Madrid, correspondiente a la Orden 3779/2017 de 17 de octubre de 2017, de la Consejería de Educación, Juventud y Deporte cuyo extracto fue publicado en el BOCM núm. 252, de 23 de octubre de 2017.

De dicha convocatoria surge el acuerdo entre la empresa Drone Hopper S.L. y la Universidad Carlos III de Madrid para la realización del proyecto de doctorado industrial "Cooperación de drones de altas capacidades de extinción contra incendios forestales y urbanos", en el que la Universidad Carlos III de Madrid actúa como entorno académico y Drone Hopper S.L. como entorno empresarial, dando como resultado la consecución de la presente tesis doctoral "Arquitectura de software para navegación autónoma y coordinada de enjambres de drones en labores de lucha contra incendios forestales y urbanos".

Resumen

Los progresos alcanzados dentro del área de los Vehículos Aéreos No Tripulados, conocidos comúnmente como drones, han ocasionado un aumento exponencial de su mercado, gracias, principalmente, al desarrollo e implementación de soluciones tecnológicas innovadoras. La capacidad de este tipo de aeronaves de poder embarcar un gran abanico de sensores provoca que, en la actualidad, se oriente el uso de esta tecnología a un amplio conjunto de aplicaciones y servicios, como son las emergencias y, en concreto, aquellas relacionadas con los incendios, tanto forestales como urbanos.

La aparición y crecimiento de empresas, como Drone Hopper S.L, cuya labor se destina al diseño y fabricación de drones de alta capacidad de carga y autonomía destinados a la lucha contra el fuego han provocado que dichas plataformas aéreas se posicionen como una potente y eficaz herramienta en el campo de las emergencias y la seguridad. Actualmente, la empresa Drone Hopper se encuentra inmersa en el diseño y desarrollo de la plataforma WILD HOPPER, capaz de trasladar hasta 600 litros de carga útil y realizar maniobras eficaces en labores de extinción de incendios, gracias, en gran parte, a su sistema patentado de liberación de líquidos. Junto a este sistema, los drones fabricados por Drone Hopper, presentan la ventaja de poder realizar trabajos durante la noche, complementando los trabajos de los medios aéreos tradicionales y, en conjunto, superando las limitaciones de otras plataformas aéreas no tripuladas, cuyo uso en trabajos relacionados con los incendios se limitan a la monitorización y vigilancia de áreas de interés.

En los últimos años, se ha producido el nacimiento y expansión de los denominados enjambres de drones, o lo que es lo mismo, equipos escalables de varias aeronaves no tripuladas que operan de manera coordinada y que permiten explotar el uso de tecnologías como la desarrollada por la empresa Drone Hopper. Actualmente, la expansión de estos sistemas es fruto del crecimiento en las investigaciones y desarrollos dentro de este campo, ocasionado, principalmente, por las ventajas que presentan los enjambres de drones en términos de robustez, versatilidad y eficacia. La posibilidad de poder desplegar en un mismo área un conjunto de drones que realicen tareas de manera coordinada provoca, en primer lugar, que se disponga de una herramienta robusta contra averías, en la que la pérdida de cualquiera

de los drones intervinientes en la misión no implicaría el fracaso de la misma. Y, en segundo lugar, que se establezca una actuación eficaz ligada a la reducción del tiempo de respuesta y, a la posibilidad de acometer diferentes tareas de manera simultánea. Junto a esto, destacar que el enjambre de drones no está únicamente relacionado al uso de aeronaves no tripuladas con similares características, sino que existe la posibilidad de emplear equipos heterogéneos de drones, o lo que es lo mismo, desplegar sobre un mismo escenario drones con diferentes características, tanto a nivel estructural como de carga de pago, lo que origina que se disponga de una herramienta tecnológica de alta versatilidad. Estas tres características convierten a los enjambres de drones en una herramienta tecnológica de alto valor añadido en trabajos relacionados con la lucha contra el fuego, los cuales se caracterizan por el dinamismo, la adversidad, condiciones extremas y rápidamente cambiantes, en las que el uso de sistemas robustos y versátiles presentan una alta aplicabilidad.

Aunque junto a estas propiedades existe un aspecto, el cual sigue constituyendo un campo de estudio, investigación y desarrollo, como es la navegación autónoma y cooperativa de dichos enjambres, lo que permitiría poder emplear esta tecnología sin supervisión humana en la zona, reduciendo de esta manera el riesgo y exposición de vidas humanas. Por este motivo, a lo largo del presente trabajo se desarrolla e implementa una arquitectura de software multi-capas capaz de permitir la navegación autónoma y coordinada de un enjambre de drones para poder acometer trabajos esenciales en la lucha contra el fuego, tanto en áreas urbanas como forestales. La arquitectura propuesta incluye un conjunto de métodos redundantes y complementarios que permiten establecer diferentes capas de control para permitir la navegación sin supervisión y cooperativa del enjambre.

La primera de las capas consiste en un planificador de trayectorias, basado en información del entorno en 2D y en 3D, que permite dotar a la arquitectura de un método eficiente y escalable que genere como solución un conjunto de trayectorias óptimas y seguras para que cada uno de los drones pueda alcanzar una ubicación determinada en el entorno. Junto a la efectividad y la escalabilidad, el método propuesto se caracteriza por ser altamente configurable, la cual permite la generación de trayectorias en diferentes situaciones, entre las que destaca la posibilidad de establecer una solución que permita al enjambre de drones alcanzar un objetivo en cuestión bajo una formación concreta, de cara a realizar labores de extinción de manera más eficaz.

La segunda de las capas consta de un gestor de colisiones, formado por diferentes desarrollos y algoritmos, que dotan al enjambre de un sistema de detección y evasión de obstáculos, tanto entre drones del enjambre como con obstáculos presentes en el entorno, que garanticen la navegación segura y libre de colisiones de cada uno de los agentes del enjambre.

Por último, la arquitectura de software desarrollada en la presente tesis docto-

ral busca dotar a cada agente del enjambre de un modelo de toma de decisiones inteligente, el cual permita a cada aeronave, de manera autónoma, escoger una secuencia de acciones que le permita alcanzar un objetivo concreto. Este modelo inteligente de toma de decisiones complementa a todos los métodos de la arquitectura propuesta y, permite, de manera redundante establecer un desarrollo adicional que garantice la navegación autónoma del enjambre en entornos dinámicos.

La combinación de estos desarrollos bajo una misma arquitectura provoca el despliegue de una flota de drones capaz de navegar y realizar trabajos de manera autónoma y cooperativa sobre entornos adversos y dinámicos, como es el caso de los incendios. Por tanto, los trabajos y desarrollos de la presente tesis doctoral se centran en crear una herramienta tecnológica de alto valor añadido, a partir del desarrollo de arquitecturas de software embarcadas en un enjambre escalable de drones que, trabajando de manera coordinada, establezca una respuesta rápida, eficiente y robusta al problema de los incendios, tanto forestales como urbanos.

Abstract

The progress achieved in the area of Unmanned Aerial Vehicles, commonly known as drones, has caused an exponential increase in its market, mainly thanks to the development and implementation of innovative technological solutions. The capacity of this type of aircraft to be able to embark on a wide range of sensors means that, at present, the use of this technology is directed at a wide range of applications and services, such as emergencies and, specifically, those related to fires, both forest and urban.

The appearance and growth of companies such as Drone Hopper S.L., whose work is aimed at the design and manufacture of high load capacity and autonomy drones for firefighting, has led to these aerial platforms being positioned as a powerful and effective tool in the field of emergencies and security. Currently, Drone Hopper is immersed in the design and development of the WILD HOPPER platform, capable of carrying up to 600 liters of payload and perform effective maneuvers in fire fighting, thanks largely to its patented system of liquid release. Together with the system, the drones manufactured by Drone Hopper have the advantage of being able to carry out work at night, complementing the work of traditional aerial means and, as a whole, overcoming the limitations of other unmanned aerial platforms, whose use in fire-related work is limited to the monitoring and surveillance of areas of interest.

In recent years, there has been the birth and expansion of the so-called drone swarms, or what is the same, scalable equipment from various unmanned aircraft that operate in a coordinated manner and that allow the use of technologies such as the one developed by the Drone Hopper company. Currently, the expansion of these systems is the result of the growth in research and development within this field, caused mainly by the advantages that drone swarms present in terms of robustness, versatility, and efficiency. The possibility of being able to deploy in the same area a set of drones that carry out tasks in a coordinated way causes, in the first place, that a robust tool against breakdowns is available, in which the loss of any of the drones involved in the mission would not imply the failure of the same one. And, secondly, that an effective action is established, linked to the reduction of the response time and to the possibility of undertaking different tasks simultaneously.

In addition to this, it is important to point out that the swarm of drones is not only related to the use of unmanned aircraft with similar characteristics, but that there is also the possibility of using heterogeneous drone teams, or what is the same, deploying drones with different characteristics on the same stage, both at a structural and payload level, which results in a highly versatile technological tool. These three characteristics make drone swarms a high value-added technological tool in firefighting related work, which is characterized by dynamism, adversity, extreme and rapidly changing conditions, in which the use of robust and versatile systems have high applicability.

Although together with these properties there is an aspect, which continues to be a field of study, research, and development, as is the autonomous and cooperative navigation of these swarms, which would allow the use of this technology without human supervision in the area, thus reducing the risk and exposure of human lives. For this reason, throughout the present work, multi-layer software architecture is developed and implemented that is capable of allowing the autonomous and coordinated navigation of a swarm of drones to be able to undertake essential fire-fighting work, both in urban and forest areas. The proposed architecture includes a set of redundant and complementary methods that allow establishing different control layers to enable unsupervised and cooperative navigation of the swarm.

The first layer consists of a path planner, based on 2D and 3D environmental information, which provides the architecture with an efficient and scalable method that generates a set of optimal and safe paths as a solution so that each of the drones can reach a particular location in the environment. Along with the effectiveness and scalability, the proposed method is characterized by being highly configurable, which allows the generation of trajectories in different situations, among which highlights the possibility of establishing a solution that allows the swarm of drones to reach a target in question under a specific training, to perform extinction work more effectively.

The second of the layers consists of a collision manager, formed by different developments and algorithms, which provide the swarm with a system for detecting and avoiding obstacles, both between drones of the swarm and with obstacles present in the environment, to ensure safe and collision-free navigation of each of the agents of the swarm.

Finally, the software architecture developed in this doctoral thesis seeks to provide each swarm agent with an intelligent decision-making model, which allows each aircraft, autonomously, to choose a sequence of actions that will allow it to reach a specific objective. This intelligent decision-making model complements all the methods of the proposed architecture and, redundantly, allows the establishment of additional development that guarantees the autonomous navigation of the

swarm in dynamic environments.

The combination of these developments under the same architecture results in the deployment of a fleet of drones capable of navigating and working autonomously and cooperatively in adverse and dynamic environments, such as fires. Therefore, the work and developments of this doctoral thesis are focused on creating a technological tool with high added value, from the development of software architectures embedded in a scalable swarm of drones that, working in a coordinated manner, establish a rapid, efficient and robust response to the problem of fires, both forest and urban.

Índice

Agradecimientos	V
Contenidos Publicados y Presentados	VII
Otros Méritos de Investigación	XI
Prefacio	XIII
Resumen	XV
Abstract	XIX
Lista de Acrónimos	XXXVII
1. Introducción	1
1.1. Contexto Político, Social y Económico	2
1.2. Motivación	7
1.2.1. Definición del problema de extinción de incendios	10
1.2.2. Definición de la problemática en enjambres autónomos de drones	12
1.3. Propuesta de la Tesis	14
1.4. Contribuciones y Estructura	16
2. Estado del Arte	19
2.1. Vehículos Aéreos No Tripulados	19
2.2. Metodologías Actuales Aplicadas a la Cooperación en Enjambres . .	26
2.2.1. Aplicaciones en emergencias	30
2.3. Planificación de Trayectorias	33
2.3.1. Métodos basados en descomposición de grafo	35
2.3.2. Métodos basados en muestreo	41

2.3.3.	Métodos basados en modelo matemático	55
2.3.4.	Métodos bio-inspirados	59
2.3.5.	Conclusiones	65
2.4.	Resumen	69
3.	Planificación de Trayectorias para Enjambres de UAVs	71
3.1.	Planificador de Trayectorias Global en 2D	72
3.2.	Planificador de Trayectorias Global en 3D	81
3.2.1.	Planificación de trayectorias para caso etiquetado	90
3.2.2.	Planificación de trayectorias para caso no etiquetado	91
3.2.3.	Planificación de trayectorias para formación de enjambres	92
3.3.	Simulación y Validación del Método de Planificación de Trayectorias	104
3.3.1.	Arquitectura de software	105
3.3.2.	Validación de la planificación de trayectorias en 2D	106
3.3.3.	Validación de la planificación de trayectorias en 3D	112
3.4.	Suavizado de Trayectorias	118
3.4.1.	Suavizado de trayectorias para planificación en 2D	120
3.4.2.	Suavizado de trayectorias para planificación en 3D	123
3.5.	Conclusiones	128
4.	Arquitectura de Navegación Autónoma para Entornos Dinámicos	131
4.1.	Sistema de Evasión de Obstáculos para Enjambres de UAVs.	133
4.1.1.	Sistema de navegación basado en control de velocidad de crucero	134
4.1.2.	Sistema de detección y evasión de colisiones basado en per- cepción del entorno	140
4.2.	Sistema de Navegación Basado en Perímetros Virtuales	149
4.3.	Conclusiones	151
5.	Sistema de Toma de Decisiones Inteligente para Navegación Au- tónoma de UAVs	153
5.1.	Implementación de Métodos Basados en AI para Navegación Autó- noma de UAVs	154
5.1.1.	Aprendizaje por refuerzo profundo	154
5.1.2.	Algoritmo Deep Q-Network	155
5.1.3.	Análisis e implementación de algoritmo de DQN para nave- gación autónoma de UAVs	156
5.1.4.	Modelo 1 de toma de decisiones inteligente	157
5.1.5.	Modelo 2 de toma de decisiones inteligente	162

5.1.6. Modelo 3 de toma de decisiones inteligente	167
5.1.7. Modelo 4 de toma de decisiones inteligente	171
5.2. Conclusiones	174
6. Conclusiones y Trabajos Futuros	177
6.1. Conclusiones	177
6.2. Trabajos Futuros	182
A. Fundamentos del Aprendizaje por Refuerzo	185
B. Arquitectura de Hardware	193
B.1. Chasis	194
B.2. Autopiloto	196
B.2.1. Módulo de posicionamiento	198
B.3. Módulo de Procesamiento	199
B.3.1. Módulo de comunicaciones	201
B.4. Módulo de Alimentación y Potencia	202
B.5. Conclusiones	205
Bibliografía	207

Índice de Figuras

1.1. Ejes productivos del mercado de UAVs.	5
1.2. Resumen de actividades desarrolladas por los operadores (11).	7
1.3. Esquema general del enjambre de UAVs autónomo propuesto.	15
2.1. Clasificación de UAVs según principio de vuelo.	20
2.2. Tipos de multirrotores y configuraciones (12).	22
2.3. Plataforma WILD HOPPER (13).	25
2.4. Ejemplos de MRS homogéneos.	29
2.5. Ejemplos de MRS heterogéneos.	29
2.6. Diagrama de métodos de planificación de trayectorias.	36
2.7. Ejemplo de grafo.	36
2.8. Ejemplo de caso fallido en métodos de muestreo aleatorio (14).	42
2.9. Algoritmo de PRM aplicado a un edificio de viviendas (2).	45
2.10. Ejemplo de diagrama de Vornoi (15) ©2017 IEEE.	46
2.11. Solución al problema de planificación de rutas para múltiples UAVs a través de un diagrama de Voronoi (15) ©2017 IEEE.	47
2.12. Muestreo del entorno con algoritmo de RRT (14).	49
2.13. Ruta final obtenida por RRT (14).	50
2.14. Generación de RRT en 3D y obtención de una ruta final sobre una nube de puntos (16).	51
2.15. Ejemplo de exploración con algoritmo de dRRT (17) ©2016, ©SAGE Publications.	52
2.16. Ejemplo de APF (18) ©2017 IEEE.	53
2.17. Ejemplo de gradiente para navegación de los agentes (19).	54
2.18. Resultados obtenidos empleando APFs para la planificación de rutas en entornos urbanos (20) ©2017 IEEE.	55
2.19. Solución a la planificación de rutas empleando algoritmos APFs me- jorados (IAPF) (21) ©2019 IEEE.	56

2.20. Planificación de rutas cooperativa basada en MIQP (22) ©2018 IEEE.	58
2.21. Planificación de rutas basada en PSO (23) ©2020 IEEE.	62
2.22. Esquema de método combinado de redes neuronales convolucionales y Deep Q-Learning (24).	64
2.23. Diagrama método de planificación para enjambre de UAVs.	66
3.1. Representación 2D del entorno de la Escuela Politécnica Superior de Leganés.	73
3.2. Solución obtenida por el planificador de trayectorias en 2D propuesto, actuando en un entorno abierto.	74
3.3. Diagrama de flujo para planificación de trayectorias en 2D.	75
3.4. Ejemplo de imagen binaria del entorno en 2D.	76
3.5. Representación del entorno con extensión de obstáculos (5) ©2019 IEEE.	76
3.6. Mapa explorado con un algoritmo de PRM de 100 nodos (5) ©2019 IEEE.	80
3.7. Solución óptima final para un enjambre de 5 UAVs (5) ©2019 IEEE.	81
3.8. Diagrama de flujo para planificación de trayectorias en 3D.	82
3.9. Diagrama del proceso de obtención de un mapa de ocupación 3D.	85
3.10. Salida generada por algoritmo de exploración de entornos en 3D.	90
3.11. Método Húngaro	92
3.12. Tipos de formaciones propuestas para vuelo en enjambre.	94
3.13. Ejemplo de formación en línea horizontal.	96
3.14. Orden de asignación de posiciones en la formación en cuadrado.	97
3.15. Ejemplo de formación en cuadrado.	100
3.16. Ejemplo de formación en cuadrado.	102
3.17. Esquema de actuación en formación cuadrada frente a un incendio con forma circular.	103
3.18. Esquema de actuación en formación tipo flecha frente a un incendio con forma de elipse.	104
3.19. Esquema de actuación en formación en línea contra un frente de llama.	104
3.20. Esquema de arquitectura de software implementada.	107
3.21. Tiempo de cómputo frente al número de UAVs para el caso no etiquetado.	109
3.22. Tiempo de cómputo frente al número de UAVs para el caso etiquetado.	109
3.23. Distancia total recorrida para el caso etiquetado y no etiquetado en 2D en función del número de UAVs.	110

3.24. Entorno de simulación para validación de método de planificación de trayectorias en 3D.	113
3.25. Distancia total recorrida por el enjambre para planificación en 3D en función del número de UAVs.	115
3.26. Tiempo de cómputo de generación de grafo 3D en función del número de nodos.	116
3.27. Porcentaje de éxito del planificador de trayectorias en 3D en función del número de nodos.	117
3.28. Distancia total recorrida para trayectorias 3D en función del número de nodos.	117
3.29. Ejemplo de suavizado de trayectorias en 2D.	121
3.30. Comparativa del total de waypoints para trayectorias sin suavizar y trayectorias suavizadas en 2D para un enjambre de 5 UAVs.	122
3.31. Comparativa de la distancia total recorrida por un enjambre de 5 UAVs para trayectorias sin suavizar y trayectorias suavizadas en 2D.	123
3.32. Comparativa del tiempo de cómputo empleado para la generación de trayectorias sin suavizar y suavizadas en 2D para un enjambre de 5 UAVs.	124
3.33. Ejemplo de suavizado de trayectorias en 3D.	125
3.34. Comparativa del total de waypoints para trayectorias sin suavizar y trayectorias suavizadas en 3D para un enjambre de 5 UAVs.	127
3.35. Comparativa de la distancia total recorrida por un enjambre de 5 UAVs para trayectorias sin suavizar y trayectorias suavizadas en 3D.	128
3.36. Comparativa del tiempo de cómputo empleado para la generación de trayectorias sin suavizar y suavizadas en 3D para un enjambre de 5 UAVs.	129
4.1. Problemática de la navegación autónoma en enjambre (6) ©2020 IEEE.	132
4.2. Sistema redundante de gestión de evasión de obstáculos.	134
4.3. Distancias relativas y aplicación de factor de reducción para el UAV ₁ (6) ©2020 IEEE.	138
4.4. Distancias relativas y aplicación de factor de reducción para el UAV ₄ (6) ©2020 IEEE.	139
4.5. Distancias relativas y aplicación de factor de reducción para el UAV ₂ (6) ©2020 IEEE.	140
4.6. Distancias relativas y aplicación de factor de reducción para el UAV ₃ (6) ©2020 IEEE.	141

4.7. Información 3D basada en la Nube de puntos de un LiDAR (6) ©2020 IEEE.	143
4.8. Mapa de ocupación generado a partir de la información 3D capturada por el LiDAR (6) ©2020 IEEE.	143
4.9. Diagrama de flujo del método de detección y evasión de obstáculos basado en información 3D del entorno (6) ©2020 IEEE.	144
4.10. Ruta alternativa segura como solución para la navegación sin supervisión de un UAV (6) ©2020 IEEE.	146
4.11. Grafo 3D generado para la obtención de rutas alternativas que eviten la colisión con los obstáculos detectados (6) ©2020 IEEE.	147
4.12. Relación de marcos de coordenadas para el método de evasión de obstáculos propuesto.	148
4.13. Entorno de simulación forestal con presencia de UAVs externos al enjambre.	149
4.14. Sistema de evasión de colisiones basado en perímetros virtuales.	151
5.1. Estructura general de la ANN empleada en el algoritmo DQN.	157
5.2. Recompensa acumulada por época de entrenamiento para el modelo 1 de toma de decisiones.	161
5.3. Recompensa total acumulada por el modelo 1 de toma de decisiones inteligente a lo largo del entrenamiento.	162
5.4. Esquema de cálculo de orientación para permitir la navegación autónoma con el UAV enfocado hacia el objetivo a alcanzar.	164
5.5. Recompensa acumulada por época de entrenamiento para el modelo 2 de toma de decisiones.	165
5.6. Recompensa total acumulada por el modelo 2 de toma de decisiones inteligente a lo largo del entrenamiento.	166
5.7. Limitación del área de exploración para navegación precisa hacia una ubicación prefijada.	168
5.8. Recompensa acumulada por época de entrenamiento para el modelo 3 de toma de decisiones.	170
5.9. Recompensa total acumulada por el modelo 3 de toma de decisiones inteligente a lo largo del entrenamiento.	171
A.1. Esquema general de métodos basados en RL.	185
A.2. Conflicto Exploración-Explotación.	187
A.3. Diagrama método Actor-Crítico.	192
B.1. Enjambre de UAVs homogéneo resultado del desarrollo de hardware realizado.	193

B.2. Chasis S500.	195
B.3. Elemento estructural de PLA para embarcar el sistema de procesa- miento, alimentación y comunicaciones.	196
B.4. Controladora Pixhawk Black Cube y sistema de posicionamiento <i>Here</i> ⁺ embarcados sobre UAV.	197
B.5. Jetson AGX Xavier embarcada sobre UAV.	199
B.6. Esquema general de la arquitectura de hardware propuesta.	200
B.7. Esquema general de conexiones eléctricas dentro de cada UAV.	203
B.8. ESC modelo Tmotor F 55A montado sobre cada uno de los UAVs.	204
B.9. Prototipo demostrador empleado para vuelo autónomo y coordinada en enjambre.	205

Índice de Tablas

1.1. Número de incendios en 2017 comparado con el promedio de 10 años (25).	8
1.2. Superficie quemada en 2017 en comparación con el promedio de 10 años (25).	8
1.3. Datos de siniestros en la Comunidad de Madrid entre los años 2006-2019.	9
1.4. Datos de superficie quemada (en hectáreas) en la Comunidad de Madrid entre los años 2006-2019.	10
2.1. Métodos empleados para planificación de rutas con aplicabilidad a enjambres de UAVs (Parte 1).	67
2.2. Métodos empleados para planificación de rutas con aplicabilidad a enjambres de UAVs (Parte 2).	68
3.1. Tiempo de cómputo en función del número de UAVs.	108
3.2. Comparativa tiempo de cómputo para generación de grafo en función del número de nodos.	111
3.3. Tiempo de cómputo en función del número de UAVs para 3D.	114
3.4. Alcanzabilidad de las tareas considerando el tamaño de los UAVs.	118
3.5. Comparativa de puntos de ruta introducidos en las trayectorias originales y suavizadas en 2D para cada UAV del enjambre.	122
3.6. Comparativa de puntos de ruta introducidos en las trayectorias originales y suavizadas en 3D para cada agente del enjambre.	126
5.1. Estructura de la ANN para el modelo número 1 de DRL.	158
5.2. Estructura de la ANN para el modelo de toma de decisiones número 2 de DRL.	163
5.3. Estructura de la ANN para el modelo 3 de DRL.	169
5.4. Comparativa entre modelos de aprendizaje.	172
5.5. Estructura de la ANN para el modelo número 4 de DRL.	173

5.6. Optimización de trayectoria propuesta por el planificador en 3D mediante el uso de algoritmos de DRL.	174
---	-----

Índice de Algoritmos

- 1. PRM en 3D. 87
- 2. Formación en línea 95
- 3. Formación en cuadrado 99
- 4. Formación en flecha 101

- 5. Control de velocidad para enjambres de UAVs 136

Lista de Acrónimos

- A*** A-Star. 37–41, 44, 45, 65–67, 72, 80, 82, 89–93, 100, 102, 114, 130, 142, 145, 146, 178, 179
- ACO** Ant Colony Optimization. 60
- AESA** Agencia Estatal de la Seguridad Aérea. 2, 4, 6
- ANN** Artificial Neural Network. 60–62, 64, 68, 155–158, 163, 166, 168, 169, 173, 182, 183
- APF** Artificial Potential Field. 51–56, 61, 67
- CHOMP** Covariante Hamiltonina Optimization for Motion Planning. 59
- CMGA** Coevolutionary Multi-population Genetic Algorithm. 60
- CNN** Convolutional Neural Network. 63, 155, 158
- CVT** Tessellation Centroidal Voronoi. 46
- D*** Dynamic A-Star. 37, 41, 45, 67
- DL** Deep Learning. 62, 154, 155, 174, 183
- DMPC** Decentralized Model Predictive Control. 58
- DNN** Deep Neural Network. 62, 155
- DQN** Deep Q-Network. 63, 155–157, 160, 161, 173–175
- DRL** Deep Reinforcement Learning. 154–156, 158–160, 162, 163, 165, 167–169, 171–175, 182–184
- dRRRT** Discrete Rapidly Exploring Random Tree. 50, 52
- EGA** Enhanced Genetic Algorithm. 61
- GA** Genetic Algorithm. 60, 61, 77
- GCS** Ground Control Station. 19, 105, 195, 199–201
- I+D+i** investigación, desarrollo e innovación. 6
- IAPF** Improved Artificial Potential Field. 54, 56
- IoT** Internet of Things. 32
- MADDPG** Multi-Agent Deep Deterministic Policy Gradient. 64
- MDP** Markov Decision Process. 189, 190
- MILP** Mixed Integer Linear Program. 55–57, 68
- MIQP** Mixed Integer Quadratic Program. 55, 57, 58, 68
- ML** Machine Learning. 62, 154
- MPC** Model Predictive Control. 58

- MRS** Multi-Robot Systems. 12, 16, 26–34, 37, 38, 40, 43, 46, 48, 51, 53, 66, 69, 177
- OACI** Organización de Aviación Civil Internacional. 2, 132, 150
- OC** Optimal Control. 55, 58, 68
- PRM** Probabilistic Roads Map. 16, 43–45, 65–67, 72, 77, 80–82, 84, 85, 87, 89, 106, 107, 111, 112, 118, 127–130, 133, 142, 145, 146, 178–180
- PSO** Particle Swarm Optimization. 60–62
- RL** Reinforcement Learning. 17, 18, 62–64, 153–155, 159, 174, 185–190
- ROS** Robot Operating System. 105, 106, 129, 148, 156, 200, 201
- RPAS** Remotely Piloted Aerial System. 1, 9, 19
- RRT** Rapidly Exploring Random Tree. 48–51, 65, 67, 77
- SARSA** state-action-reward-state-action. 190
- TD** Temporal Difference. 191
- TRPO** Trust Region Policy Optimization. 155
- UAV** Unmanned Aerial Vehicle. 1–7, 10–21, 23, 24, 26, 28–35, 38, 40, 41, 43, 44, 46–48, 51–54, 56–61, 63–66, 71–86, 88–98, 100–116, 118–175, 177–184, 186, 193–205
- UE** Unión Europea. 2–4, 6
- UGV** Unmanned Ground Vehicle. 26, 29–31
- WoLF-PHC** Win or Learn Fast Policy Hill-Climbing. 64

Capítulo 1

Introducción

Los Vehículos Aéreos No Tripulados (Unmanned Aerial Vehicles (UAVs)) son aquellos vehículos caracterizados por la ausencia de un piloto a bordo. Este término incluye tanto a los denominados Sistemas Aéreos Pilotados Remotamente (Remotely Piloted Aerial System (RPAS)), como a aquellos UAVs capaces de desempeñar de manera autónoma tareas para las que han sido programados (26; 27).

En las últimas décadas, se ha producido un incremento significativo de las investigaciones relacionadas con UAVs, tanto en el ámbito militar como civil, debido a la capacidad de estos vehículos de poder generar valor añadido. La posibilidad de convertir estos vehículos en potentes herramientas, capaces de actuar en un amplio espectro de escenarios y llevar embarcados diferentes sensores, a un coste inferior al de la aviación tradicional, han provocado la irrupción de un potencial mercado destinado a la investigación y comercialización de los UAVs (28; 29).

Con el fin de aumentar la explotación de estos sistemas han surgido, en la actualidad, los denominados enjambres de UAVs, es decir, equipos formados por distintos UAVs que, trabajando de manera coordinada, permiten aumentar la aplicabilidad de estos sistemas. Las investigaciones relacionadas con la capacidad de los UAVs para desempeñar tareas de manera autónoma y coordinada son uno de los principales impulsos para el desarrollo de estos enjambres inteligentes.

En este capítulo se recoge, en primer lugar, el impacto político, social y económico provocado por el desarrollo de la industria de los UAVs, junto con los antecedentes tecnológicos que han ido ligados a la irrupción y avance de esta tecnología. En segundo lugar, se establecen los desafíos que motivan esta disertación, los objetivos establecidos en la ejecución de la tesis y, por último, la estructura a seguir a lo largo del presente documento.

1.1. Contexto Político, Social y Económico

El auge de los UAVs, o también llamados popularmente drones, ha provocado que, las autoridades competentes, se vean obligadas a introducir cambios y mejoras a nivel legislativo que permitan la convivencia de este tipo de aeronaves con la aviación comercial tradicional.

Por este motivo, desde la Organización de Aviación Civil Internacional (OACI) se han realizado cambios que han permitido al conjunto de países establecer una legislación en referencia al uso de los UAVs. Así pues, en el año 2011, en busca de garantizar un marco común de navegabilidad aérea, la OACI estableció que: *‘El hecho de que la aeronave sea tripulada o no tripulada no afecta su condición de aeronave’* (30). Este cambio legislativo, motivó a todos los países a realizar cambios en sus normativas estatales para incluir dentro de los Sistemas de Transporte Aéreo a todos los actores derivados del sector de los drones.

A nivel nacional, la irrupción del sector de los drones en España produjo la introducción de cambios legislativos que garantizaran el crecimiento y desarrollo de un conjunto de actividades relacionadas con el sector. Por esta razón, la Ley 18/2014 del 15 de octubre extendió el concepto de aeronave a *“cualquier máquina pilotada por control remoto que pueda sustentarse en la atmósfera por reacciones del aire que no sean las reacciones del mismo contra la superficie de la tierra”* (31). Además, se introdujeron una serie de limitaciones y obligaciones que permitieran regular, de manera clara, las operaciones y actividades a desempeñar por los operadores de drones. Estas reformas legislativas trajeron un crecimiento exponencial del sector, reflejado, principalmente, en el número de operadoras de drones, que aumentó de 31 empresas en 2014 a 2.513 empresas en agosto de 2017, según datos la Agencia Estatal de la Seguridad Aérea (AESA).

El rápido auge de este mercado condujo a la necesidad de introducir nuevos cambios a nivel legislativo, creándose un nuevo marco regulador para el sector de los UAVs. Así, el Real Decreto 1036/2017, del 15 de diciembre, establecía, con respecto a la norma anterior, un mayor número de aplicaciones y servicios a explotar con este tipo de aeronaves, constituyendo además un marco estable para la extensión de toda la cadena de valor dentro de esta industria (32).

Recientemente, los estados miembros de la Unión Europea (UE) han trabajado por establecer un conjunto de reglas comunes que estandarice las normativas nacionales en el ámbito de las operaciones con drones, de tal manera que se favorezca la expansión del sector dentro de Europa y, permita alcanzar un posicionamiento de este dentro del mercado mundial, a la altura de las grandes potencias de esta área, como son China y Estados Unidos (33). De este trabajo, la Comisión Europea ha elaborado dos reglamentos sobre aeronaves no tripuladas comunes para todos los Estados, que se encuentran orientados a regular el uso civil de estas aeronaves.

A partir del Reglamento Delegado (UE) 2019/945 de la Comisión Europea sobre los sistemas de aeronaves no tripuladas y los operadores de terceros países (34) y, del Reglamento de Ejecución (UE) 2019/947 de la Comisión Europea relativo a las normas y procedimientos aplicables a la utilización de aeronaves no tripuladas (35), se ha establecido en todos los países de la UE una nueva normativa comunitaria cuyas disposiciones contrarias al Real Decreto 1036/2017 son aplicables a nivel nacional a partir del 31 de diciembre de 2020, manteniendo vigentes aquellos aspectos no contemplados en el marco europeo, como son las disposiciones en materia de seguridad pública o restricciones al vuelo de sistemas aéreos no tripulados por operar en lugares contemplados en el Real Decreto 1036/2017. No obstante, se espera que se produzca una nueva normativa a nivel nacional que regule los aspectos que la normativa europea deja a criterio de los estados miembros.

Esta nueva normativa comunitaria se basa en establecer unos principios generales de operaciones regulados en función del riesgo, es decir, se establecen tres categorías de operaciones a las que se les asocia un conjunto de requisitos determinados en función del riesgo de la operación y de las características operacionales del dron y del área de operaciones. Por tanto, siguiendo estos aspectos, se establece una división en tres categorías de las operaciones con UAVs: *categoría abierta*, *categoría específica* y *categoría certificada*, cuyo objetivo no es otro que dinamizar el mercado de los UAVs en el ámbito civil de una manera segura y sostenible, a través de un marco normativo común y estable que se aplique a toda la cadena de valor del sector. Junto a estos cambios, contempla las operaciones autónomas, es decir, sin intervención por parte del piloto y, la posibilidad, en función de la categoría operacional, de realizar transporte aéreo con UAVs y volar en enjambre (36).

El análisis legislativo y político de cualquier tecnología no debe abordarse de manera individualizada, ya que se encuentra directamente relacionado con como estos avances tecnológicos son instaurados y aprovechados por la sociedad, y con ello, como dicho mercado afecta de manera directa a la economía.

La aparición de los UAVs trajo consigo las primeras regulaciones que han favorecido su desarrollo y crecimiento, convirtiendo al sector en un mercado emergente y de gran potencial, siendo necesario continuar con el desarrollo de tecnologías orientadas a la consolidación de las aplicaciones más extendidas y, generando nuevos servicios y actividades. Los continuos avances en el área de los UAVs, representados en aspectos como la reducción de los costes o la mejora de la autonomía y fiabilidad, han ayudado a la expansión y consolidación de la industria.

Todos los avances, tanto legislativos como tecnológicos, han permitido, que en la actualidad, los UAVs se consideren una herramienta tecnológica potente, heterogénea y versátil capaz de desempeñar multitud de tareas con el simple hecho de adaptar los sistemas embarcados (o carga de pago) en la aeronave.

La alta aplicabilidad de los drones en el sector civil, ligado a su uso en el ámbito

recreativo, ha provocado que esta tecnología presente un alto impacto dentro de la sociedad y, su posicionamiento como uno de los mercados con mayores expectativas de crecimiento en los próximos años dentro de la UE. Estos hechos han provocado que un sector altamente técnico y cualificado, como es el de la aviación tradicional, comience a adaptarse de cara a poder convivir con el crecimiento de estas aeronaves. Ventajas como la reducción de las barreras de entrada al sector, la ayuda o sustitución de tareas susceptibles de ocasionar riesgos personales, el poder de adaptación y versatilidad para desempeñar un amplio número de trabajos y servicios y, todo ello, acompañado de un menor impacto sobre el medio ambiente constituyen razones suficientes para acometer medidas que permitan a este sector convivir con la aviación tradicional, coexistiendo los drones con el resto de usuarios y gestores del espacio aéreo.

En la actualidad, según AESA, en España hay declarados 5019 operadores de UAVs (11), crecimiento que demuestra que, tanto la reducción de costes, a nivel de adquisición o construcción de una aeronave no tripulada y a nivel de operativa, como la reducción de los requisitos profesionales necesarios para desempeñar trabajos dentro de la industria de los UAVs, son barreras de entrada menos restrictivas que las contempladas por la aviación comercial tradicional. Esto ha provocado, en primer lugar, que se produzcan cambios legislativos en favor de los drones, tal y como se ha descrito previamente; y, en segundo lugar, que esta tecnología haya tenido un alto impacto social y económico en los últimos años.

La aceptación de esta tecnología no viene solo ligada a la disminución de los requisitos para acceder a ella, sino también a la posibilidad de incorporar estas herramientas a un gran conjunto de trabajos y sectores tradicionales, mejorando la eficiencia y productividad a medida que se reducen los riesgos laborales. Junto a esta aceptación, empresas e instituciones, tanto a nivel público como privado, continúan con investigaciones que desembocan en nuevos servicios y aplicaciones de los UAVs. La posibilidad de contar con una aeronave capaz de flexibilizar las fases de despegue y aterrizaje sin necesidad de infraestructuras específicas, de volar en áreas de difícil acceso o de acceder a zonas peligrosas sin poner en riesgo vidas personales, han provocado que sectores como la obra civil, la agricultura o la minería hayan incorporado esta tecnología en busca de aumentar la productividad.

Actualmente, los drones son empleados en tareas muy heterogéneas que pueden ir desde la inspección de zonas de difícil acceso (aerogeneradores, paneles solares, tendido eléctrico), los levantamientos topográficos o el seguimiento de cultivos o ganados, hasta el control de fronteras o del tráfico. Aunque el sector de los drones no se acota únicamente al uso de estas aeronaves como una herramienta para prestar servicios dentro de otros sectores, sino que, tal y como se muestra en la Figura 1.1, involucra a otras organizaciones como las destinadas al diseño y fabricación de nuevos sistemas, a la formación de personal técnico dentro del

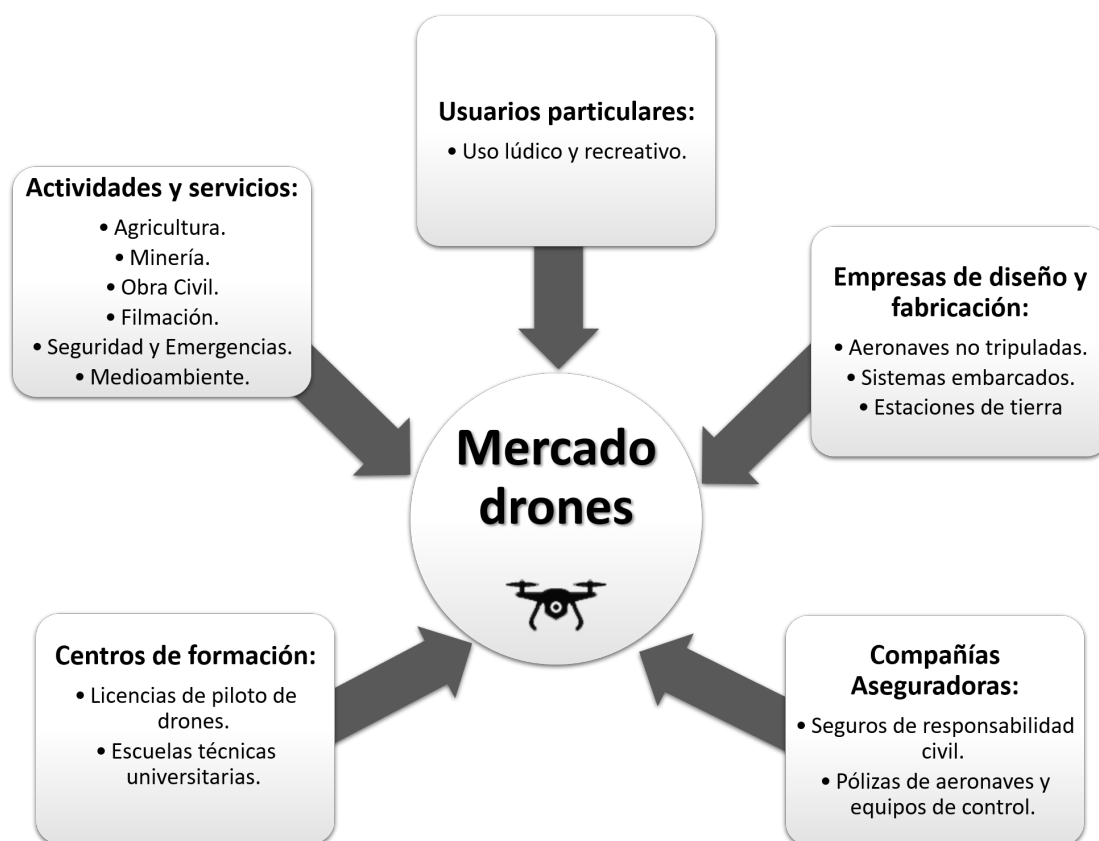


Figura 1.1: Ejes productivos del mercado de UAVs.

sector o, a asegurar tanto el material empleado como posibles daños a terceros debido a que, actualmente, las normativas vigentes exigen la suscripción de seguros que cubran responsabilidades en caso de accidentes, por lo que las compañías aseguradoras se posicionan como un elemento clave dentro del sector de los drones.

Para cerrar el análisis, es necesario poner cifras al sector que ayuden a entender los avances y cambios realizados, tanto a nivel legislativo como social. A nivel mundial, son China y Estados Unidos los principales dominadores de la aviación no tripulada, aunque en diferentes ámbitos, ya que, en EE.UU. es el ámbito militar el principal consumidor de esta tecnología, mientras que China posee un mercado más orientado al sector civil. A continuación, aparece Europa dónde se estima que, en la actualidad, el parque de drones podría situarse entre 1 y 1,5 millones de aeronaves, de las cuales, 10.000 unidades serían drones de uso profesional (37).

El auge experimentado por el sector ha sido tal que, en 10 años, se ha triplicado tanto el número de modelos de drones, como las empresas dedicadas a su fabricación a nivel mundial, pasando de 785 modelos y 252 empresas en 2006 a 2007

modelos y 681 empresas en el año 2016 (38). Este crecimiento se ha acentuado en los últimos años y, solo del año 2015 al año 2016 se produjo un incremento en el volumen de fabricación de drones del 60,3 % (39).

A nivel europeo, el trabajo comunitario tanto a nivel legislativo como financiero ha permitido posicionar a los países de la UE y, reducir la desventaja con países como Estados Unidos. Así pues, en el año 2016 el mercado de los drones destinados a usos civiles generó 259 millones de euros en ingresos dentro de la UE (40), siendo los drones comerciales para usos profesionales el segmento más importante, a diferencia del mercado estadounidense marcado por el ámbito militar y, del mercado chino basado en el segmento de drones recreativos.

A nivel nacional, el sector en España se ha sustentado en dos ejes productivos: el diseño y fabricación de UAVs junto con sus sistemas embarcados; y, el uso de UAVs profesionales para prestar servicios a través de operadores habilitados.

En cuanto al primer eje, el tejido empresarial se caracteriza por empresas de pequeño y mediano tamaño de reciente creación, con importantes inversiones en investigación, desarrollo e innovación (I+D+i), en busca de nuevos prototipos y sistemas aéreos innovadores.

En cuanto al segundo eje, son cada vez más el número de operadores registrados cuya finalidad se centra en prestar servicios con drones al usuario final. Analizando los datos proporcionados por AESA y, tal y como se observa en la Figura 1.2, todas las operadoras registradas contemplan como actividad la fotografía, filmación o levantamientos, pero en un amplio número de casos, combinan este tipo de servicios con otros tan importantes como la vigilancia aérea, las operaciones de emergencias, búsqueda y rescate o, las actividades de investigación y desarrollo. Como sucede con las empresas del primer eje, estos operadores destacan por ser empresas de pequeño tamaño y reciente creación con un posicionamiento claro en cuanto al tipo de actividades que desarrollan, aunque, recientemente, está aumentando el uso de esta tecnología entre los cuerpos y fuerzas de seguridad del Estado, lo que posiciona a los drones como potentes herramientas en el campo de la vigilancia, las emergencias, la seguridad y el salvamento. En este sentido, es importante destacar que las previsiones realizadas indican que en el año 2035 se podría alcanzar una flota de 51.400 drones profesionales, y llegar a los 53.500 en el año 2050, lo cual supone un aumento considerable comparado con las 4402 aeronaves no tripuladas registradas en febrero de 2018 por AESA. En cuanto al impacto económico de este crecimiento, se prevé que en el año 2035 el volumen de negocio alcance los 1.220 millones de euros a raíz de la actividad generada por cerca de 11.000 puestos de trabajo, mientras que en el 2050 podría alcanzar los 1.520 millones de euros y los 11.500 puestos de trabajo (38).

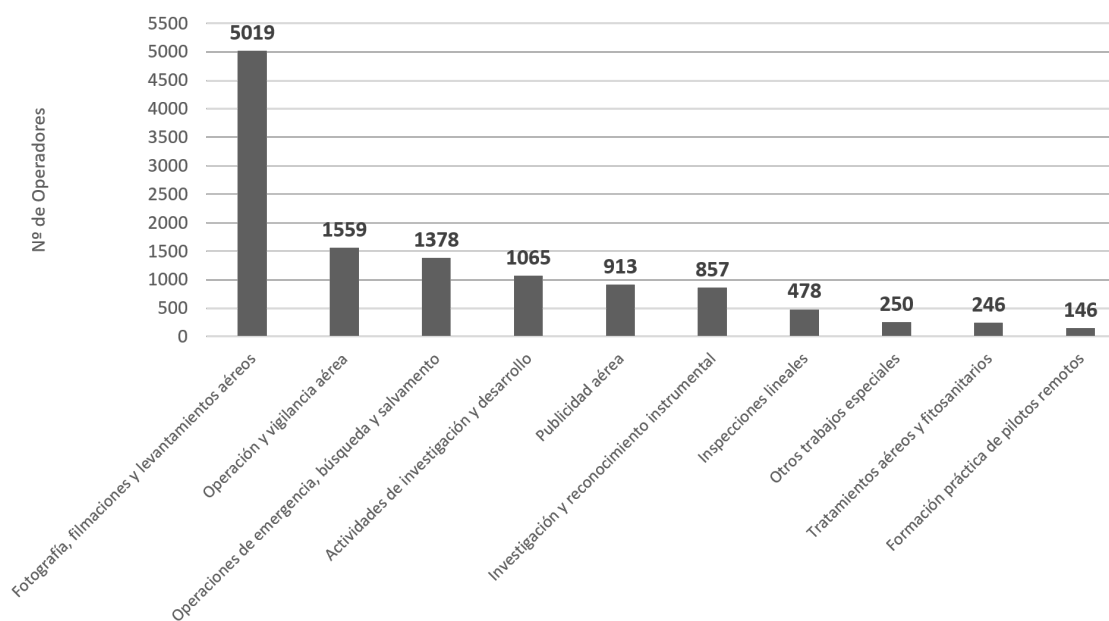


Figura 1.2: Resumen de actividades desarrolladas por los operadores (11).

1.2. Motivación

Una de las áreas dónde se pronostica que los drones van a tener una alta influencia es en el campo de la seguridad y las emergencias. En estos momentos, una parte de los cuerpos de seguridad nacionales ya cuentan con esta herramienta para llevar a cabo labores de vigilancia, control del tráfico o, incluso, para dar soporte en accidentes, problemas de salud pública como la crisis sanitaria del COVID-19 o catástrofes medioambientales como los incendios forestales. Y, en referencia a esta última cuestión, los UAVs han irrumpido con un impacto positivo en cuanto al medio ambiente se refiere, no solo por su papel dentro de la agricultura de precisión, sino por las investigaciones y desarrollos que han provocado que los drones sean una solución real y, un apoyo a emplear, en la detección y extinción de incendios, lo cual, supone uno de los principales problemas medioambientales del mundo y, en concreto, de países del Sur de Europa como España.

Los incendios son considerados como uno de los peligros vitales para la vida silvestre, el medio ambiente salvaje o urbano y, como uno de los principales factores que afectan gravemente a las economías de los países. Además, la lucha contra los incendios requiere de un gran número de personas para llevar a cabo actividades peligrosas, que desgraciadamente, causan un gran número de víctimas al año. Los incendios forestales afectan a 67 millones de hectáreas (ha) en todo el mundo por año, aproximadamente el 1,7% de la superficie terrestre (41), y cuestan más de

2.000 millones de euros al año, considerando tanto la lucha contra el fuego como los daños económicos (42). Los costos sociales y ambientales incluyen los daños a la salud humana y las muertes (estimadas en 340.000 muertes prematuras al año a causa de los incendios) (43) e importantes daños a la vida silvestre y al suelo, que provocan la deforestación y la liberación de gases de efecto invernadero, ligado al impacto económico directo en el paisaje (turismo) y los daños a las infraestructuras, los cuáles se consideran también importantes.

En muchos casos, estos accidentes son causados por la falta de información en tiempo real sobre el estado del incendio o por la dificultad para alcanzar en tiempo el punto de origen del fuego. Según el informe sobre las pérdidas económicas a causa de incendios en Estados Unidos (44), se produjeron alrededor de 1.298.000 incendios en 2014, los cuales causaron 15.775 lesiones y un coste económico estimado de 11,6 mil millones de dólares en pérdidas directas de propiedad.

A nivel nacional, en el año 2017, España sufrió uno de los peores años de la década con un incremento del 10 % en el número de incendios y un 95 % en cuanto hectáreas afectadas, tal y como se recoge en el informe anual de la Comisión Europea (25) y, se muestra en las Tablas 1.1 y 1.2.

Tabla 1.1: Número de incendios en 2017 comparado con el promedio de 10 años (25).

	Promedio 2007–2016	2017
Número de incendios < 1 ha	8.228	8.705
Número de incendios \geq 1 ha	4.135	5.088
Total	12.363	13.793

Tabla 1.2: Superficie quemada en 2017 en comparación con el promedio de 10 años (25).

	Promedio 2007–2016	2017
Área quemada en otras tierras boscosas (ha)	27.226,41	66.839,02
Área de bosque quemada (ha)	91.846,74	178.233,93

Si se analiza el impacto de los incendios forestales a nivel autonómico (45; 46; 47; 48; 49), se observa que, tanto el nivel de siniestros, mostrados en la Tabla 1.3, como la cantidad de superficie quemada, recogida en la Tabla 1.4, en los últimos años, sigue siendo un problema medioambiental sin resolver, dado que tanto el nú-

mero de siniestros, como la superficie forestal total quemada se mantiene constante con el transcurrir de los años.

Tabla 1.3: Datos de siniestros en la Comunidad de Madrid entre los años 2006-2019.

Año	Siniestros		
	Nº		Total Siniestros
	Conatos	>1ha	
2006	196	75	271
2007	162	68	230
2008	171	52	223
2009	269	58	327
2010	158	21	179
2011	234	60	294
2012	313	86	399
2013	233	53	286
2014	297	66	363
2015	264	73	337
2016	224	47	271
2017	293	51	344
2018	234	21	255
2019	285	81	366

Si bien es cierto que, en el año 2012, se produjo un máximo de incidencias que conllevó a un total de 2.381 hectáreas de área forestal quemadas y, que a partir de entonces se ha ido reduciendo el impacto del fuego sobre el medio ambiente, no se puede considerar un problema resuelto o controlado, ya que, como recoge el informe del Ministerio de Agricultura, Pesca y Alimentación del año 2019 (49), ese mismo año ardieron en la Comunidad de Madrid 3.088,23 hectáreas de superficie forestal, siendo el dato más alto de los registrados hasta la fecha desde el año 1985.

De cara a poder disminuir y, en gran medida, evitar desastres naturales relacionados con los incendios forestales, las aeronaves no tripuladas se presentan como una importante herramienta tecnológica capaz de realizar tareas de supervisión, detección, vigilancia, recopilación de información en tiempo real y tareas de extinción. Aunque en la actualidad, ya se cuenta con tecnología que permita llevar a cabo parte de estas tareas mediante el manejo de RPAS, es necesario continuar con la investigación y desarrollo de aplicaciones que permitan, a este tipo de aeronaves, trabajar de manera autónoma en entornos tan dinámicos, complejos y adversos como son los incendios. De esta manera, no solo se podrá reducir los efectos negativos del fuego y, mejorar los tiempos de actuación y respuesta,

Tabla 1.4: Datos de superficie quemada (en hectáreas) en la Comunidad de Madrid entre los años 2006-2019.

Año	Superficie Forestal				Total Forestal	No Forestal
	Arbolada	No Arbolada				
		Leñosa	Herbácea	Total		
2006	29,04	235,45	830,46	1.065,91	1.094,95	276,46
2007	10,16	166,23	273,36	439,59	449,75	143,37
2008	24,87	130,69	6,72	137,41	162,28	15,42
2009	15,58	59,64	249,42	309,06	324,64	65,52
2010	0,67	14,97	85,82	100,79	101,46	157,66
2011	0,86	37,37	346,77	384,14	385	302,03
2012	886,78	839,82	654,49	1.494,31	2.381,09	880,48
2013	212,1	429,68	691,76	1.121,44	1.333,54	359,52
2014	1,14	56,48	306,42	362,9	364,04	117,92
2015	46,47	68,56	471,96	540,52	586,99	207,75
2016	8,98	28,6	380,08	408,68	417,66	Sin datos
2017	10,31	47,07	201,36	248,43	258,74	Sin datos
2018	2,55	6,84	93,07	99,91	102,46	Sin datos
2019	1407,05	726,53	954,65	1.681,18	3.088,23	Sin datos

sino que el trabajo autónomo de estas herramientas traerá consigo la reducción del personal desplegado físicamente en el área de incidencia del fuego y, con ello, se disminuirá la probabilidad de que se ocasionen riesgos personales o pérdidas de vidas humanas. Por este motivo, entre los objetivos del presente trabajo se encuentra el estudio, desarrollo e implementación de métodos que permitan navegar, sin supervisión, a un enjambre de UAVs destinado a cooperar y trabajar de manera coordinada en entornos dinámicos como son los incendios, tanto forestales como urbanos, pudiendo desempeñar tareas de apoyo a los equipos de emergencia desplegados en la zona.

1.2.1. Definición del problema de extinción de incendios

La extinción de un incendio forestal está basada en un conjunto de tareas de alta complejidad realizadas por helicópteros, hidroaviones y bomberos (50). De ellos, los helicópteros e hidroaviones constituyen herramientas esenciales y críticas para el apoyo a los equipos y personal en tierra y, para una consecución exitosa del trabajo. Sin embargo, estos sistemas aéreos presentan un conjunto de limitaciones y desventajas a la hora de afrontar trabajos en la lucha contra el fuego.

La primera y más significativa, es la necesidad de poner en riesgo vidas huma-

nas, debido a la necesidad de volar en entornos adversos a baja altura. En segundo lugar, encuentra la imposibilidad de realizar labores de extinción por la noche, debido a la seguridad de los pilotos (51), lo que conlleva, en algunos casos, a la reactivación de los incendios durante este período del día y limita el tiempo de operación a un promedio de 12 horas diarias.

Por otro lado, los hidroaviones y helicópteros son máquinas complejas, lo que provoca que, tanto su precio de compra, como los precios derivados de la operación con los mismos sean considerablemente altos, sobre todo, al compararlo con el coste de adquisición y operación de drones profesionales convencionales.

Por último, tanto hidroaviones como helicópteros presentan restricciones durante las maniobras de liberación de agua, lo que compromete la eficacia y precisión de las tareas de extinción de incendios (52). La liberación de agua es altamente ineficiente, ya que se basa en la apertura de un depósito y el vertido del agua por gravedad a una cierta velocidad. Además, presenta la problemática de la distancia, ya que en el caso de los hidroaviones no pueden llevar a cabo la tarea a una velocidad horizontal por debajo de su velocidad de parada (superior a 100 km/h) y, en el caso de los helicópteros debe realizarse a una altura superior a los 50 metros , ya que la corriente de aire del rotor avivaría el fuego.

En este escenario, los drones se presentan como un sistema con un alto potencial al superar todos estos inconvenientes, pudiendo en primer lugar, combatir el fuego sin la necesidad de poner en riesgo vidas humanas. Además, estas aeronaves no tripuladas pueden embarcar sistemas que les permita realizar operaciones nocturnas y a baja altitud, mejorando la eficacia en la lucha contra el fuego (53; 54; 55).

La lucha contra los incendios está formada por cuatro fases clásicas: prevención, detección, extinción y evaluación de daños, y en cada una de ellas se engloban aspectos en los cuales los UAVs pueden aportar soluciones esenciales. Así pues, en el caso de la prevención destacan como tareas adaptables a los UAVs el inventario de especies o la recopilación de información del entorno como puede ser la densidad del arbolado, la altura de la vegetación o el grado de humedad. En el caso de la detección, el uso de este tipo de vehículos se orienta a la vigilancia de áreas específicas de alto riesgo. En el caso de la extinción, el uso de UAVs se enfoca a actuar como soporte de los equipos de emergencia localizando nuevos focos, personas en zonas de difícil acceso o controlando la posición de los equipos de extinción en tierra. Por último, una vez sofocado el incendio, los drones pueden ser de utilidad para la elaboración de informes de evaluación de daños.

A día de hoy, y a pesar de estas posibilidades, el uso de drones en incendios se encuentra mayoritariamente destinado a tareas de supervisión, detección, vigilancia y obtención de datos e información en tiempo real, todos ellos trabajando, en su mayoría, de manera teleoperada desde pilotos situados en tierra (56), pero sin la presencia aún de UAVs capaces de abordar labores completas de extinción

de incendio, debido principalmente, a ser sistemas alimentados por componentes eléctricos, lo que les hace carecer de potencia suficiente para levantar altas cargas y, limitando los tiempos de autonomía.

Por lo que a esta actividad se refiere, el uso de drones se encuentra limitado, en primer lugar, a operaciones teleoperadas o donde la aeronave realiza tareas de manera automática mientras se supervisa y controla su misión con pilotos en tierra; y en segundo lugar, a tareas en las que la carga de peso necesaria para su correcta ejecución no requiera una alta capacidad de carga, dado que se encuentra limitada, al igual que la autonomía, al uso de baterías y sistemas eléctricos que alimenten a la aeronave.

Por tanto, aunque se cuenta con nuevas herramientas que permiten mejorar la eficacia de las misiones orientadas a la extinción de incendios, se hace necesario el estudio y desarrollo de nuevas tecnologías y aplicaciones que permitan a estas aeronaves acometer tareas más complejas, con mayor versatilidad y capacidad de carga y, aumentando el tiempo de operatividad durante las misiones a la vez que operan de manera autónoma. Junto a estos avances y desarrollos, serán necesarios cambios legislativos que flexibilicen el uso de drones profesionales de altas capacidades, en los que la masa máxima al despegue sea mayor de 25 Kg y, que permitan la integración en el espacio aéreo de drones autónomos, dotados de Inteligencia Artificial (Artificial Intelligent (AI)) y capaces de operar sin la necesidad de un piloto de drones (57). En línea con estos avances se encuentra el presente trabajo, en el que, tras un estudio de los distintos métodos presentes en la literatura, se trata de establecer un conjunto de desarrollos que permitan a un enjambre de UAV trabajar de manera coordinada y autónoma en áreas en las cuales se encuentran declarados incendios.

1.2.2. Definición de la problemática en enjambres autónomos de drones

El empleo de drones en actividades en las que tanto el tiempo de respuesta como la eficacia a la hora de completar la misión son aspectos claves, han dirigido sus avances e investigaciones hacia el uso de sistemas multi-robot (Multi-Robot Systems (MRS)), en los que un conjunto de vehículos no tripulados trabaja de manera autónoma y coordinada en un mismo entorno de trabajo. Esta solución tecnológica, basada en el uso coordinado de varias aeronaves no tripuladas, permite disponer de una herramienta más versátil, capaz de acometer tareas más complejas, mejorar la capacidad de carga global del sistema junto con el aumento de la eficacia y la autonomía de todo el sistema.

Para hacer de estos enjambres de drones una solución real a problemas como la extinción de incendios, son necesarios avances tecnológicos en el campo de la AI que

permitan, al conjunto de aeronaves no tripuladas, trabajar de manera coordinada en un misma área, teniendo conocimiento del conjunto de aeronaves y actuando en consecuencia. En definitiva, es necesario dotar a los UAVs de la capacidad para tomar decisiones y completar misiones en entornos adversos y dinámicos de manera efectiva, considerando todos los aspectos del entorno, incluido el resto de aeronaves que forman parte del enjambre.

Del conjunto de desafíos que forman parte de la AI, la planificación de trayectorias, la percepción y entendimiento del entorno y, la toma de decisiones es crucial para conseguir enjambres formados por sistemas aéreos inteligentes. Todos ellos, son campos muy presentes en la literatura actual, con grandes líneas de investigación y avances tecnológicos importantes, los cuales han permitido que día tras día los sistemas de transporte sean cada vez más inteligentes, pero sin duda, es necesario continuar la investigación para resolver problemas como la cooperación y navegación autónoma de enjambres de UAVs.

Para ello, los laboratorios y equipos dedicados a la investigación en el campo de los vehículos aéreos no tripulados deben trabajar para implementar sistemas capaces de:

- **Establecer rutas y trayectorias seguras**

Permitir al enjambre navegar de manera coordinada y alcanzar determinadas ubicaciones a través de trayectorias óptimas que aseguren la navegación segura del enjambre a través de puntos de ruta.

- **Recoger y comprender información del entorno**

Dotar al enjambre de equipos y sistemas necesarios para una navegación coordinada en la que cada aeronave adquiera, interprete y comparta la información del entorno, generando un conocimiento común de la situación.

- **Tomar decisiones por si mismos**

Poder resolver el desafío de que los sistemas, de manera individualizada, pero en este caso, teniendo en cuenta conocimiento común, sean capaces de tomar sus propias decisiones, es decir, dotar a los sistemas aéreos no tripulados de la inteligencia necesaria para actuar a partir de la información obtenida. La reciente aparición de algoritmos de aprendizaje por refuerzo o de aprendizaje profundo han provocado que surjan líneas de investigación que se orientan hacia la posibilidad de que estos sistemas tomen decisiones por si mismos y, puedan navegar de manera autónoma a través de todo tipo de entornos.

1.3. Propuesta de la Tesis

El principal objetivo de esta disertación es proponer una arquitectura de software que permita a un enjambre de drones completar misiones sin supervisión, navegando de manera autónoma y coordinada hasta zonas de interés considerando la información percibida del entorno y, llevando a cabo, tomas de decisiones que permitan acometer tareas de manera satisfactoria, entre las cuales se encuentren actividades relacionadas con la lucha contra el fuego. La idea es constituir una arquitectura formada por diferentes capas, tal y como se presenta en la Figura 1.3, que permita obtener una solución tecnológica robusta, versátil y escalable, de tal manera que un enjambre de UAVs pueda completar, de manera autónoma, diferentes tipos de misiones mediante el uso de un número configurable de UAVs.

En concreto, los métodos propuestos son implementados para alcanzar los siguientes objetivos:

- Generar y planificar trayectorias seguras que permitan al enjambre de drones navegar a través de puntos de ruta de manera coordinada, segura y sin supervisión. Los algoritmos propuestos se caracterizan por ser configurables y escalables, de tal manera que, la planificación de trayectorias pueda generar soluciones para un número indeterminado de drones y, además, permita al enjambre navegar hasta una misma ubicación, en la cual se establezca una formación concreta, o bien, permita a cada uno de los agentes del mismo alcanzar una ubicación concreta, de tal manera que, el conjunto de trayectorias sea óptimo. De esta forma, se establece un método de planificación de trayectorias que permita a dicho enjambre acometer trabajos cooperativos orientados a la lucha contra incendios.
- Establecer un conocimiento común de posicionamiento y localización de las aeronaves para permitir la navegación segura del enjambre. Mediante la creación de perímetros virtuales alrededor de cada miembro del enjambre, se podrá determinar cuando el posicionamiento de al menos dos de los drones suponga un peligro para la seguridad del mismo, momento en el cual es necesario realizar maniobras evasivas que garanticen la no colisión de las aeronaves. De esta forma, gracias al conocimiento del conjunto y a la posibilidad de realizar maniobras evasivas, cada miembro podrá adaptar sus movimientos a las situaciones cambiantes propias de entornos como los de los incendios forestales y urbanos.
- Adquirir, procesar e interpretar la información percibida del entorno a través de sensores embarcados en las aeronaves, de tal manera que se puedan identificar obstáculos o situaciones externas adversas al enjambre de UAVs.

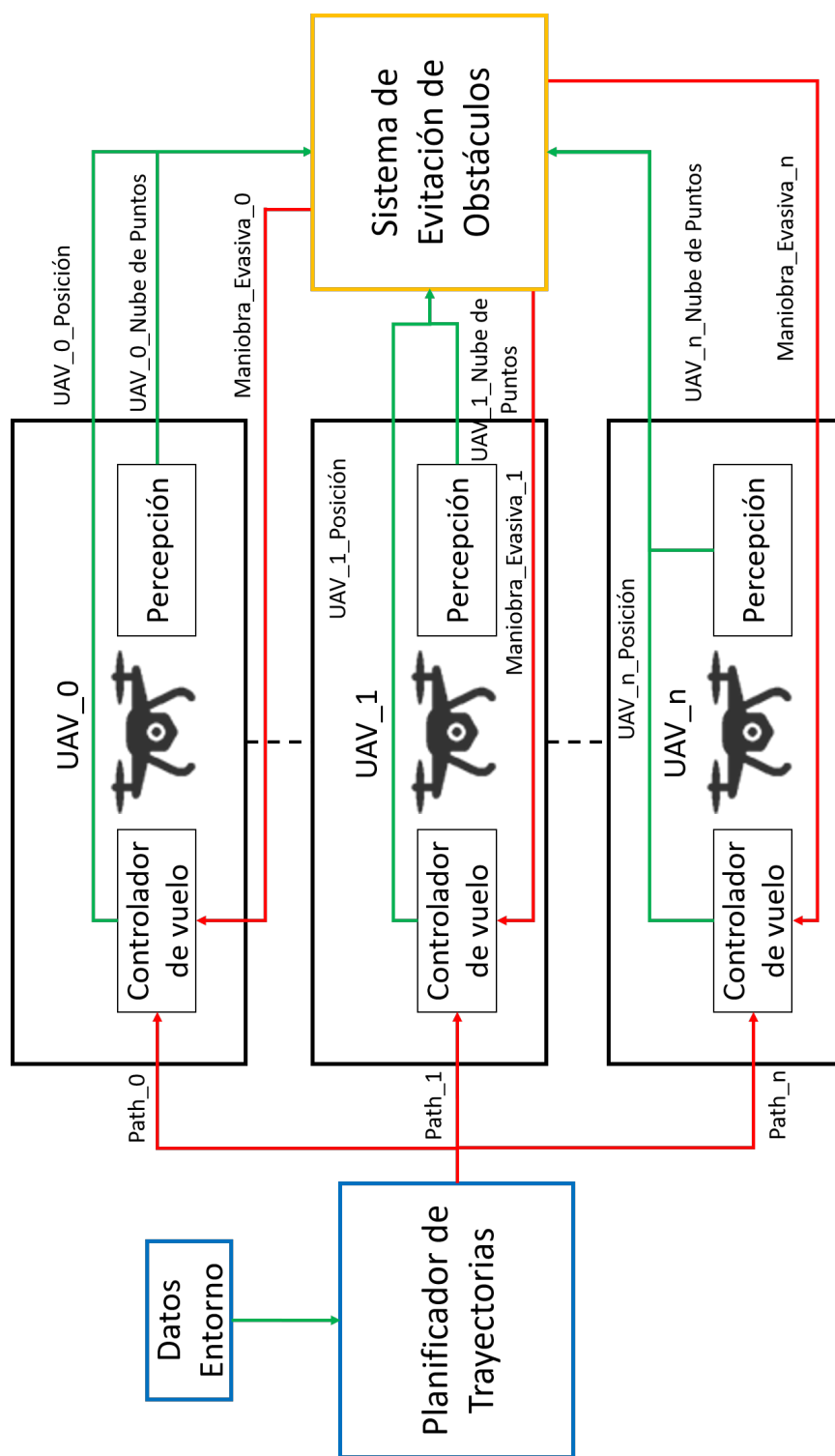


Figura 1.3: Esquema general del enjambre de UAVs autónomo propuesto.

El entendimiento de esta información permitirá al sistema replanificar, en tiempo real, nuevas trayectorias que permitan alcanzar, de manera segura y sin supervisión, una localización determinada. De nuevo, la idea principal de poder desplegar enjambres de UAVs para acometer labores en la lucha contra incendios hace indispensable el implementar sobre cada agente algoritmos que permitan percibir e interpretar la información del entorno y actuar en consecuencia para evitar amenazas y alcanzar objetivos de interés de manera autónoma.

- Dotar a las aeronaves del sistema de algoritmo inteligente que les permita tomar decisiones sobre las acciones y movimientos a realizar con el objetivo de navegar de un punto de ruta a otro de manera segura y eficiente. La metodología propuesta basada en aprendizaje por refuerzo permitirá a cada aeronave tomar decisiones en tiempo real de manera automática a través de la interpretación del entorno en el que se desarrolla la operación, mejorando las prestaciones de los sistemas aéreos y, flexibilizando su adaptación a nuevos entornos de navegación de forma autónoma.

1.4. Contribuciones y Estructura

Esta disertación demuestra la eficiencia y el rendimiento de los algoritmos de planificación de trayectorias empleados en enjambres de drones. A lo largo de la tesis doctoral se proponen un conjunto de métodos basados en planificación de trayectorias, percepción del entorno y aprendizaje por refuerzo para resolver los problemas de navegación autónoma y coordinada de un enjambre de drones, capaz de adquirir e interpretar información del entorno para detectar obstáculos y evitar colisiones gracias a la implementación de algoritmos de toma de decisión.

Las contribuciones se organizan en la tesis de la siguiente manera:

- **Capítulo 2: Estado del Arte**

Este capítulo presenta una revisión de la literatura sobre los algoritmos y técnicas de planificación de trayectorias para enjambres de UAVs. La revisión se completa con el estado de la tecnología alcanzada en los vehículos aéreos no tripulados junto con los avances en el área de los MRS, área tecnológica que engloba a los enjambres de drones, sobre los que se hace hincapié.

- **Capítulo 3: Planificación de Trayectorias para enjambres de UAVs**

Este capítulo presenta los algoritmos de planificación de trayectorias basados en Hojas de Ruta Probabilísticas (Probabilistic Roads Map (PRM)), y se enfoca en la evaluación del rendimiento de algoritmos de planificación

de trayectorias, tanto en 2D como en 3D, para enjambres heterogéneos y escalables de drones.

Las implementaciones realizadas muestran que los métodos basados en las PRMs constituyen un método capaz de establecer rutas seguras en tiempo real para enjambres formados por un número variable de UAVs, que pueden actuar en diferentes casos de uso y, en entornos de grandes dimensiones y urbanos, de cara a poder desplegar enjambres de UAVs que realicen tareas de apoyo y soporte a los equipos de emergencias tanto en incendios urbanos como forestales.

■ **Capítulo 4: Arquitectura de navegación autónoma para entornos dinámicos**

- Este capítulo recoge los métodos implementados para permitir la navegación segura del enjambre mediante algoritmos que permitan capturar y compartir información del enjambre y del entorno para poder detectar y evitar obstáculos en el mismo. Este capítulo se encuentra dividido en diferentes desarrollos:
 - Una primera parte basada en una arquitectura centralizada encargada de recopilar información del estado y posicionamiento de los drones del enjambre para garantizar una navegación segura sin colisiones entre miembros del mismo. El algoritmo se centra en comprobar si dos o más UAVs se aproximan de tal manera que los perímetros de seguridad virtuales creados alrededor de las aeronaves se solapan, y en caso positivo, aplicar acciones de control para evitar la colisión.
 - La segunda parte consiste en un método de detección y evitación de obstáculos basados en sistemas de visión y percepción del entorno que, de manera descentralizada, permite a cada UAV capturar e interpretar información del entorno para detectar obstáculos, tanto estáticos como dinámicos, y establecer maniobras de evasión a través de nuevas rutas seguras.
- Los resultados de este capítulo muestran una arquitectura redundante basada en estos dos métodos, que permiten en tiempo real llevar a cabo maniobras de evasión que garanticen la navegación coordinada, sin supervisión y libre de colisiones del enjambre de UAVs. La acción combinada de ambos métodos permite a cada agente del enjambre poder navegar sin supervisión por el entorno y, adaptar sus movimientos y rutas a situaciones cambiantes como las que se pueden generar en entornos dinámicos como son los incendios.

- **Capítulo 5: Sistema de toma de decisiones inteligente para navegación autónoma de UAVs**

En este capítulo se presenta un método de AI basado en técnicas de aprendizaje por refuerzo (Reinforcement Learning (RL)), que dota a los drones de la capacidad de toma de decisión para elegir las acciones a acometer que aseguren la navegación autónoma del UAV, entre dos ubicaciones concretas, en diferentes entornos a través de rutas óptimas. La irrupción de estos novedosos métodos abre una potente línea de investigación en la que poder establecer modelos basados en RL que doten al UAV de una AI con la que poder acometer tareas de manera autónoma en diferentes entornos y situaciones.

- **Capítulo 6: Conclusiones y Trabajos Futuros**

Resume la conclusión final de los algoritmos propuestos para el sistema de navegación autónoma del enjambre de UAVs, la adaptación y uso de dichos resultados a la empresa Drone Hopper y, presenta las posibilidades de los futuros trabajos.

Capítulo 2

Estado del Arte

2.1. Vehículos Aéreos No Tripulados

Los UAVs son aquellas aeronaves que navegan sin la presencia de un piloto a bordo. Los UAVs pueden ser dirigidos por un piloto desde tierra, bien a través de un sistema de mando por radiofrecuencia (RPAS), o bien guiados de manera automática a través de una estación de control de tierra (Ground Control Station (GCS)), pero también pueden realizar navegación autónoma sin supervisión, empleando sensores embarcados en el mismo. Aunque esta terminología no se ciñe únicamente a la aeronave, si no que incluye todo el software de apoyo, incluyendo sensores, controladoras de vuelo o autopilotos, estaciones terrestres, sistemas de comunicación e interfaces de usuario (58).¹

Según el principio de vuelo, los UAVs se pueden clasificar en dos tipos, según la necesidad de un rotor o sistema de empuje para volar o, si por el contrario su navegación depende de la fuerza de flotación. En caso de que el UAV tenga una masa sustancial y, por tanto, necesite de un sistema de empuje, se dividen en UAVs de tipo ala o UAVs de tipo rotor, mientras que aquellos UAVs que sean más livianos del aire, son conocidos como dirigibles o globos. Dentro de los primeros se puede establecer un escalón más, así pues, los de ala pueden dividirse en tres grupos: ala fija, ala voladora y ala flotante, mientras que los de tipo rotor se diferencian en helicópteros o multirrotores, siendo estos últimos clasificables en función de la

¹Publicaciones del autor relacionadas con el capítulo:

- Al-Kaff, A.; Madridano, Á.; Campos, S.; García, F.; Martín, D.; de la Escalera, A. Emergency Support Unmanned Aerial Vehicle for Forest Fire Surveillance. *Electronics* 2020, 9, 260 (1).
- Madridano, Á.; Al-Kaff, A.; Martín, D.; de la Escalera, A.A. 3D Trajectory Planning Method for UAVs Swarm in Building Emergencies. *Sensors* 2020, 20, 642 (5).

cantidad de rotores que compongan el sistema de propulsión (59), quedando todo ello reflejado claramente en la Figura 2.1.

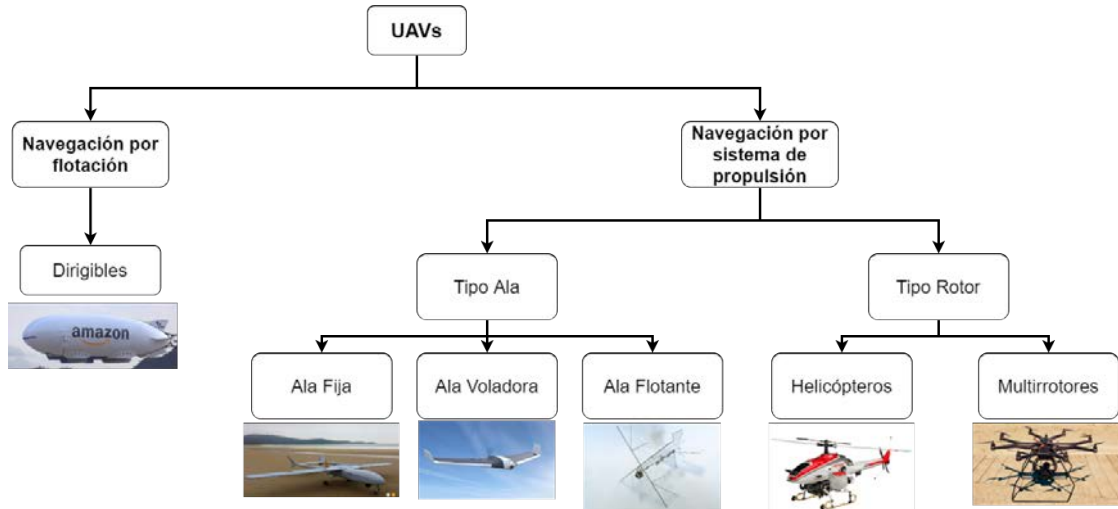


Figura 2.1: Clasificación de UAVs según principio de vuelo.

Existen otros aspectos que permiten establecer otras clasificaciones de los UAVs como puede ser la masa de la aeronave, su tamaño, la capacidad de operación, tanto en altitud como en alcance, el peso máximo al despegue o una combinación de varias (60; 61).

A nivel legislativo, un aspecto clave ha sido la masa de la aeronave, ya que drones con una masa menor a 250 *gramos* presenta una libertad de operación más flexible que drones de mayor masa. Aunque, bien es cierto que esta limitación de masa provoca que sea prácticamente imposible encontrar drones profesionales que operen en este rango de peso, principalmente, por dos razones. La primera porque la carga de pago embarcada en la aeronave es muy limitada o prácticamente nula, lo cual reduce por completo las posibilidades de operar con este tipo de dron en un ámbito profesional y, la segunda, es que el peso de la aeronave está ligado a la autonomía de operación, es decir, aquellos UAVs de pequeño tamaño y peso ven reducida su autonomía al tener limitadas las fuentes de potencia que permitan el vuelo de la aeronave (62).

Aunque los drones nacieron como una tecnología diseñada y orientada al ámbito militar y de defensa (63; 64), su incorporación al mundo civil se ha visto incrementada en los últimos años, lo cual es palpable si se analizan aspectos como el número de publicaciones y temas de investigación relacionadas con esta tecnología a lo largo del siglo XXI (59; 65; 66; 67; 68). Parte de la irrupción de esta tecnología está ligada a los avances y desarrollos en el campo de la microelectrónica y la mejora en los sistemas de procesamiento, lo cual ha provocado, en primer

lugar, la aparición de UAVs con mayor aplicabilidad y maniobrabilidad, alejándose esta tecnología de grandes aeronaves tácticas no tripuladas diseñadas para el sector militar y, en segundo lugar, el desarrollo de herramientas potentes a un precio competitivo.

El crecimiento del mercado y aplicabilidad de los UAVs también ha evolucionado desde el punto de vista del tipo de drones utilizados para dichas aplicaciones. Si bien, al comienzo de la tecnología proliferaban los aviones de ala fija y los helicópteros, en la última década han sido los multirrotores los que han experimentado un gran crecimiento, hasta suponer el 50 % de la cuota de mercado del sector. Uno de los factores que ha afectado a este aumento del uso de los multirrotores es su capacidad para realizar vuelos estables a baja velocidad e, incluso, poder realizar vuelos estacionarios sobre una ubicación concreta, lo cual es recomendable en aplicaciones y servicios relacionados con la captura y toma de datos con precisión o aplicaciones como la agricultura de precisión o la entrega de productos (69).

Por tanto, los multirrotores se han convertido en los drones más empleados en la actualidad debido a sus características y versatilidad para realizar diferentes tipos de tareas. Los avances en el control de estabilidad y en los autopilotos, ligados al desarrollo de nuevos tipos de baterías y motores sin escobillas, han permitido mejorar la construcción y el manejo de este tipo de aeronaves, evitando problemas iniciales de inestabilidad, que provocaban una elevada carga de trabajo de los pilotos en tierra (70). Junto a estos aspectos, las mejoras en el campo de los multirrotores han generado un aumento tanto de la autonomía como de la capacidad de carga de este tipo de aeronaves, aunque sin alcanzar una solución equilibrada ni un avance tecnológico que permita el uso de este tipo de aeronaves en grandes intervalos de tiempo y con una carga de pago considerable.

Dentro del campo de los multirrotores, existe un amplio espectro de aeronaves diferenciadas, principalmente, por el número de rotores y la configuración de estos. De entre todos, el más ampliamente empleado, por su simplicidad dinámica y una robustez aceptable (71), es el UAV de cuatro rotores o quad-cóptero. Aunque en la actualidad, y ante el creciente mercado de los drones profesionales, son los UAVs formados por seis y ocho rotores, hexa-cópteros y octa-cópteros, los que están adquiriendo mayor cuota de mercado al tener, en primer lugar, la capacidad de poder llevar una mayor carga útil y disponer de mayor autonomía, siendo idóneos para tareas como la fotografía o la radiodifusión (72) y, en segundo lugar, por poseer mayor robustez, ya que la pérdida de un rotor no está ligada a la pérdida total del control de la aeronave.

A parte de los ya citados, los drones multirrotor pueden estar formados, normalmente, por tres, cuatro, seis y ocho rotores, que en todos los casos pueden distribuirse de distintas formas, creando multirrotores con el mismo número de rotores pero con diferente configuración (73), tal y como se muestra en la Figura 2.2.

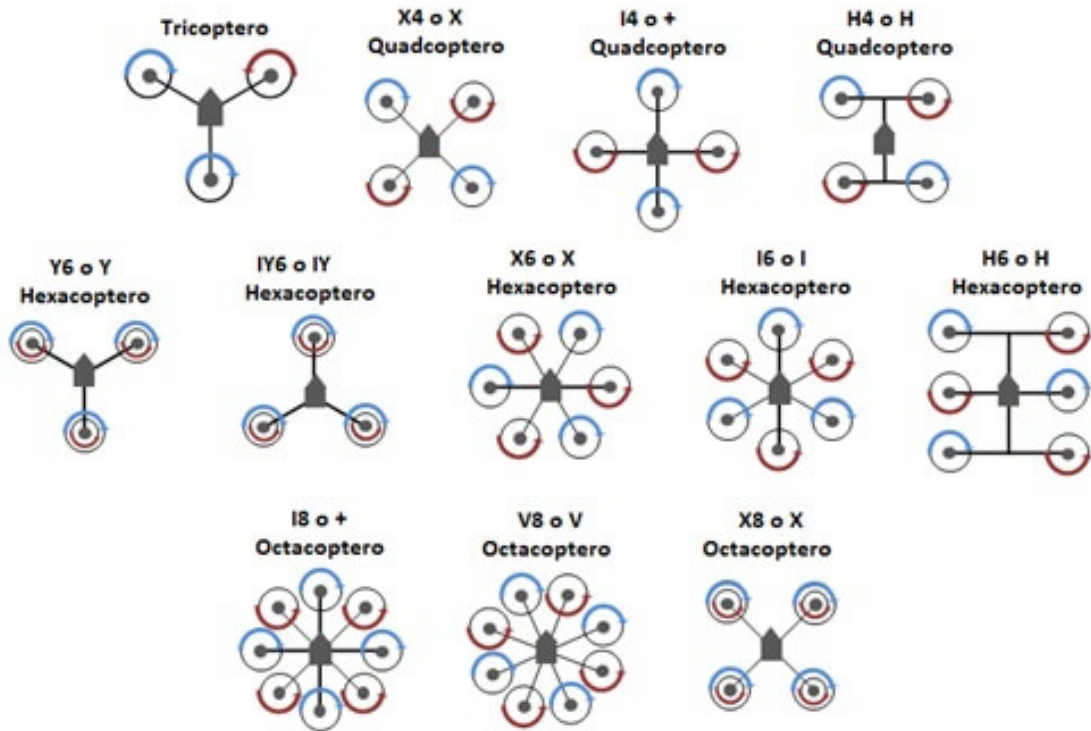


Figura 2.2: Tipos de multirrotores y configuraciones (12).

Cada tipo de multirrotor y configuración dota a los drones diferentes características, provocando que la aplicabilidad y maniobrabilidad de los distintos modelos cambie en función de dichas características:

- Tri-cóptero:** Presenta la ventaja de ser el multirrotor con menor número de motores, lo que reduce su costo y, además, presenta la ventaja de ser aquel con el ángulo más amplio entre los motores (120°), lo cual permite poseer rangos de visión sin hélices mayores, siendo la toma de vídeos e imágenes una aplicación para este tipo de aeronaves (73). Por norma general, la configuración más empleada es aquella con los motores dispuestos en forma de *Y*, aunque en ocasiones, pueden establecer configuración en forma de *T*, con los dos motores delanteros situados a 180 . Aunque es una configuración empleada actualmente, es altamente inestable comparado con el resto de multirrotores. El giro a lo largo del eje *Z* o, más comúnmente conocido como guiñada, suele conseguirse mediante la inclinación lateral del motor trasero.
- Quad-cóptero:** Formado por cuatro rotores, cuya configuración principal es en *X*, con los motores a 90 y, sentidos de giro inverso 2 a 2, dando mayor

estabilidad a la aeronave y, permitiendo el control de la guiñada. Existen otras dos configuraciones más, como son la tipo H y la tipo $+$. Las principales ventajas que presenta este tipo de multirrotores es su buena estabilidad y control, junto con un diseño más sencillo y su menor coste de adquisición al disponer de menos elementos mecánicos.

- **Hexa-cóptero:** Formado por seis rotores, su principal configuración es en X , al igual que los quad-cópteros, pero al disponer de más motores, consigue tener más poder de elevación y, con ello, poder tener una carga de pago mayor que los tipos de multirrotores detallados anteriormente. Además, este tipo de aeronave presenta la ventaja de que la pérdida de, al menos, un motor no lleva a la pérdida total del control de la aeronave y, por tanto, se está ante un sistema más robusto capaz de realizar maniobras de aterrizaje ante esta situación de emergencia.

Como se observa en la Figura 2.2, existen hasta cuatro configuraciones diferentes para este tipo de multirrotores. La configuración X y la configuración I o $+$ se asemejan bastante, siendo la configuración en $+$ la más empleada para vuelos acrobáticos, por poder asemejar su vuelo al de un avión convencional (73). Las dos posibles configuraciones en Y asemejan este modelo al tri-cóptero, con la diferencia de que son dos los motores colocados en cada brazo, lo cual dota al UAV de mayor equilibrio de fuerzas y un control de guiñada más sencillo logrado por diferencia de revoluciones entre los motores que giran en direcciones opuestas. Esta configuración dota al dron de mayor compactibilidad, pero le resta eficiencia.

- **Octa-cóptero:** Por último, se encuentran los drones formados por 8 rotores. Su principal ventaja es que permite la pérdida de hasta dos motores para poder realizar un aterrizaje seguro, lo cual provoca que, al igual que los hexa-cópteros, sean las principales configuraciones empleadas para drones profesionales, equipados normalmente con potentes y costosas cargas de pago. Al igual que en los hexa-cópteros, se dan hasta 3 configuraciones diferentes de este tipo de multirrotores, sin que exista una diferencia sustancial entre las características de cada configuración de este tipo, comparada con los drones de 6 motores.

La versatilidad que presentan estos tipos de sistemas, unidas a todas las características y ventajas comunes con el resto de UAVs, han provocado que sean muchas las aplicaciones y servicios realizados, sobre todo, en el ámbito civil con este tipo de tecnología (74; 75). Entre los principales campos de aplicación de los UAVs se encuentra la topografía y la cartografía (76; 77; 78; 79), labores de vigilancia y monitorización (80; 81; 82; 83; 84), operaciones de búsqueda y rescate (85; 86; 87; 88; 89), inspección de infraestructuras (90; 91; 92; 93), operaciones

en emergencias (94; 95; 96; 97; 98; 1), agricultura de precisión (99; 100; 101), o tareas de manipulación, transporte y entrega de mercancías (102; 103; 104).

A pesar de esta alta aplicabilidad y de la consolidación de este mercado, siguen existiendo desafíos y tendencias emergentes dentro del mundo de los UAVs, como pueden ser, a nivel civil, el incremento de sistemas formados por múltiples UAVs, en los que cada elemento tenga conciencia del grupo o el aumento de la autonomía de vuelo de las aeronaves, lo cual iría ligado a un aumento de la capacidad de operación en las misiones (105). En la actualidad, y dejando al margen a los UAVs de largo alcance empleados en el ámbito militar, los drones profesionales utilizados en el ámbito civil son alimentados con baterías eléctricas, lo cual, limita el tiempo de uso de estas aeronaves, mientras que una pequeña minoría emplea motores de combustión o sistemas térmicos alternativos para implementar los sistemas de propulsión (69).

Aunque el sector de las baterías ha permitido mejorar la autonomía de los UAVs, sigue existiendo una limitación en este punto, ya que, aunque existen cada vez baterías de mayor capacidad, también su peso es mayor, lo cual provoca que los drones cuenten con un mayor peso al despegue, limitando así su autonomía del vuelo. Por tanto, es difícil encontrar el punto de equilibrio en el que el aumento de peso debido al sistema de alimentación compense en términos de autonomía. Por este motivo, los avances en el sector de los UAVs se están dirigiendo hacia la incorporación de sistemas de alimentación térmicos que permitan ganar autonomía a los drones y, a su vez, aumentar la capacidad de los mismos para levantar grandes cargas.

En este aspecto, empresas como Drone Hopper S.L. (106), ofrece soluciones aeronáuticas a través del desarrollo y la fabricación de drones con prestaciones industriales que superan las actuales limitaciones en autonomía de vuelo y carga de pago de las aeronaves no tripuladas que operan en el mercado.

Drone Hopper es una empresa dedicada al diseño y fabricación de drones multirrotor de gran potencia, cuya meta principal se centra en el proyecto WILD HOPPER (13) y, en el desarrollo de tecnologías patentadas para crear una plataforma aérea capaz de transportar grandes cargas, de manera segura y fiable, junto a un sistema de liberación eficaz de las mismas.

WILD HOPPER es una plataforma de gran tamaño, tal y como se muestra en la Figura 2.3, capaz de transportar 600 *litros*, concebida como una herramienta eficaz en la lucha contra los incendios forestales. Con esta carga útil, la plataforma supera las limitaciones de otros UAVs, cuyo uso en la lucha contra incendios se limita a tareas de monitorización y vigilancia. Esta herramienta, incorpora tecnología desarrollada por la empresa Drone Hopper para mejorar la eficacia en la lucha contra el fuego, mediante el uso de drones de altas capacidades.

Las capacidades del WILD HOPPER complementan a los medios aéreos em-



Figura 2.3: Plataforma WILD HOPPER (13).

pleados actualmente en labores de extinción y supera sus principales limitaciones, especialmente, la de poder abarcar operaciones nocturnas, lo cual amplía el horario de operación en un incendio forestal y ayuda a reducir considerablemente la duración de los incendios forestales, al permitir un apoyo aéreo continuo.

Como se ha indicado, el WILD HOPPER incorpora otras tecnologías desarrolladas por la empresa, como es un sistema de liberación de líquidos de gran precisión, que se deriva de las capacidades dinámicas del multirroto. Eso, junto con un sistema patentado de nebulización por chorro de agua (107), produce una eficiencia alta en comparación con los medios de extinción aéreos tradicionales.

El WILD HOPPER es una plataforma menos costosa que los medios aéreos convencionales, lo cual permite a organismos públicos regionales, de pequeño y mediano tamaño, ser propietarios de recursos y herramientas eficaces de respuesta rápida. Al ser máquinas menos complejas, el coste de la propiedad se reduce de manera considerable.

Por último, la plataforma WILDHOPPER es una evolución natural de desarrollos anteriores de la empresa y, forma parte de una familia de aeronaves que comparten tecnologías. En esta familia se incluyen tanto el dron AGROHOPPER, con una capacidad de 60-80 *litros*, lo cual lo orienta a labores de agricultura, y el URBANHOPPER, un avión teledirigido de 150 *litros* de carga, destinado al control de los incendios en escenarios urbanos, reduciendo en gran medida el tiempo de desarrollo y los riesgos de los mismos.

2.2. Metodologías Actuales Aplicadas a la Cooperación en Enjambres

De cara a lograr una cooperación inteligente de un enjambre de UAVs, para acometer, dentro de un mismo espacio de trabajo, un conjunto de tareas de manera autónoma, cooperativa y segura es necesario conocer el estado de la tecnología actual en el campo de los enjambres de UAVs.

En primer lugar, es importante aclarar que los enjambres de vehículos autónomos y, en concreto, de los UAVs, forman parte de un campo tecnológico mayor conocido, en el ámbito de la robótica y los vehículos autónomos, como los sistemas multi-robot (MRS).

En los últimos años se ha extendido el uso de los MRS, formados tanto por UAVs como por Vehículos Terrestres No Tripulados (Unmanned Ground Vehicles (UGVs)), ganando versatilidad y robustez en su operación (108). La posibilidad de utilizar equipos robóticos heterogéneos permite abordar, de forma autónoma, un amplio espectro de tareas con diferentes características en un mismo entorno y de manera simultánea.

Los avances en el campo de la electrónica han mejorado características cruciales, tanto de los UAVs como de los UGVs, como son la autonomía o el peso, junto con una considerable reducción de los costes de producción y adquisición. Esto, junto con la amplia variedad de vehículos disponibles en ambas clases, ha llevado al aumento exponencial en el uso de estos vehículos en un gran conjunto de aplicaciones y servicios. Además, la posibilidad de emplear MRS de manera autónoma y simultánea mejora la eficiencia de estos vehículos en misiones en las que el tiempo de respuesta y la accesibilidad a diferentes áreas de diversas características se establecen como factores cruciales para el éxito de la misma como, por ejemplo, misiones de búsqueda y rescate, emergencias, monitoreo de grandes superficies o rastreo de múltiples objetivos.

Los MRS incluyen a todos aquellos grupos formados por dos o más robots que comparten el mismo espacio de trabajo. Este concepto general, abarca los brazos robóticos industriales, los robots humanoides, sistemas móviles terrestres y aéreos y vehículos autónomos.

Si bien esta tesis doctoral se enfoca en el desarrollo de métodos y algoritmos destinados a permitir un funcionamiento cooperativo, inteligente y autónomo de enjambres de UAVs y, aunque el presente estado del arte ponga su énfasis en la situación actual de esta tecnología concreta, es necesario estudiar y analizar un conjunto de conceptos importantes dentro del campo de aplicación de esta tecnología, que permiten, en primer lugar, establecer clasificaciones en los MRS y, en segundo lugar, entender su terminología.

En el campo de la robótica existen múltiples términos como sistemas multi-robot, sistemas multi-agente, enjambres robóticos o redes de sensores que generalmente se usan indistintamente para referirse a grupos formados por más de un robot trabajando de forma coordinada en un misma área (109). Si bien, este concepto general se mantiene en cada subgrupo, cada uno de los términos presenta unas características particulares:

1. **Sistemas multi-robot:** Incluye todos los sistemas que contienen varios robots. Por lo tanto, un enjambre robótico está incluido dentro de un MRS, pero no todos los MRS son enjambres.
2. **Sistemas multi-agente:** Este término no pertenece solo a la robótica y, en general, se refiere a un sistema compuesto por múltiples agentes inteligentes capaces de interactuar entre sí. Por este motivo, áreas como la informática, la biología, la psicología o la economía también presentan investigaciones en este campo.
3. **Enjambres robóticos:** Este término incluye algunos aspectos esenciales que distinguen al enjambre, como son la escalabilidad, las comunicaciones entre robots y la ventaja del conjunto sobre la individualidad. Aunque existen otros estudios que incluyen aspectos como la autonomía para relacionarse con el medio o la homogeneidad de todos los elementos que forman el enjambre (110), siendo los tres primeros comunes en la mayoría de los estudios.
4. **Redes de sensores:** Este término está relacionado con un conjunto de sensores móviles que interactúan el uno con el otro. Por tanto, cualquiera de los grupos descritos anteriormente podría ser en sí mismo una red de sensores.

Otro aspecto importante dentro de los MRS y, que tiene especial importancia en aspectos como la planificación de trayectoria, es la arquitectura de control y toma de decisiones del MRS. Así pues, se establecen dos tipos principales de arquitecturas:

1. **Arquitectura centralizada:** se caracteriza por tener dentro del MRS un solo elemento o nodo encargado de recolectar toda la información, procesarla y establecer el conjunto de acciones o decisiones a realizar (111; 112; 113; 114). Por tanto, la principal característica de estos sistemas es la capacidad de tener una visión global de todo el sistema en un solo agente y poder establecer planes globales óptimos. Esto facilita el proceso de toma de decisiones, ya que existe un solo agente que, con toda la información, comunica a los diferentes miembros del sistema las acciones a realizar teniendo el control de todos los movimientos y decisiones del sistema.

Por el contrario, presenta desventajas como las limitaciones de los sistemas de comunicación. La arquitectura centralizada se caracteriza por un sistema de comunicación esencial, de manera que el agente central recibe información del resto de elementos del sistema y comunica las decisiones o acciones a tomar. Por lo tanto, el radio de uso de un MRS con arquitectura centralizada está limitado a la distancia permitida por los sistemas de comunicación. Además, los anchos de banda de las comunicaciones pueden establecer una limitación o desventaja en cuanto a la escalabilidad del MRS. Una arquitectura centralizada reduce la robustez del sistema, limitando el éxito de la misión al mantenimiento de ese agente central, cuya pérdida supone el fin de las tareas del resto de elementos del MRS (115; 116).

2. **Arquitectura descentralizada:** A diferencia del otro tipo de arquitectura, no hay un solo agente o nodo encargado de controlar a todos los miembros del MRS (117; 118; 119; 120; 121). Dentro de este tipo de arquitectura se distinguen:
 - a) **Arquitecturas distribuidas:** Se caracterizan por el hecho de que cada miembro del MRS tiene el poder de tomar decisiones por sí mismo. Si bien, es necesario establecer un sistema de comunicación que permita cierto intercambio de información entre los elementos del sistema para cooperar y lograr objetivos comunes. La pérdida de uno o más elementos no conduce al fracaso de la misión.
 - b) **Arquitecturas jerárquicas:** Caracterizado por establecer un orden local entre los diferentes elementos del sistema. Son arquitecturas complejas, pero presentan buena robustez en cuanto a fallos y autonomía de operación (122).

Una de las características esenciales dentro de los MRS es el tipo de elementos que lo forman. Así, los **MRS homogéneos** consisten en robots o vehículos con características idénticas (123; 124; 125), tal y como se muestra en la Figura 2.4.

Por el contrario, cuando al menos uno de los agentes que forma el MRS presenta características o capacidades diferentes al resto, se dice que los **MRS son heterogéneos** (126; 127; 128; 129), como son los ejemplos recogidos en la Figura 2.5. Se considera que un sistema formado por UAVs de diferentes características es también un MRS heterogéneo, como el mostrado en la Figura 2.5a.

Finalmente, otro aspecto crítico dentro de los MRS es la interacción de los elementos del sistema entre sí, es decir, cómo se relacionan entre agentes por el simple hecho de compartir el mismo entorno. Esto genera la aparición de un comportamiento colectivo, el cual se manifiesta de diferentes formas:

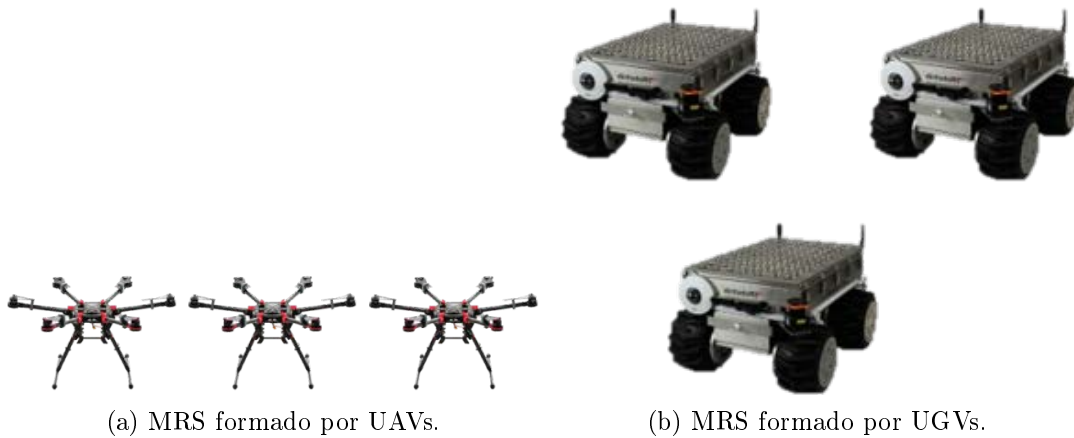


Figura 2.4: Ejemplos de MRS homogéneos.

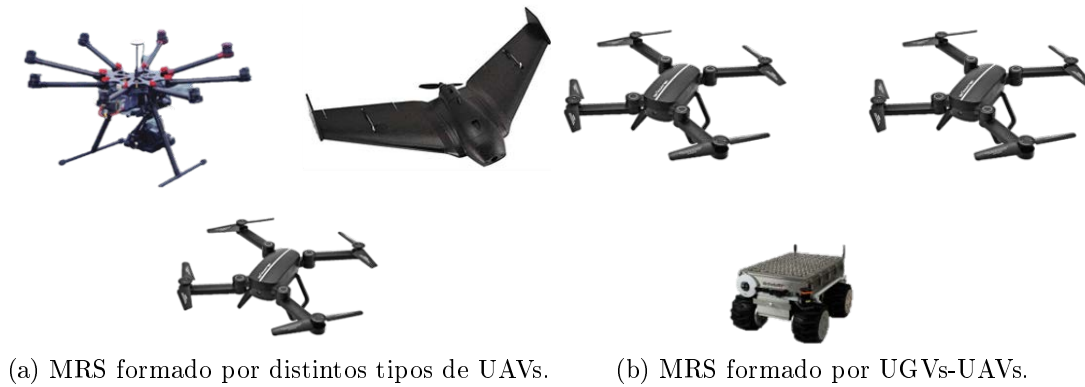


Figura 2.5: Ejemplos de MRS heterogéneos.

- **Indiferencia:** Cada misión de un agente es independiente del resto y puede que no exista relación con el resto de agentes.
- **Cooperación:** Asociación de diferentes agentes cuyo objetivo es un fin o tarea común. Asociado a la cooperación aparece el término *conciencia*, es decir, la propiedad de dichos agentes del MRS de conocer la existencia del resto del sistema.
- **Competencia o antagonismo:** Los objetivos de los agentes del MRS son totalmente incompatibles entre sí.

En este apartado, se han sintetizado un conjunto de conceptos intrínsecos a los MRS que permiten encuadrar, en el campo de la robótica, a los enjambres

de vehículos autónomos y, poner en contexto y comprender conceptos que surjan dentro del análisis de los métodos de planificación de trayectorias de la Sección 2.3.

2.2.1. Aplicaciones en emergencias

El trabajo cooperativo de los MRS presenta ciertas ventajas sobre la actuación individual, lo cual los convierte en una mejor solución para actuar en aplicaciones específicas que requieren de la capacidad de realizar tareas de manera simultánea, aumentando la eficiencia general del sistema y la posibilidad de establecer diferentes configuraciones. Además, el conjunto de habilidades heterogéneas que caracteriza a los MRS, y que garantizan la adaptación de cada agente a la tarea, unido a la capacidad de tolerar fallos o averías, al disponer de diferentes fuentes de información y diferentes elementos dentro del sistema, permiten disponer de una solución tecnológica con un mayor nivel de robustez y versatilidad, características importantes para actuar en entornos dinámicos y adversos.

Uno de los campos donde los MRS han experimentado un desarrollo exponencial, aportando soluciones innovadoras, es el de las emergencias, tales como la lucha contra incendios, labores de búsqueda y rescate o vigilancia (126; 130; 131).

Trabajos como (132) presentan el proyecto denominado SEMFIRE, en el cual un sistema formado por un UAV y un UGV trabajan de manera coordinada para realizar acciones que ayuden a prevenir los incendios forestales. Mientras que la misión principal del UGV es eliminar residuos forestales, el UAV trabaja de manera coordinada para explorar y monitorear áreas donde la acción del UGV es necesaria.

En (126), se presenta una herramienta basada en un MRS que da soporte al personal desplegado en tierra. El trabajo se centra en dotar al MRS de navegación autónoma para que, en caso de pérdidas de comunicación y, por tanto, de la posibilidad de teleoperación, pueda continuar con la misión. El MRS posee la capacidad de navegar sin supervisión para poder recabar información del entorno en áreas donde, a causa de un desastre natural, se han producido pérdidas de comunicación.

Las misiones de búsqueda y rescate han sido otra área donde, los MRS, se han enfocado como una solución tecnológica eficaz aprovechando la ventaja de poder disponer de un conjunto de vehículos con diferentes características para explorar grandes extensiones de terreno. Además, una actuación coordinada de dichos vehículos permite reducir el tiempo necesario para la búsqueda de supervivientes.

En (133), se presenta un algoritmo de búsqueda, para equipos de robots móviles, que permite explorar y recolectar información del entorno para proporcionar puntos de interés de alta prioridad. A partir de esta salida, se establece una división del terreno en regiones, asignando un grado de esfuerzo de exploración a cada una de ellas, de tal manera que se pueda establecer una asignación de áreas a los diferentes robots permitiendo una exploración rápida, eficiente y coordinada.

Finalmente, en (134), se presenta un enjambre formado por un UGV y un UAV, en el que la idea principal es combinar la información recopilada por ambos vehículos de modo que se obtenga una reconstrucción del entorno, con información desde dos puntos de vista diferentes. Se propone un vínculo de comunicaciones entre ambas plataformas de tal forma que tengan un seguimiento mutuo y, puedan mantener dicho enlace de comunicaciones mientras se despliegan por un área de interés.

No solo los MRS formados por estructuras heterogéneas han experimentado un crecimiento en este sector, sino que los enjambres robóticos y, en concreto, los enjambres de UAVs, se han posicionado como una herramienta de alta utilidad para acometer tareas en el ámbito de las emergencias tal y como se analiza a continuación.

En (135), se propone una herramienta basada en un enjambre de UAVs que sirven de apoyo a los equipos de emergencia. La idea es poseer un conjunto de UAVs equipados con diferentes sensores, de tal manera que puedan acceder, de manera autónoma, a áreas remotas, recolectar información e, incluso, a través de tecnología portátil de lucha contra incendios, realizar una mitigación de conatos de incendio.

(136) presenta un trabajo que demuestra que el uso de enjambres robóticos, actuando sin supervisión, es una solución viable y potente en la lucha contra incendios. La idea de este artículo es desarrollar mecanismos de auto-coordinación que permitan un comportamiento efectivo del enjambre en tareas de lucha contra el fuego. Los autores presentan un algoritmo de auto-organización para UAVs capaz de adaptarse a modelos físicos de propagación del fuego, mostrando como resultado una solución eficaz, escalable y tolerante a fallos.

Trabajos como (137) se centra en el uso de enjambres robóticos coordinados para buscar víctimas en áreas de desastres naturales. Se propone un algoritmo heurístico que resuelve problemas de optimización continua no lineal. Con esta implementación logran que la ubicación de la víctima sea la mejor solución aportada por el algoritmo. Dentro del enjambre, el robot líder o maestro supervisa el enjambre y, en colaboración, intenta llegar a la ubicación de la víctima. Este método presenta buenos resultados en términos de velocidad de convergencia con respecto a otros algoritmos comparados dentro del trabajo.

Proyectos como SWIFTERS (138) emplean un enjambre de drones para la gestión de desastres, realizando tareas que van desde la monitorización de las áreas afectadas hasta la realización de evaluaciones de la situación, el mapeado de áreas de interés para contabilizar los daños en dichas zonas o para realizar búsquedas colaborativas de víctimas. En los resultados del proyecto se presenta una herramienta formada por un enjambre de UAVs capaces de alcanzar ubicaciones de manera individual o establecerse en una formación determinada para realizar tareas de búsqueda colaborativa.

Un área dentro de las emergencias donde los enjambres UAVs se presentan como

una potente herramienta es en las comunicaciones. La posibilidad de emplear estos enjambres como repetidores de señal, creando redes de telecomunicaciones y redes de Internet de las Cosas (Internet of Things (IoT)), en zonas de desastres naturales o emergencias, dónde las comunicaciones habituales no se encuentran disponibles, es un campo de investigación importante, tal y como se recoge en un conjunto de trabajos presentes en la literatura (139; 140; 141).

Por último, trabajos como (142) presenta un enjambre de UAVs destinado al trabajo en casos de emergencias sanitarias. Mediante algoritmos de planificación de trayectorias y de formación líder-seguidor establecen un orden dentro del enjambre de UAVs para acometer tareas en el ámbito de una crisis sanitaria.

Tras el análisis de los MRS y, en concreto, de los enjambres robóticos de UAVs, se observa que el crecimiento de estos sistemas colaborativos está ligado al desarrollo y avance de una herramienta tecnológica caracterizada por ser:

- **Robusta:** Al poder contar con más de un vehículo o elemento en el sistema, en caso de fallo o pérdida de un agente, la tarea puede ser completada, aunque sea de manera parcial.
- **Eficaz:** Al disponer de varios vehículos, se pueden realizar diferentes trabajos de manera simultánea, reduciendo así el tiempo empleado en completar misiones complejas.
- **Autónoma:** El poder trabajar sin supervisión aumenta la aplicabilidad de las áreas de uso de estos sistemas.
- **Versátil:** Poder contar con diferentes elementos equipados con diferentes sensores o que presenten distintas características provoca que se aumente la versatilidad y, por tanto, crezcan las posibilidades de ser empleados en entornos dinámicos.

Estas cuatro principales características, que acompañan a los enjambres de UAVs, dan como resultado una herramienta eficaz para emplear en labores de emergencia y, en concreto, en la lucha contra incendios, los cuales se caracterizan por ser entornos dinámicos, con condiciones rápidamente cambiantes, en las que sistemas robustos y versátiles presentan un valor añadido. Junto a este aspecto, el hecho de poder trabajar de manera autónoma, sin poner en riesgo vidas personales y, trabajar de manera colaborativa mejorando los tiempos de respuesta y, con ello, la eficacia, complementan los motivos por los cuales emplear enjambres de UAVs en labores de extinción y lucha contra el fuego y, los convierte en una tecnología y línea de investigación emergente con un alto grado de aplicabilidad en el mercado de los UAVs.

2.3. Planificación de Trayectorias

En los múltiples campos que cubre la AI, la planificación de trayectorias es sin duda uno de los aspectos estudiado por un amplio número de líneas de investigación. Poder encontrar una solución óptima que permita que, uno o varios vehículos, puedan alcanzar un estado final a partir de un estado inicial, de manera segura y efectiva, es un desafío que se sigue estudiando en la actualidad. El uso, cada vez más extendido de MRS y, en concreto, de enjambres robóticos de UAVs operando sin supervisión y simultánea, en un mismo entorno, ha provocado que la planificación de trayectorias se convierta en un aspecto crucial.

Por tanto, existe la necesidad de desarrollar algoritmos que generen una solución al problema de alcanzar una ubicación específica por parte de cualquier UAV que se mueva libremente por un entorno de trabajo sin intervención ni supervisión humana. En el caso de los enjambres de UAVs, se añade la dificultad de que los vehículos deben ser capaces de alcanzar dicho objetivo sin colisionar con obstáculos presentes en el entorno ni con el resto de agentes que forman parte del enjambre.

No sólo es necesario alcanzar el objetivo de forma segura, sino que, en aquellas aplicaciones y entornos donde la distancia recorrida y el tiempo invertido son factores determinantes, se hace imprescindible una solución final óptima. Por esta razón, muchos de los métodos de planificación de trayectorias se basan en la combinación de diferentes algoritmos que permitan, por un lado, encontrar un conjunto de trayectorias seguras hacia la meta y, por otro, establecer cuál de estas posibilidades es la óptima en términos de distancia recorrida, dinámica del sistema o tiempo empleado en recorrer dicho trayecto. Si bien la planificación de trayectorias presenta dificultades a la hora de generar una solución, su aplicación a los enjambres de UAVs implica considerar una serie de aspectos vinculados a este tipo de vehículos y a la posibilidad de moverse libremente en las 3 dimensiones del espacio.

En el caso de las planificaciones en 2D, su principal limitación en cuanto a la planificación de trayectorias es la imposibilidad de llegar a determinados lugares o acceder a ubicaciones remotas al no poder cruzar o superar determinados obstáculos.

La característica más crucial, desde el punto de vista de la planificación para UAVs, es la posibilidad de modificar su altitud a la hora de evitar obstáculos o áreas restringidas, es decir, la posibilidad de establecer trayectorias en 3D. Así mismo, se deben considerar aspectos como la posibilidad de cubrir grandes áreas en poco tiempo o su facilidad para acceder a lugares remotos. Por otro lado, los UAVs presentan una serie de restricciones a considerar durante la fase de planificación como pueden ser la influencia de las condiciones climáticas, en particular el viento, que puede causar que los UAVs no puedan seguir una ruta prefijada o alcanzar una ubicación determinada con precisión, o el “efecto Downwash”, es decir, la imposi-

bilidad de los UAVs de volar cerca de otro UAV a causa de las perturbaciones en el aire generadas por los rotores de un UAV sobre la dinámica del resto de drones.

La expansión de los UAVs y su uso frecuente dentro de los enjambres ha llevado a la adaptación de métodos clásicos de planificación de rutas inicialmente diseñados para un solo vehículo y, a la aparición de nuevas técnicas que consideran múltiples vehículos en la planificación y permiten planificar, en tiempo real, a medida que los vehículos navegan por el entorno.

Antes de analizar los diferentes métodos presentes en la literatura orientados a la planificación de trayectorias en enjambres de UAVs o, en su defecto a MRS genéricos, pero cuya solución tiene su aplicación a enjambres de UAVs, es conveniente detallar algunos aspectos esenciales. Existe un conjunto de términos que se pueden utilizar de forma indiferente para hablar de planificación, pero que tienen aspectos que permiten su diferenciación. Estos términos son:

- **Planificación de ruta:** Se encuentra relacionado con encontrar una curva continua, no necesariamente suavizada, en el espacio de configuración, que comience desde un punto inicial $X_{inicial}$ y alcance un destino objetivo X_{final} . Dicha curva está formada por un conjunto de segmentos y posiciones intermedias definidas a lo largo de la misma. La solución ligada a este término es una solución en crudo y, por esta razón, en ocasiones, es necesario emplear métodos complementarios para obtener una solución óptima.
- **Planificación de ruta óptima:** Este término introduce el concepto de función de coste, la cual puede estar basada en aspectos como la distancia recorrida o el tiempo, para tratar de encontrar un conjunto de curvas que optimicen esta función de coste.
- **Planificación de trayectoria:** Este término está intrínsecamente ligado al conocimiento de las características dinámicas de los vehículos. Por lo tanto, los métodos de planificación de trayectorias son un paso más en la obtención de rutas óptimas. No sólo se determina por dónde se mueven los agentes que intervienen en la misión, sino que también se establece cómo deben moverse a través de dicha ruta. La planificación de trayectorias se incluye en un problema de planificación cinético y dinámico al considerar velocidades, aceleraciones y restricciones cinético-dinámicas de los vehículos (143).

En la definición de planificación de rutas aparece otro término relevante como es el espacio de configuración. Este espacio de configuración es una herramienta matemática desarrollada para recopilar todas las configuraciones y posiciones del vehículo (144). Este espacio de configuración se divide en dos subconjuntos: el

espacio libre, en el que se encuentran todas las posiciones que el vehículo puede alcanzar, y el espacio de obstáculos, con aquellas posiciones inalcanzables o susceptibles de colisión.

El análisis y comparación de métodos de planificación de rutas deben asentarse sobre una serie de criterios objetivos. Los términos empleados en la planificación de rutas permiten obtener dos tipos de enfoques a la hora de comparar resultados (145):

- **Factibilidad:** Encontrar un camino seguro desde la configuración inicial hasta el objetivo, independientemente de la eficiencia.
- **Optimización:** Generar un plan que permita alcanzar de manera óptima el objetivo.

La planificación de trayectorias es un problema cuya complejidad computacional está ligada a las dimensiones del problema. Por tanto, dependiendo del tipo de vehículo y del tamaño del enjambre, el problema varía considerablemente desde el punto de vista computacional. Este aspecto hace que aquellos métodos que logren reducir el tiempo de cómputo estén mejor posicionados para generar soluciones óptimas a este problema y puedan ser utilizados en una planificación dinámica en tiempo real.

Por otra parte, los métodos de planificación de trayectorias pueden ser considerados completos o no. Siendo completos si siempre que hay una solución posible son capaces de encontrarla y, en caso contrario, informar de la incapacidad para establecer un camino.

En este apartado, los métodos de planificación de trayectorias se dividen en cuatro grandes categorías, tal y como se muestra en la Figura 2.6. Cada categoría se caracteriza por poseer especificaciones y características que les permiten tener ventajas y desventajas para su uso en enjambres de UAVs y, en función, del entorno de trabajo. A continuación, se analiza cada categoría y algoritmos de la misma a partir de trabajos recientes presentes en la literatura.

2.3.1. Métodos basados en descomposición de grafo

La idea de estos métodos es generar una representación del entorno de trabajo a través de un grafo, sobre el cual llevar a cabo la planificación de trayectorias. De esta forma, los mapas del entorno real se modelan con un conjunto de vértices (“V”) y bordes (“E”, *del inglés Edge*). Para ello, es necesario una división previa del entorno en cuadrículas que permita establecer una discriminación entre obstáculos y espacio libre mediante la diferenciación de celdas ocupadas y libres. A partir de esta representación, los vértices o nodos del grafo se sitúan en las celdas libres y

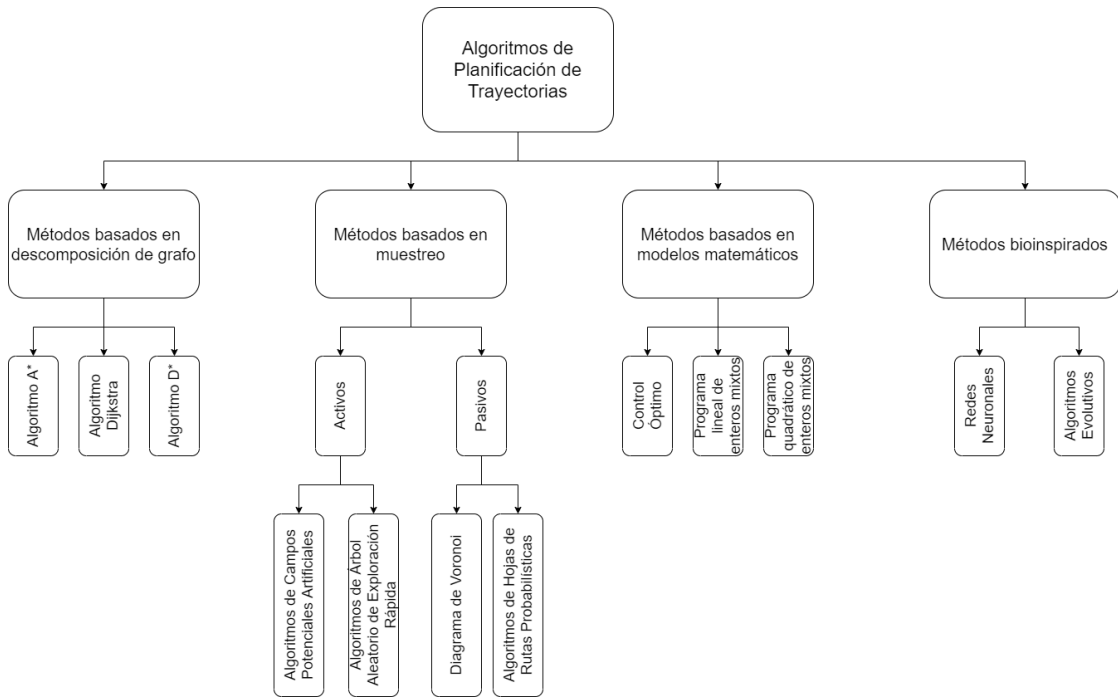


Figura 2.6: Diagrama de métodos de planificación de trayectorias.

se unen entre sí a través de bordes libres de colisiones, generando la estructura previamente definida como grafo, tal y como se muestra en la Figura 2.7.

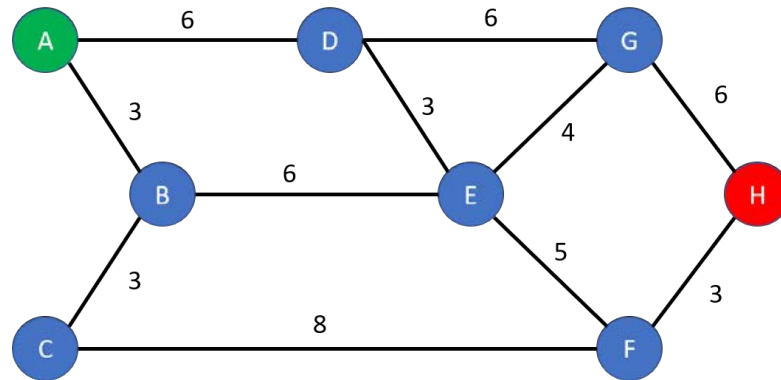


Figura 2.7: Ejemplo de grafo.

Por tanto, la solución al problema de la planificación de trayectoria es encontrar una secuencia de aristas consecutivas que unan un nodo inicial con uno final. Estos algoritmos permiten planificar rutas óptimas mediante la búsqueda de una optimización discreta a través de la descomposición celular del entorno y, minimizando el costo de los pesos de cada nodo y borde que se atraviesa. Por tanto, es

frecuente encontrar desarrollos que combinan, en fases posteriores, estos métodos con otros para encontrar una solución óptima.

Este tipo de método incluye a tres de los más empleados algoritmos clásicos de la planificación de rutas como son Dijkstra, A-Estrella (A-Star (A*)) y A-Estrella Dinámico (Dynamic A-Star (D*)).

2.3.1.1. Algoritmo de Dijkstra

El algoritmo de Dijkstra surgió como una solución al problema de encontrar la ruta más corta entre dos ciudades y, se centra en encontrar el camino más corto entre una ubicación inicial y una final a partir de un grafo del cual ya se conocen los pesos de sus bordes (146).

El procedimiento es el siguiente, partiendo del nodo inicial con valor 0 se recorren todos los nodos adyacentes buscando el peso del borde más bajo. Este procedimiento se va repitiendo mientras existan nodos adyacentes por cubrir y mientras se van comparando pesos y almacenando aquel que optimiza el recorrido. Cuando el nodo final es alcanzado, el procedimiento termina.

Este algoritmo presenta la ventaja de poder encontrar la ruta más corta entre dos ubicaciones, sin embargo, lo consigue a costa de realizar una exploración completa de todo el entorno, lo cual puede suponer un gran problema para aplicaciones en tiempo real si la complejidad del entorno o del problema aumenta, dado que se establece que el costo computacional crece cuadráticamente con el número de nodos del grafo.

En algunos métodos de planificación este algoritmo es empleado en conjunto con otros métodos para obtener una solución óptima (147). En el artículo, el algoritmo Dijkstra se emplea en un segundo nivel de desarrollo para obtener trayectorias libres de colisiones a partir de un grafo obtenido por descomposición triangular. Una vez el grafo es generado, el algoritmo encuentra la solución óptima para varios objetivos. Este artículo también incluye una tercera fase basada en algoritmos genéticos que permiten lograr el suavizado de las trayectorias generadas por el algoritmo Dijkstra. El artículo sirve para comprobar como los tiempos de ejecución aumentan a medida que el entorno se vuelve más complejo, aunque siempre dentro de unos rangos que permiten considerar este desarrollo como una solución para la planificación de rutas para múltiples vehículos en trabajos como exploración, vigilancia o labores agrícolas.

Aunque se trata de un algoritmo clásico, investigaciones recientes han adaptado su desarrollo para poder aplicarlo a problemas de planificación en MRS. En (148) se ha propuesto un algoritmo de planificación completo para un MRS heterogéneo, en el que, a partir de una fase de asignación de tareas, se establece un algoritmo de planificación de rutas basado en Dijkstra para minimizar el tiempo de viaje

entre dos ubicaciones determinadas. Con el uso de este algoritmo se obtiene una solución óptima al problema de la planificación a partir de un entorno previamente dividido en celdas y, la heterogeneidad del MRS empleado permite considerar su solución como válida para su uso en enjambres de UAVs.

En (149), los autores desarrollan un algoritmo de planificación de rutas basado en Dijkstra para generar una solución de múltiples rutas óptimas y coordinadas para robots de inspección de subestaciones. El trabajo desarrollado en dicho artículo, una vez asignada una tarea a un robot, el algoritmo de Dijkstra planifica el camino más corto, sumando a la solución el tiempo de ocupación de cada celda. De esta forma, al establecer rutas para el conjunto de robots, el algoritmo puede verificar, a través de una ventana de tiempo, si existe algún conflicto en la planificación, y si es así, establecer una nueva ruta libre de conflictos. Por lo tanto, a través de la investigación recogida en el artículo, el algoritmo de Dijkstra se posiciona como una posible solución a los problemas de planificación de rutas dinámicas. Esta solución, totalmente adaptable a enjambres de UAVs, permite generar un conjunto de rutas óptimo evitando conflictos entre las trayectorias.

Otro trabajo que centra la planificación de rutas mediante la combinación del algoritmo Dijkstra con otros métodos de planificación es (150). En este caso la solución generada se orienta hacia un proceso de guiado del enjambre basado en una auto-organización jerárquica, sobre un entorno en 2D, en el que el algoritmo Dijkstra permite obtener una solución óptima de rutas para llevar a cabo ese proceso de guiado hacia un objetivo.

Por último, trabajos como (142), incluyen el algoritmo Dijkstra sobre la fase de planificación de rutas globales para un enjambre de UAVs destinado a misiones en crisis sanitarias. Este método genera una trayectoria global óptima considerando la dimensión del sistema físico y las restricciones de aceleración y velocidad de los UAVs.

2.3.1.2. Algoritmo A*

Al igual que ocurre con el algoritmo Dijkstra, el algoritmo A* es un algoritmo completo, es decir, un método de planificación de rutas capaz de encontrar una solución óptima siempre que esta exista (145). Sin embargo, a diferencia del algoritmo de Dijkstra, el algoritmo A* no recorre todo el grafo en busca de esta posible solución, por lo que obtiene mejores resultados en problemas en los que las dimensiones son mayores.

El algoritmo A* reúne un conjunto de procedimientos llamados algoritmos de búsqueda “best-first”, que buscan un conjunto de posibilidades utilizando una función heurística de costo aproximado para ordenar las diferentes alternativas e, inspeccionar las distintas opciones en orden. Específicamente, las funciones heu-

rísticas (H), recogidas en la Ecuación 2.1 y en la Ecuación 2.2, se utilizan para mapear los nodos en el grafo y devolver un valor no negativo que indica la distancia desde el nodo al objetivo. Los criterios de función heurística son:

$$H(\text{Objetivo}) = 0 \quad (2.1)$$

$$H(x) \leq H(y) + d(x, y) \quad (2.2)$$

dónde x e y son nodos adyacentes y $d(x, y)$ es la longitud del borde entre esos dos nodos.

En los problemas de planificación de rutas las siguientes funciones heurísticas son, normalmente, las más empleadas:

1. Distancia Euclídea:

$$H(x_n, y_n) = \sqrt{((x_n - x_g)^2 + (y_n - y_g)^2)} \quad (2.3)$$

2. Distancia de Manhattan:

$$H(x_n, y_n) = |(x_n - x_g)| + |(y_n - y_g)| \quad (2.4)$$

La implementación del algoritmo A* se realiza siguiendo un procedimiento que consiste en que cada nodo conoce la distancia inicial desde el nodo y, la suma de esta distancia junto a la distancia estimada al nodo objetivo. En cada iteración, el algoritmo intenta seleccionar el nodo que tiene más probabilidades de estar en el camino más corto entre el inicio y el objetivo, en definitiva, este algoritmo considera la dirección hacia el nodo final a la hora de realizar la planificación.

Considerar la dirección del nodo final con respecto a la posición de inicio hace que el algoritmo A* sea sustancialmente más rápido que otros algoritmos basados en grafo y, en el peor de los casos, el gasto computacional sea el mismo que en el caso del algoritmo Dijkstra. Por tanto, su principal ventaja es considerar la ubicación del nodo final con respecto al inicial y trabajar en direcciones que podrían resultar fructíferas.

La literatura recoge trabajos en los que se ha implementado el algoritmo A*, en ocasiones con modificaciones, adaptado a la planificación de rutas con múltiples vehículos (151). En dicha investigación, se realiza un cálculo dinámico del valor de los costes asociados a los nodos de la red. A medida que la ruta del vehículo pasa a través de un nodo, el valor de dicho nodo cambia, de modo que el resto de los agentes consideran esa ruta predefinida y generan su ruta en consecuencia. Los resultados obtenidos de las simulaciones muestran cómo el algoritmo A*, junto con la modificación dinámica de los costos, puede generar soluciones óptimas para

la navegación de MRS. Esta solución, que en dicho trabajo se implementa sobre un MRS genérico, es totalmente adaptable a problemas de planificación para enjambres de UAVs y, por tanto, una solución posible al problema de la generación de rutas colaborativas para enjambres.

Otra modificación del algoritmo A^* clásico se presenta en (152). Donde, para un robot re-configurable, se establecen soluciones de planificación de rutas de modo que cada una de las partes del robot posea una trayectoria propia. Esta implementación permite al robot atravesar pasillos estrechos gracias al movimiento coordinado de sus elementos, lo que permite cambios en su morfología. En este caso, se utiliza un enfoque A^* basado en Zig-Zag para lograr una cobertura total del área. En concreto, se definen un conjunto de waypoints con un patrón en Zig-Zag y, se usa el algoritmo A^* para calcular la ruta más corta entre una ubicación inicial y un destino. Este artículo es interesante desde el punto de vista de que la solución presentada permite orientar este algoritmo hacia la resolución de los problemas de navegación en formación de enjambres de UAVs, ya que una adaptación del mismo al espacio en 3D permitiría dotar al enjambre de una solución capaz de generar movimientos coordinados de los agentes a través del entorno, modificando su formación para poder atravesar áreas estructuralmente complejas.

La literatura también recoge trabajos en los que el algoritmo A^* se combina con otro tipo de algoritmos para conseguir una solución al problema de la planificación cooperativa de rutas (153). En el algoritmo referenciado el algoritmo A^* es parte de un desarrollo complejo que busca establecer rutas cooperativas para conseguir una cobertura completa de un terreno. Una vez que se ha determinado y delimitado las áreas dentro del entorno y, se establecen los puntos de ruta a los que debe llegar cada agente, se implementa el algoritmo A^* para optimizar las rutas previamente calculadas, de tal manera que se minimiza la distancia recorrida por los vehículos y, se genera una solución óptima en un problema de cobertura total de un enjambre.

Por último, el algoritmo A^* ha sido empleado en vehículos autónomos que se desplazan a través de entornos viales estructurados (154). Los autores de este artículo presentan una modificación del algoritmo clásico A^* , denominado A^* “Unidad Flexible” (FU- A^*), en el cual, la planificación de rutas se realiza en entornos estructurados teniendo en cuenta tanto obstáculos estáticos como dinámicos. La idea principal de esta línea de investigación consiste en tener una cuadrícula dinámica que se adapte a la velocidad y obstáculos dinámicos presentes en el entorno, permitiendo hacer una predicción de la posición de estos obstáculos en el mapa. De nuevo, esta solución, aunque aplicada en el trabajo a vehículos terrestres, presenta una alta aplicabilidad en la planificación de rutas para enjambres de UAVs, ya que dicha solución tiene en cuenta aspectos dinámicos como la velocidad de los objetos en movimiento, es decir, de cada una de las aeronaves prediciendo la posición en el futuro de cada uno de ellos y estableciendo rutas de movimiento seguro.

2.3.1.3. Algoritmo D*

El algoritmo D* se deriva de la abreviatura de algoritmo A* dinámico (155). El algoritmo D* intenta detectar obstáculos dinámicos y reajustar los pesos de los bordes en tiempo real, de tal manera que se cree un mapa temporal. A continuación, empleando dicho mapa temporal, se establece una navegación segura de los vehículos desde la ubicación actual hasta el destino, minimizando el tiempo a través de un borde no bloqueado.

Al igual que el algoritmo A*, el algoritmo D* emplea funciones heurísticas, las cuales son actualizadas y minimizadas ante la presencia de obstáculos, lo que permite una búsqueda de rutas potente y eficiente.

La literatura recoge trabajos en los que el algoritmo D* es aplicado a problemas de planificación de trayectoria para múltiples vehículos (156). En esta investigación, se implementa un algoritmo D* mejorado (D* Lite) en el que se generan soluciones considerando aspectos como el tamaño del vehículo. A través de un servidor, los agentes proporcionan información de su posicionamiento, provocando que se actualicen las variables de coste del resto de los agentes del enjambre. A medida que los diferentes elementos del enjambre se desplazan, las variables de coste se van actualizando y, las rutas se van ajustando y modificando. Los resultados del trabajo muestran cómo el algoritmo D* Lite puede obtener una solución en menos tiempo que otros métodos de planificación como, por ejemplo, los algoritmos genéticos. Además, con el procesamiento en paralelo, el tiempo necesario para encontrar una solución es menor que en el caso del algoritmo clásico D*. De nuevo, esta solución genérica tiene su adaptación a enjambres de UAVs que trabajan de manera cooperativa, de tal manera que al considerar el movimiento de las aeronaves se reajusten las rutas fijadas y se evitan colisiones entre los agentes del enjambre.

2.3.2. Métodos basados en muestreo

Estos métodos están basados en la realización de un mapeado aleatorio del entorno para intentar encontrar una ruta posible entre dos ubicaciones concretas. Para realizar esta exploración aleatoria es necesario tener una representación matemática que describa el espacio de trabajo. Por ello, dicho muestreo aleatorio se suele realizar en forma de nodos. En cada iteración, se accede a una nueva ubicación del entorno y se determina si corresponde a espacio libre, lo cual lleva a conectar dicho nodo con muestras cercanas generando conexiones y caminos seguros dentro del espacio libre. De esta manera, se consigue generar una estructura similar al grafo descrito anteriormente dentro del entorno de trabajo.

Una clasificación importante de los métodos basados en muestreo es la diferenciación entre activos o pasivos. Los métodos activos se caracterizan por dar como

solución la mejor ruta al objetivo, mientras que los métodos pasivos generan un conjunto de rutas desde el inicio hasta el destino y, es necesario el uso de algoritmos complementarios para determinar cuál de toda la red es el camino óptimo.

Otra característica esencial de los métodos de muestreo aleatorio es que, a pesar de funcionar bien en grandes entornos 3D, no son métodos completos. Esto significa que puede haber una solución posible que conecte una ubicación inicial con una final, pero el algoritmo no logre encontrarla. Esto se debe a que, en ocasiones, el muestreo aleatorio realizado no es suficiente y genera conflictos en situaciones como la que se refleja en la Figura 2.8, y que se conoce como el caso fallido del “Twisty Passageway”. La presencia de espacios estrechos, conexiones de estancias a través de áreas reducidas o zonas de alta densidad de obstáculos pueden provocar la aparición de este problema.

Twisty Passageway – Failure Case

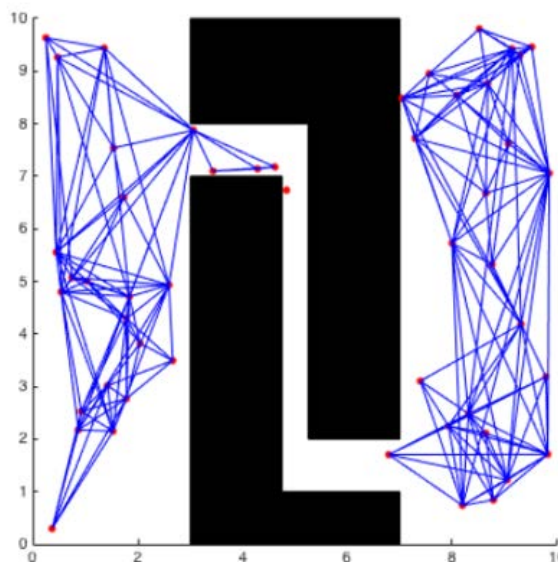


Figura 2.8: Ejemplo de caso fallido en métodos de muestreo aleatorio (14).

Aunque existen técnicas para abordar este problema, como puede ser una mayor densidad de muestreo o un muestreo más denso en áreas cercanas a obstáculos, ninguna garantiza un funcionamiento adecuado en todos los casos posibles. Además, este tipo de técnicas orientadas a mejorar la efectividad de los métodos de muestreo se contraponen a la principal ventaja de los métodos, que no es otra que la capacidad de controlar el gasto computacional en la exploración y generación de rutas a pesar de aumentar las dimensiones del problema.

Finalmente, otro aspecto de los métodos de muestreo es que, en ocasiones, provocan la generación de rutas siguiendo un patrón de Zig-Zag (157; 158), por lo

que en ocasiones es requerido un proceso de suavizado de trayectorias para mejorar el comportamiento dinámico de los sistemas.

2.3.2.1. Hojas de rutas probabilísticas

Las hojas de rutas probabilísticas o PRM es el primero de los algoritmos pasivos incluidos en los métodos basados en muestreo. Es un algoritmo que permite explorar grandes entornos con un menor costo computacional en comparación a los métodos detallados en la sección anterior (145).

La implementación de un algoritmo de PRM es la siguiente: a partir de un espacio de configuración, se toman ubicaciones de forma aleatoria y se comprueba si corresponden a espacio libre u ocupado. En caso de situarse en espacio libre, se buscan nodos cercanos con los que establecer una conexión continua y libre de obstáculos. De esta forma, se explora el entorno y se elabora una red dentro del espacio de configuración sobre la cual se emplean métodos basados en grafo para obtener la solución óptima del conjunto de rutas posibles entre el punto de origen y el de destino.

Aunque ser un método pasivo podría ser una desventaja, en ocasiones, la combinación de algoritmos simples, computacionalmente ligeros, permite obtener una solución óptima con un menor gasto computacional en comparación con emplear únicamente métodos activos. Además, en el caso de los algoritmos de PRM, presenta una ventaja que lo convierte en un algoritmo competitivo dentro del campo de los MRS en general, como es la posibilidad de reutilizar el grafo generado para encontrar el conjunto de trayectorias de todos los agentes del sistema. De esta manera, se podría implementar un grafo más denso, al que tuviera acceso, en este caso el enjambre de UAVs, y de esta manera todos los agentes emplear esta única exploración del entorno para encontrar la solución óptima.

De entre las ventajas de los algoritmos de PRM se encuentra su capacidad de explorar entornos 3D con un menor coste computacional, por lo que son varias las líneas de investigación que han puesto el foco en este tipo de algoritmos para resolver los problemas de planificación de rutas en sistemas formados por múltiples UAVs ya que, comparado con otros métodos de muestreo, permite una exploración más rápida (159).

Existe una línea de investigación orientada hacia este tipo de algoritmos que ha presentado un conjunto de trabajos cuya base para solucionar la planificación de rutas es el uso de algoritmos de PRM (160; 161; 162). Estos trabajos, además de considerar aspectos importantes para los UAVs como es la imposibilidad de los mismos de volar cerca de las perturbaciones generadas por el rotor de otro UAV, incluye aspectos descritos anteriormente, como la necesidad de combinar estos algoritmos con métodos de optimización y suavizado de rutas o, restricciones tem-

porales para adaptar el problema a los enjambres de UAVs, pudiendo así generar el conjunto de trayectorias seguras y óptimas. Entre los resultados de los trabajos, se puede observar cómo se justifica el uso de este tipo de algoritmo para enjambres de UAVs por tratarse de un algoritmo escalable en términos de tiempo de cómputo.

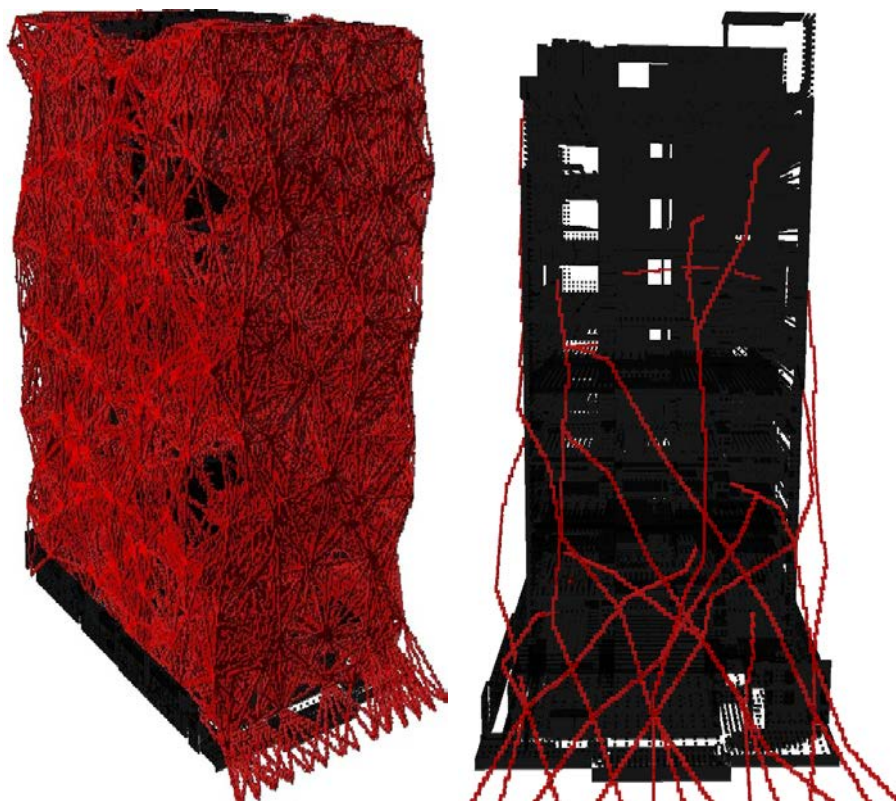
En trabajos como (2), se orienta la planificación de rutas en 3D para un enjambre de UAVs destinado a trabajar en tareas de emergencia en áreas urbanas. Para ello, se utilizan algoritmos de PRM para establecer, con un bajo coste computacional, un conjunto de posibles trayectorias en entornos grandes como el mostrado en la Figura 2.9a. En los resultados de este artículo se puede observar cómo la posibilidad de reutilizar el grafo generado permite controlar el aumento del gasto computacional, al emplear enjambres con un número variable de agentes. De esta manera, se presenta, basado en algoritmos de PRM, un método de planificación de rutas escalable, con aplicabilidad en campos tan cruciales como las emergencias. Además, aunque no es un método completo, su desarrollo junto con algoritmos de optimización clásicos como el A* permite generar un conjunto óptimo de múltiples trayectorias, tal y como se presenta en la Figura 2.9b.

Si bien su principal ventaja lo hace adecuado para problemas relacionados con UAVs y planificación en 3D, trabajos como (5), demuestran que es una herramienta útil para aplicaciones en 2D, en las cuales el tiempo computacional es incluso menor que para las aplicaciones en 3D. El trabajo está relacionado con UAVs y, dicha planificación en 2D incluye un parámetro encargado de fijar la altura de vuelo a cada miembro del enjambre, evitando así posibles colisiones entre miembros del enjambre.

Trabajos como (163) proponen un algoritmo de PRM cinético-dinámico junto con un algoritmo de control predictivo no lineal para la generación de trayectorias óptimas dentro de la planificación de rutas de un enjambre de UAVs. De forma similar, (164) propone un algoritmo de PRM para el seguimiento y búsqueda de trayectorias en entorno urbano para un enjambre de UAVs que se articula bajo en una arquitectura descentralizada y cuyo resultado muestra la eficiencia de este tipo de algoritmo para llevar a cabo misiones autónomas mediante la planificación de rutas para UAVs.

2.3.2.2. Diagramas de Voronoi

El segundo de los algoritmos incluido en los métodos de muestreo pasivos son los diagramas de Voronoi, los cuales son ampliamente utilizados en el campo de la planificación de rutas. Su concepto se basa en generar una conexión topológica que permita dividir el espacio en regiones, teniendo en cuenta la presencia de obstáculos (165). El diagrama está constituido de tal manera que las distancias desde los bordes hasta los obstáculos más cercanos sean las mismas.



(a) Ejemplo de grafo generado por algoritmo de PRM.

(b) Rutas finales planificadas.

Figura 2.9: Algoritmo de PRM aplicado a un edificio de viviendas (2).

El procedimiento para generar el diagrama de Voronoi comienza con la selección de un punto de partida cuyas coordenadas tienen la propiedad de poseer la distancia mínima a los obstáculos cercanos a la misma. A continuación, se calcula el resto de las regiones, las cuales están determinadas por los obstáculos presentes en el entorno. De esta manera, y tal y como se muestra en la Figura 2.10, el espacio de trabajo queda fragmentado en regiones, con todos los obstáculos registrados.

Por tanto, el algoritmo de Voronoi genera un grafo global, como ocurre con el algoritmo de PRM y, al igual que estos algoritmos, la propiedad de ser pasivo conlleva al uso de otros métodos para la obtención de una solución óptima. Por lo general, son los algoritmos A^* o D^* los empleados en combinación con el Diagrama de Voronoi para, a partir de los bordes de las regiones del diagrama, establecer rutas óptimas y seguras.

La literatura recoge trabajos como (15), en el que el diagrama de Voronoi es empleado junto con métodos basados en la teoría de la consistencia para establecer

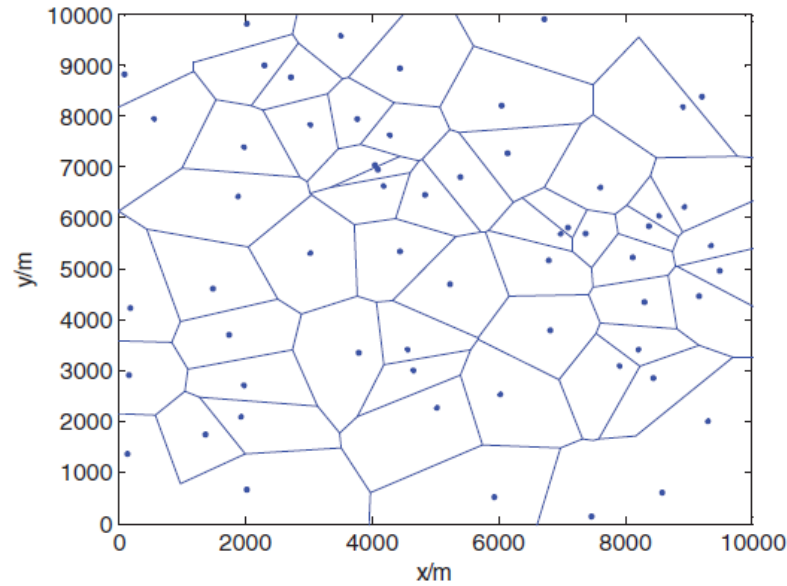


Figura 2.10: Ejemplo de diagrama de Voronoi (15) ©2017 IEEE.

rutas óptimas en un sistema formado por varios UAVs. Este artículo referenciado, se enfoca en resolver la planificación de múltiples rutas para UAVs que deben atacar múltiples objetivos en un entorno estático. El diagrama de Voronoi se establece para representar el gráfico de amenazas y, se emplea como entrada para que los UAVs busquen, de manera cooperativa y a través de la teoría de la consistencia, rutas óptimas. De esta forma, esta combinación de métodos y teorías permite generar una solución al problema de la planificación de rutas múltiples como se representa en la Figura 2.11.

Existen trabajos en los que se ha propuesto un sistema de control para la coordinación de agentes móviles basado en diagramas de Voronoi (166). En el trabajo referenciado, el algoritmo de los diagramas de Voronoi se extiende y es implementado como un diagrama de garantía de potencia, el cual permite habilitar particiones dinámicas de Voronoi. Esta modificación del diagrama de Voronoi clásico permite modificar los pesos en las áreas del espacio de trabajo que se va asignando a los diferentes agentes permitiendo que las rutas fijadas converjan hacia un movimiento coordinado del MRS. Esta planificación genérica, orientada a agentes móviles, es aplicable a problemas de vuelo coordinado en enjambres de UAVs.

Otros trabajos orientan los diagramas de Voronoi a la construcción de un Tesselado Centroidal de Voronoi (Tessellation Centroidal Voronoi (CVT)) que permite el movimiento y montaje de un enjambre de manera autónoma (167). El CVT se caracteriza por el hecho de que los puntos correspondientes a las celdas del diagrama de Voronoi se encuentran ubicados en los centroides de las celdas. El uso

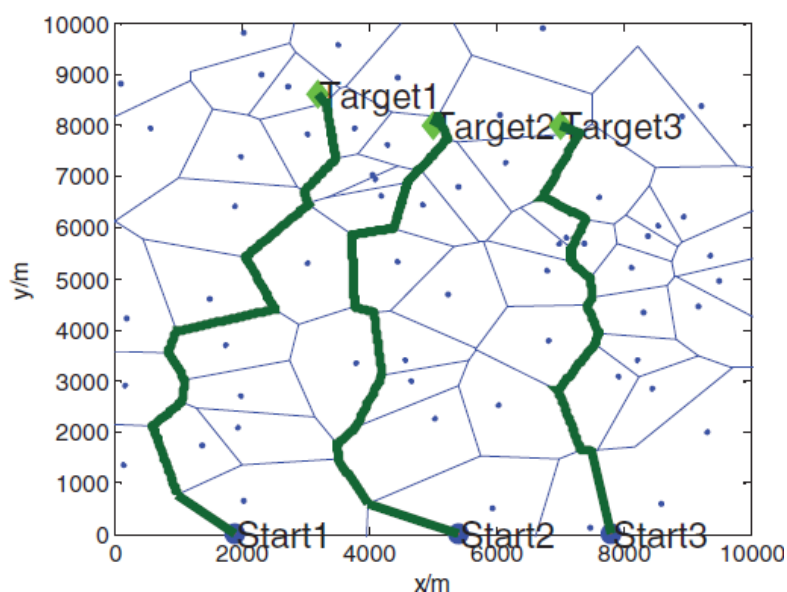


Figura 2.11: Solución al problema de planificación de rutas para múltiples UAVs a través de un diagrama de Voronoi (15) ©2017 IEEE.

de este método permite la planificación de diferentes caminos para cada uno de los agentes del enjambre, permitiendo el auto-ensamblaje del mismo en diferentes configuraciones.

La pasividad de los diagramas de Voronoi provocan que sean empleados en conjunto con otros métodos para poder generar la solución final al problema de planificación de trayectorias óptimas. En (168), el diagrama de Voronoi es empleado en una arquitectura flexible y escalable de planificación y asignación de misiones para un enjambre de UAVs. Esta arquitectura, formada por varias capas, posee un planificador global de misión el cual, haciendo uso de los diagramas de Voronoi, divide el espacio de trabajo en áreas de menores dimensiones en las que los centroides de las mismas corresponden a ubicaciones empleadas para construir las misiones de cada uno de los UAVs.

En trabajos como (169), emplean los diagramas de Voronoi combinados con redes auto-organizadas bio-inspiradas para mejorar la distribución de un enjambre de UAVs dentro de un entorno sin cobertura de Global Positioning System (GPS). En concreto, con esta mejora del algoritmo de Voronoi clásico, no solo se elimina la distribución aleatoria, sino que se establece una localización homogénea a través de un proceso iterativo en el que se examinan continuamente la posiciones relativas de los nodos entre sí, con las paredes y con otros obstáculos para asegurar que la energía invertida se dirija hacia la nueva zona del espacio descubierto, manteniendo la conectividad de la red a pesar de expandir el espacio de trabajo. De esta forma,

se consigue que el enjambre de UAVs se establezca en una distribución que mejore la cobertura del área disminuyendo la energía consumida.

Otra solución presente en la literatura para la planificación de rutas cooperativas de un enjambre de UAVs basado en diagramas de Voronoi es (170). En este caso, se emplea el diagrama de Voronoi para el modelado del entorno, pero a diferencia del algoritmo clásico, en el trabajo se incluyen modificaciones que permiten determinar los costes de los bordes de Voronoi en función de diferentes tipos y fuentes de amenazas y, de esta manera, generar una solución completa en la que se consideren aspectos como posibles impactos con amenazas externas.

En línea con estos trabajos se presenta (171), en el que de nuevo, los diagramas de Voronoi forman parte de la etapa inicial de un método de planificación de trayectorias para enjambres de UAVs. En esta primera etapa, el diagrama de Voronoi es empleado para crear un campo de amenazas y establecer, en primer lugar, el costo total del trayecto.

Finalmente, trabajos como (172), muestran otra aplicación de los diagramas de Voronoi para la planificación de rutas. Consiste en utilizar estos diagramas en tareas de exploración y búsqueda, ya que el grafo generado permite maximizar los caminos resultantes en función de la distancia recorrida. En este artículo, un MRS orientado a tareas agrícolas lleva a cabo un conjunto de procesos a partir de una asignación de tareas basada en la partición del espacio de trabajo usando un diagrama de Voronoi. Aunque la investigación referente a este artículo se orienta a un MRS genérico, su desarrollo es interesante desde el punto de vista de ofrecer una solución para enjambres de UAVs, en la que se maximiza la distancia total recorrida de cara a misiones de cobertura completa de áreas.

2.3.2.3. Árbol Aleatorio de Exploración Rápida

El primero de los métodos activos por muestreo es el Árbol Aleatorio de Exploración Rápida (Rapidly Exploring Random Tree (RRT)). Este algoritmo tiene como objetivo muestrear el entorno considerando las ubicaciones de partida y de destino. El RRT tiene la capacidad de manejar múltiples grados de libertad, lo que permite que sea adecuado para trabajar con robots personales y brazos robóticos, aunque también presenta aplicaciones en MRS.

El procedimiento implementado para crear el grafo, o en este caso árbol, es el siguiente. Tras dividir el espacio de configuración en libre y ocupado, se introducen tanto el nodo de inicio como el final en el espacio libre. A continuación, se crea un nodo aleatorio, el cual debe estar posicionado siempre en el espacio libre, y se comprueba que puede establecer una conexión libre con el nodo inicial. El siguiente paso consiste en comprobar si se puede llegar al nodo final desde este nuevo nodo intermedio. En caso negativo, el proceso se repite, pero en este caso el nuevo nodo

creado aleatoriamente en el espacio libre es conectado al nodo final y, la comprobación se realiza desde esta nueva ubicación hacia la rama creada previamente en el nodo inicial.

De esta forma, en cada iteración, las ramas del árbol procedentes de los nodos inicial y final van creciendo a medida que realiza la exploración del espacio libre, tal y como se observa en la Figura 2.12.

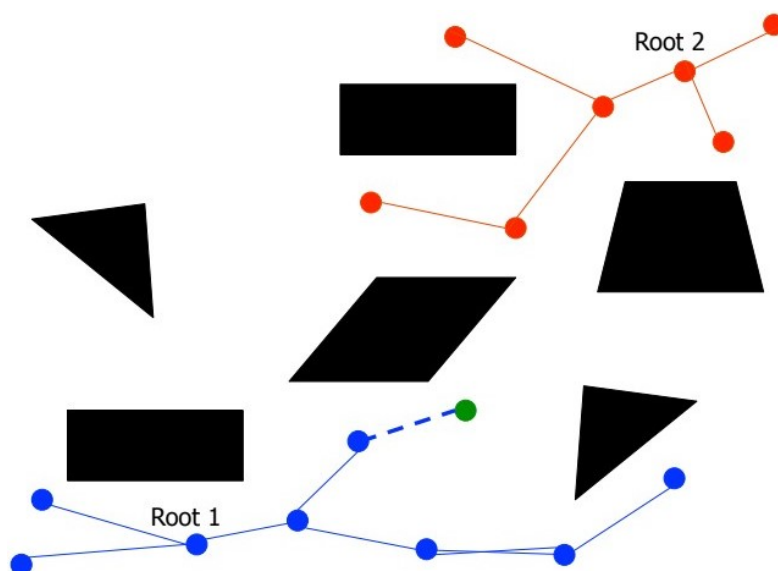


Figura 2.12: Muestreo del entorno con algoritmo de RRT (14).

En cada iteración, el sistema genera una muestra aleatoria en una de las ramas e intenta hacerla crecer hasta que se pueda establecer un puente o conexión entre ambas ramas. En caso de que sea positivo, la ruta es encontrada y el objetivo alcanzado, tal y como se puede ver en la Figura 2.13. En caso contrario, se sigue el procedimiento, haciendo crecer en cada iteración una rama distinta del árbol.

El RRT es un algoritmo completo y eficiente para problemas de planificación de trayectorias con espacios de configuración de alta dimensión, capaz de trabajar con restricciones dinámicas. Los trabajos recientes relacionados con algoritmos de RRT se aplican a la planificación de trayectorias introduciendo mejoras y modificaciones, pero manteniendo las características esenciales del RRT clásico descritas previamente.

El RRT* multidimensional es una de las mejoras presentes en la literatura para llevar a cabo la planificación de rutas dentro de un sistema con múltiples vehículos autónomos (173). La idea de esta investigación es desarrollar un método para mejorar la eficiencia en labores de búsqueda, vigilancia y seguimiento en un área de trabajo. El propósito del RRT* es crear un conjunto de caminos óptimos,

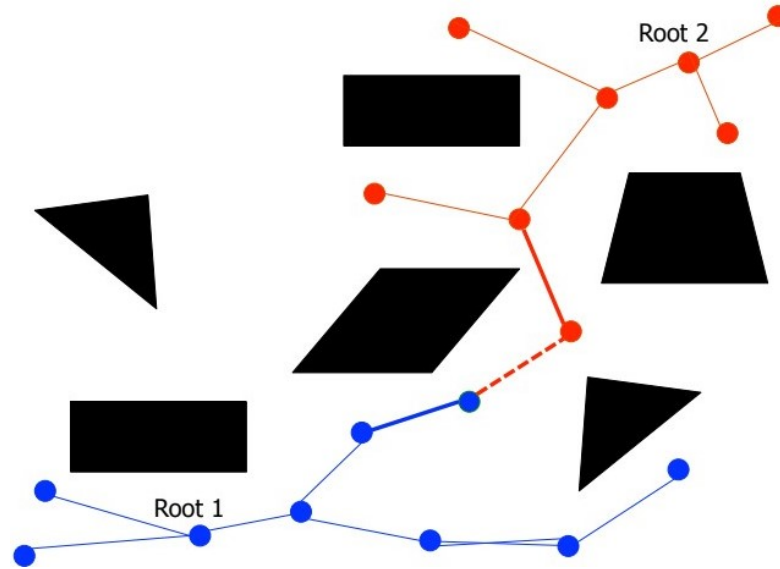


Figura 2.13: Ruta final obtenida por RRT (14).

alcanzables y libres de colisiones a través de los cuales los diferentes vehículos naveguen desde un estado inicial a uno final.

La variante RRT^* está presente en otros trabajos de la literatura, en este caso aplicados a la planificación de rutas en 3D para agentes móviles a partir de la nube de puntos generada por una cámara Kinect V2 (16). En este caso, a partir de la nube de puntos, se establecen qué regiones del entorno están libres de obstáculos y cuáles ocupadas, lo cual puede comprenderse mejor a través de la Figura 2.14, y se introducen dos fases de optimización al algoritmo RRT clásico para minimizar el coste computacional final obtenido a partir del árbol. Las fases introducidas se basan en cambiar la probabilidad de generación de los nodos aleatorios y, son llamadas RRT^* *Goal* y RRT^* *Limits*. Los resultados obtenidos en el artículo referenciado establecen que el RRT^* *Goal* reduce considerablemente el tiempo de búsqueda de soluciones factibles a cambio de un incremento en los recursos computacionales debido a la optimización, mientras que el RRT^* *Limits* permite generar trayectorias con menores costes que un RRT^* estándar.

Otra de las modificaciones es el denominado Árbol Aleatorio de Exploración Rápida discreto (Discrete Rapidly Exploring Random Tree (dRRT)) (17). Esta adaptación permite generar un grafo discreto a medida que se realiza una exploración rápida del espacio de configuraciones. La técnica desarrollada se centra en la búsqueda de caminos dentro del grafo creado en escenarios que implican el acoplamiento de varios robots. Para ello, se realiza una exploración parcial, de tal forma que solo se considera vértice vecino aquel que puede ser visitado en cada paso. De esta manera, al trabajar con árboles locales y reducidos, tal y como se muestra en

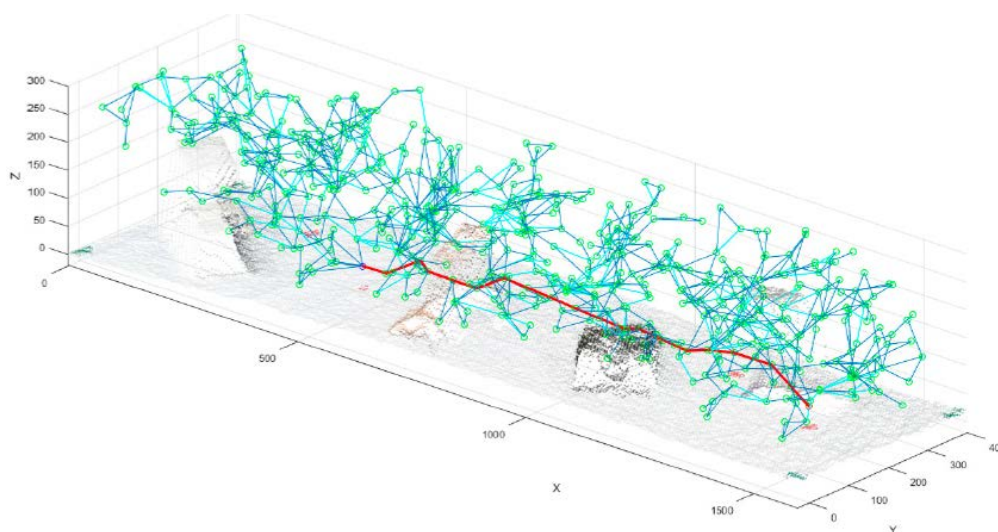


Figura 2.14: Generación de RRT en 3D y obtención de una ruta final sobre una nube de puntos (16).

la Figura 2.15, el dRRT puede explorar el espacio de configuración rápidamente, aunque se trabaje con enjambres de múltiples agentes al realizar planificaciones sobre una pequeña exploración de todo el espacio.

Otra modificación del algoritmo RRT presente en la literatura y empleada en el campo de los UAVs es el denominado RRT-Connect. Este algoritmo combinado con Campos Potenciales Artificiales (Artificial Potential Fields (APFs)) permite generar rutas óptimas para los UAVs mejorando sustancialmente tanto la distancia recorrida como el tiempo de cómputo de un algoritmo RRT clásico, tal y como se recoge en los resultados del trabajo presentado en (174).

Finalmente, trabajos como (175) presenta métodos basados en una extensión del algoritmo RRT clásico para poder generar un planificador global de rutas para un MRS en general y, por tanto, de aplicación a enjambres de UAVs. El trabajo combina la teoría del *Mantenimiento de la Conectividad Generalizada* con una extensión del algoritmo RRT para dar solución al problema de la planificación. Con esta combinación, el planificador global genera las rutas respetando los requisitos de conectividad, garantizando la coordinación entre los agentes del enjambre. La extensión de RRT presentada, se denomina T- RRT y se utiliza para la gestión de problemas relacionados con la planificación en entornos desordenados, con el objetivo de evitar mínimos locales gracias a un mecanismo de auto-adaptación que explora con eficacia los valles.

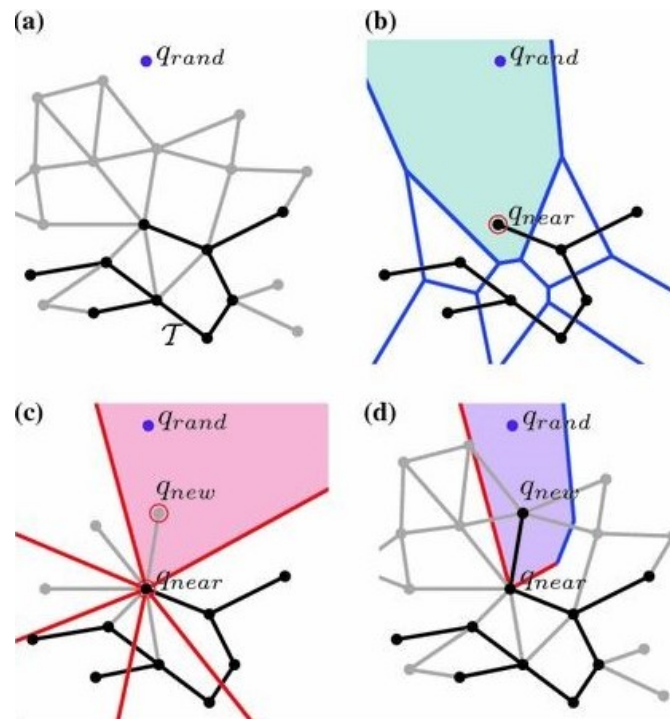


Figura 2.15: Ejemplo de exploración con algoritmo de dRRT (17) ©2016, ©SAGE Publications.

2.3.2.4. Métodos de Campos Potenciales Artificiales

Los algoritmos de Campos Potenciales Artificiales o APFs, se utilizan ampliamente debido a su baja complejidad computacional. Se basan en establecer una función de potencial del espacio de configuración para poder crear una relación entre espacios libres y obstáculos, de tal manera que la función tenga valores altos en las proximidades de los obstáculos y bajos cuando se aleja de los mismos. Además, se establece un mínimo en la función que coincide con la ubicación del objetivo (176).

La primera parte de la implementación consiste en crear esta función potencial, la cual se requiere para definir un campo repulsivo relacionado con los obstáculos presentes en el espacio de configuración y un campo atractivo con un mínimo correspondiente al objetivo, generando una función del potencial que queda reflejado en el entorno como se observa en la Figura 2.16.

Una vez se establece la función del potencial, se hace uso del gradiente para poder guiar a los UAVs desde el punto inicial hasta el final, tal y como se muestra en la Figura 2.17. La ventaja de estos métodos es que son relativamente simples, funcionan en tiempo real y pueden trabajar en problemas de grandes dimensiones.

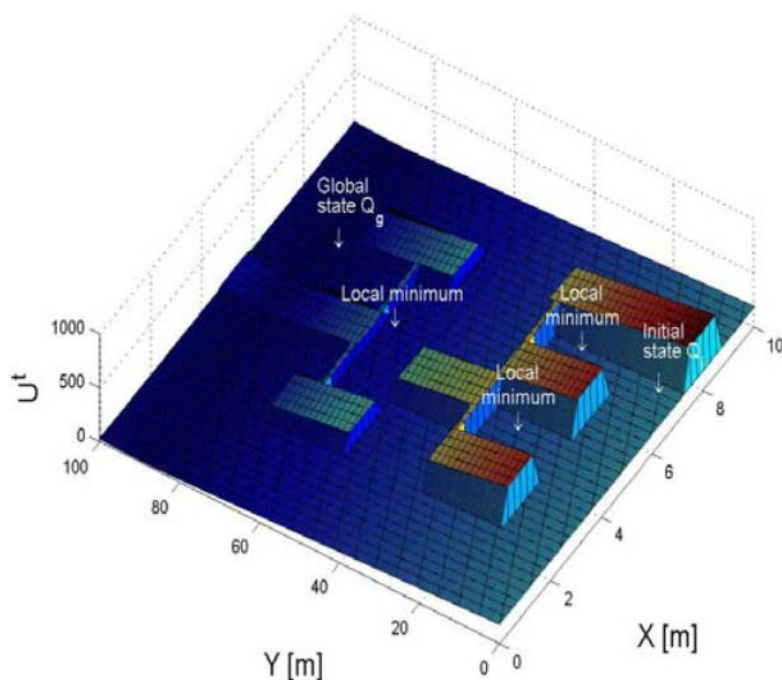


Figura 2.16: Ejemplo de APF (18) ©2017 IEEE.

Sin embargo, presentan la desventaja de que son incompletos y no pueden asegurar el éxito de la misión, debido, principalmente, a que la función del potencial obtenida de la combinación de fuerzas repulsivas y atractivas puede generar mínimos locales. Estos mínimos pueden provocar que los vehículos caigan en uno de ellos y no puedan alcanzar el destino.

En la práctica, es difícil eliminar estos mínimos locales y saber cuándo el algoritmo puede converger y cuando no. Este problema se ha tratado de evitar implementando aspectos como las funciones de navegación o el cálculo de potencial con restricciones.

La literatura recoge trabajos que han implementado algoritmos de APFs para la navegación libre de formaciones de robots móviles con estrategias de líderes-seguidores (177). Para ello, se generan dos campos potenciales, uno atractivo para el destino y otro repulsivo para los obstáculos, que trabajan en conjunto para formar un campo potencial compuesto que permite al MRS alcanzar la meta evitando los obstáculos y moviéndose en formaciones de manera segura y fiable. Aunque la solución trabaja con MRS genéricos, los resultados y la solución entregada en el artículo referenciado es extrapolable al caso de investigación del presente trabajo, como son los enjambres de UAVs.

El algoritmo de APF también tiene su aplicación en entornos dinámicos para sistemas de múltiples UAVs (20). En el trabajo de investigación referenciado se

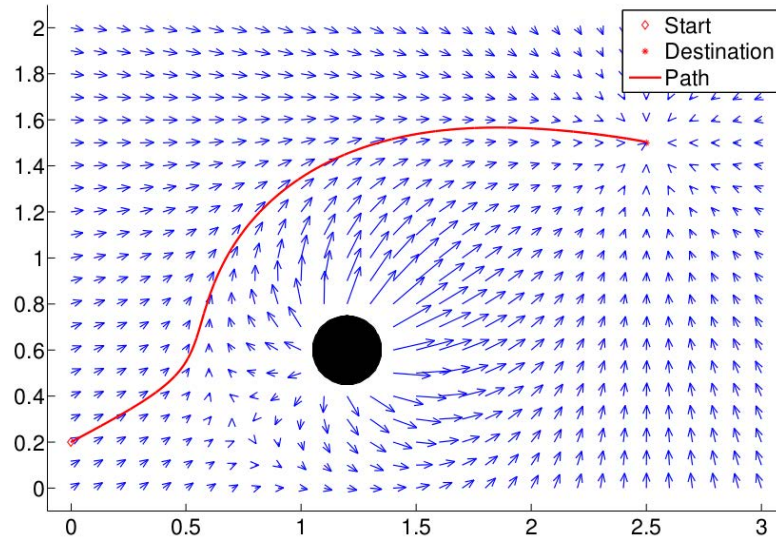


Figura 2.17: Ejemplo de gradiente para navegación de los agentes (19).

desarrolla un método APF con un factor de distancia y una estrategia de salto que permite resolver problemas comunes como la evitación de obstáculos, no solo estáticos, sino también al resto de vehículos presentes en el entorno, los cuales constituyen obstáculos dinámicos. Los resultados de las simulaciones muestran cómo estos algoritmos son válidos para enjambres de UAVs en entornos con presencia de obstáculos, tal y como muestra la Figura 2.18.

En esta línea se presenta un trabajo basado en APF para conseguir que dicho enjambre navegue de manera autónoma y segura por el entorno (178). Para ello, se crea un mapa potencial en 3D a partir de un campo atractivo hacia la meta y dos campos repulsivos, uno relacionado con los obstáculos y otro por los propios agentes del enjambre, para evitar colisiones. Además, se implementa un método de obstáculos virtuales para abordar el problema de mínimos locales. Con todo esto, los resultados obtenidos muestran que los APFs constituyen una solución para la planificación de rutas en enjambres y, aunque la aplicación final de dicho trabajo se oriente a vehículos terrestres, la planificación y desarrollo del método sobre entornos en 3D hacen que sea totalmente aplicable a enjambres de UAVs.

Finalmente, han surgido trabajos que desarrollan algoritmos de APFs mejorados (Improved Artificial Potential Field (IAPF)) para obtener soluciones óptimas, resolviendo los problemas de mínimos locales o suavizando las trayectorias. El IAPF propuesto introduce una restricción de ganancia al modelo de campo repulsivo potencial. Además, se agrega un factor aleatorio para evitar la caída en estos mínimos locales y, finalmente, un método basado en las curvas B-Spline, para suavizar las rutas obtenidas. La restricción de ganancia permite controlar las

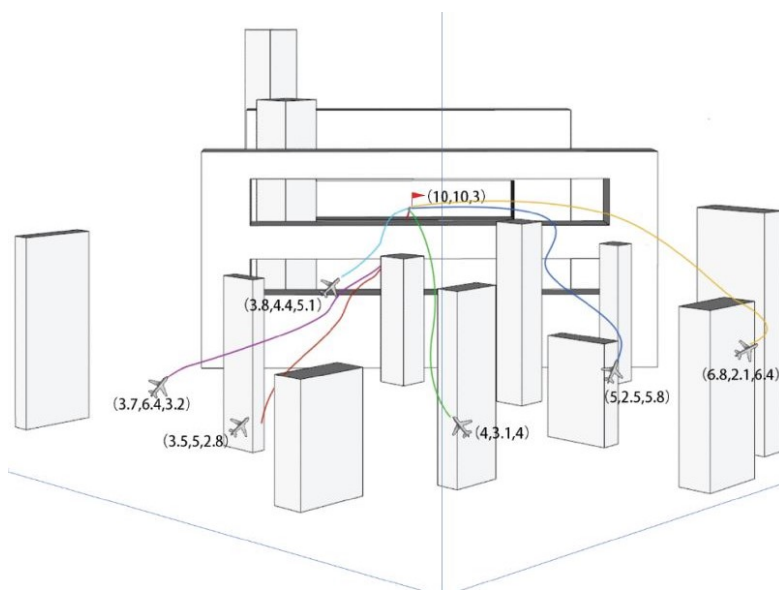


Figura 2.18: Resultados obtenidos empleando APFs para la planificación de rutas en entornos urbanos (20) ©2017 IEEE.

fuerzas repulsivas de los obstáculos en función de la distancia de los vehículos a la meta, de tal manera que si el obstáculo está cerca de la meta se reduce el campo repulsivo de dicho obstáculo. Los resultados obtenidos muestran, como se ve en la Figura 2.19, que los métodos implementados permiten generar una solución óptima, segura y suavizada al problema de la planificación, evitando problemas como mínimos locales.

2.3.3. Métodos basados en modelo matemático

Esta sección incluye los algoritmos basados en modelos matemáticos como son el programa lineal de enteros mixtos (Mixed Integer Linear Program (MILP)), el programa cuadrático de enteros mixtos (Mixed Integer Quadratic Program (MIQP)) y el control óptimo (Optimal Control (OC)).

Estos métodos se caracterizan por establecer restricciones cinemáticas y dinámicas para modelar el entorno y el sistema. Además, se emplea una función de coste para incluir los límites de las restricciones como ecuaciones o desigualdades, cuya minimización permite obtener una solución óptima.

Los métodos basados en modelos matemáticos tienden a tener una formulación compleja y, por lo tanto, un alto coste computacional. Para afrontar este problema se establecen procesos de decisión discretos o, en ocasiones, los algoritmos se orientan a describir partes concretas del problema combinando métodos para

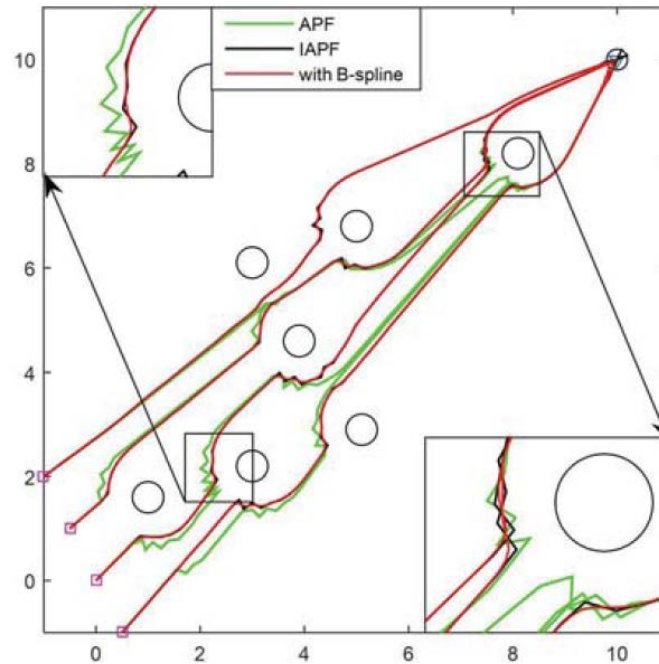


Figura 2.19: Solución a la planificación de rutas empleando algoritmos APFs mejorados (IAPF) (21) ©2019 IEEE.

cubrir todo el problema de planificación de rutas.

2.3.3.1. Programa lineal de enteros mixtos

Los algoritmos MILPs se basan en una función de coste que considera aspectos como las restricciones cinético-dinámicas, la distancia mínima, la energía o las amenazas presentes en el entorno. Además, se caracteriza por ser un algoritmo lineal que modela tanto la cinemática como la dinámica del entorno para representar el espacio de trabajo y los sistemas. En resumen, los algoritmos MILP presentan una alta capacidad para modelar aspectos esenciales del problema y describir casi toda la información (176).

En (179), se propone una formulación MILP para poder realizar un servicio de escolta con UAVs, que trabajen de manera simultánea para diferentes clientes. El modelo MILP permite representar formalmente los problemas de esta aplicación e introducir la posibilidad de utilizar un planificador basado en las ubicaciones iniciales y el estado de las baterías de los UAVs. Se presenta una planificación de ruta basada en MILP, en la que la función objetivo minimiza la suma de la distancia total ponderada recorrida por el UAV junto con el número de trabajos atendidos. Además, se introducen restricciones dentro del modelo MILP para permitir la

coordinación de trayectos de los diferentes UAVs. Estas restricciones consideran la ubicación inicial del UAV, la garantía de que los UAVs lleguen a su destino, los niveles de batería o que las tareas son al menos realizadas por un UAV. Con la formulación MILP y el desarrollo de una heurística eficiente, es posible desarrollar una herramienta para la planificación de rutas, de tal manera que un conjunto de UAVs puede proveer un servicio de escolta a un grupo de consumidores.

Otro trabajo que presenta un desarrollo de un algoritmo MILP es (180). El objetivo de dicha investigación es generar una planificación de rutas óptima para un enjambre destinado a la fumigación en campos agrícolas. Para ello, se genera un modelado exhaustivo del espacio de trabajo y de los componentes del enjambre empleado y, mediante formulación MILP, se establecen restricciones para obtener un recorrido completo para cada agente, de manera que sea capaz de visitar todos los nodos previamente configurados.

2.3.3.2. Programa cuadrático entero mixto

Los algoritmos MIQPs están relacionados estrechamente con los algoritmos MILPs, con la principal diferencia de que la resolución debe realizarse sobre una función cuadrática objetivo (181).

Existen líneas de investigación que han empleado métodos basados en MIQP para generar trayectorias 3D en entornos con presencia de obstáculos (182). La idea es crear rutas óptimas en sistemas con múltiples quad-cópteros de diferentes tamaños, características y capacidades. Es importante enfatizar que, en la línea de investigación presentada en este artículo, el uso de funciones polinómicas, para sintetizar las trayectorias en el espacio de salida, aparece como un aspecto clave. Esto permite reforzar la continuidad entre las ubicaciones intermedias hasta llegar a cualquier derivación de la posición deseada. Los resultados muestran una solución viable para pequeños sistemas que se desplazan por entornos sencillos, es decir, con un número reducido de obstáculos. Además, es necesario conocer tanto la posición inicial como la posición final de los diferentes agentes del equipo. Este método puede imponer restricciones en las posiciones, velocidades, aceleraciones, colisiones, permitiendo diferentes tamaños, capacidades y efectos dinámicos que varían entre los diferentes UAVs empleados, llegando a soluciones factibles en un orden de magnitud de *milisegundos*.

Este mismo grupo de investigación presentó otro trabajo que recoge el uso del método MIQP para generar trayectorias en enjambres de micro UAVs (183). Con el fin de evitar un aumento del coste computacional que sufren los algoritmos MIQP al aumentar las variables y las restricciones, el conjunto de UAVs se divide en pequeños grupos con formaciones rígidas.

Este tipo de algoritmos también ha sido empleado en el campo de los co-

ches autónomos (22). En el trabajo de investigación, se introduce una formulación cooperativa MIQP para la planificación de la trayectoria de múltiples vehículos. En particular, se plantea el problema de establecer un conjunto de trayectorias cooperativas en vehículos autónomos que se comuniquen entre sí y estén orientadas a escenarios viales no peligrosos. Mediante MIQP se establece una solución óptima global que, junto con los resultados obtenidos, muestra la viabilidad de este método para resolver problemas de planificación cooperativa de rutas en escenarios no peligrosos, tal y como se muestra en la Figura 2.20, y por tanto, aplicable a enjambres de UAVs.

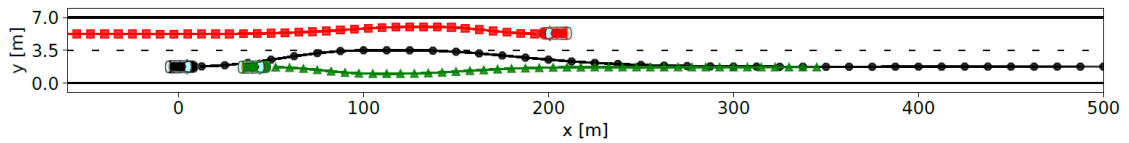


Figura 2.20: Planificación de rutas cooperativa basada en MIQP (22) ©2018 IEEE.

2.3.3.3. Control óptimo

El tercer algoritmo incluido en los modelos matemáticos es la planificación mediante OC. El OC busca encontrar el estado y la ruta basándose en el control de un conjunto de ecuaciones diferenciales. Se considera una extensión de los métodos lineales, pero funciona con un número infinito de variables. Para resolver los problemas mediante un OC, se utiliza el *Hamiltoniano*, el cual permite resolver el problema de optimización basándose en el principio de máximo y, continuar con un procedimiento de solución óptima para la obtención de una solución óptima global (176).

El OC no se centra en un solo tipo de algoritmo, sino que bajo un planteamiento inicial similar se establecen diferentes procedimientos que permiten alcanzar el objetivo final de que un enjambre navegue por el entorno de manera autónoma.

Una de las técnicas más extendida dentro de los métodos basados en optimización es el Modelo de Control Predictivo (Model Predictive Control (MPC)). La idea principal del MPC es encontrar las acciones de control óptimas, que deben llevarse a cabo en el futuro, a partir de la predicción del comportamiento del sistema. En (184) se formula un algoritmo MPC para permitir la planificación de trayectorias de un líder y controlar los movimientos de los seguidores. Los resultados obtenidos permiten observar como el método MPC es capaz de generar caminos libres de colisiones en entornos con obstáculos estáticos y dinámicos manteniendo una formación preestablecida.

Un MPC descentralizado (Decentralized Model Predictive Control (DMPC)) se presenta en (185) para generar rutas en tiempo real para múltiples UAVs. Me-

diante la paralelización del método, los autores obtienen una alta escalabilidad en su trabajo. Los resultados obtenidos muestran la posibilidad de enviar rutas a un enjambre de 20 UAVs a una frecuencia de $20Hz$. Además, con el algoritmo DMPC, es posible reducir hasta un 50 % el tiempo de vuelo requerido para completar todo el trayecto, si comparamos este enfoque con algoritmos basados en diagramas de Voronoi.

Otra técnica incluida en esta sección es la optimización covariante hamiltoniana para la planificación del movimiento (Covariante Hamiltonina Optimization for Motion Planning (CHOMP)). CHOMP es un método que permite la optimización y mejora de trayectorias optimizando un objetivo funcional. Esta optimización se realiza buscando suavizar la ruta y evitando obstáculos. Su desarrollo se basa en el uso de nociones superiores de geometría y permite posicionar el algoritmo como una poderosa herramienta en el campo de la planificación de trayectorias (186).

Trabajos como (187), muestran como CHOMP podría ser empleado para problemas de planificación en enjambres de vehículos aéreos autónomos. En concreto, en este artículo, los autores utilizan CHOMP para una fase de refinamiento de la trayectoria de tal manera que se permite la navegación autónoma y segura a través de dichas trayectorias. CHOMP se utiliza como planificador local para evitar obstáculos dinámicos e incluye una extensión del método para poder resolver posibles conflictos de ruta entre varios agentes. Aunque este artículo se orienta a vehículos auto-guiados, su solución es extensible a enjambres de UAVs, pues el algoritmo de CHOMP se emplea en una fase posterior para el refinamiento de trayectorias previamente calculadas y, se introduce una extensión del método, para poder considerar conflictos en las trayectorias de distintos agentes, lo cual es una herramienta útil en el caso de enjambres de UAVs que trabajan de manera cooperativa.

2.3.4. Métodos bio-inspirados

Estos métodos buscan imitar el comportamiento biológico para tratar de obtener una solución al problema de la planificación de rutas. A diferencia de los modelos matemáticos, los métodos bio-inspirados no se centran en modelar el entorno y presentan un desarrollo que evita problemas como caer en mínimos locales o resolver funciones objetivas complejas.

Los algoritmos bio-inspirados intentan imitar cómo se comportan y actúan los organismos vivos para intentar aplicarlo a la generación de rutas óptimas. Entre las principales características de estos métodos se encuentran: no ser deterministas, presentar estructuras paralelas y ser adaptativos. Este conjunto de factores permite que estos métodos generen soluciones óptimas a los problemas de búsqueda de trayectorias sin tener que conocer, de forma exhaustiva, el entorno en el que se está desarrollando la misión. Este hecho los convierte en métodos efectivos para

resolver problemas de múltiples objetivos (188).

Dentro de estos métodos se establecen dos grupos diferenciados: los algoritmos evolutivos, que analizan el comportamiento de una especie, y las redes neuronales artificiales (Artificial Neural Network (ANN)), basadas en imitar las conexiones y funcionamiento de las neuronas humanas al procesar información.

2.3.4.1. Algoritmos evolutivos

Los algoritmos evolutivos fueron implementados desde un principio para resolver problemas tradicionales de programación lineal y dinámica caracterizados por poseer una gran cantidad de variables. Su desarrollo se basa en la búsqueda estocástica que imita la evolución y el comportamiento social de los sistemas biológicos (189).

El procedimiento seguido por los algoritmos evolutivos consiste, en primer lugar, en seleccionar aleatoriamente las posibles soluciones con una primera generación. En el siguiente paso, se consideran aspectos como la capacidad de los UAVs, el objetivo a alcanzar o las limitaciones existentes en todo el entorno-sistemas. A continuación, se selecciona un conjunto de padres de la primera generación para la siguiente. Finalmente, se realiza un proceso de mutación y cruce que se repite hasta lograr el objetivo. Los mejores individuos son decodificados y empleados como nodos para establecer la ruta óptima.

Los algoritmos evolutivos engloban un conjunto de técnicas que siguen este procedimiento y que son: el *Algoritmo Genético (Genetic Algorithm (GA))*, el *Algoritmo Memético*, la *Optimización del Enjambre de Partículas (Particle Swarm Optimization (PSO))*, la *Optimización de Colonias de Hormigas (Ant Colony Optimization (ACO))* y el *Shuffled Frog Leaping Algorithm*. El GA fue la primera propuesta de algoritmo evolutivo y, posteriormente, el resto de técnicas aparecieron inspiradas en diferentes procesos de la naturaleza.

Al igual que con el resto de métodos analizados, el algoritmo genético también ha sido modificado y mejorado tratando de encontrar óptimas soluciones. Así pues, con el objetivo de planificar rutas que minimicen el tiempo total de viaje de un conjunto de vehículos y, tratar además, de asignar tareas a cada uno de ellos, haciendo que cada vehículo alcance un conjunto de destinos, se ha implementado un algoritmo genético coevolutivo de múltiples poblaciones (Coevolutionary Multi-population Genetic Algorithm (CMGA)) (148). Esta solución se aplica a un conjunto de vehículos autónomos aéreos que navegan en un campo de deriva. Los resultados obtenidos muestran que el CMGA tiene un excelente desempeño en la resolución de problemas de planificación de rutas con múltiples objetivos al trabajar con enjambres de UAVs heterogéneos.

En línea con la mejora de los algoritmos genéticos, surgen trabajos que de-

sarrollan GA mejorados para conseguir una solución a la planificación de trayectorias (190). Las mejoras implementadas en el artículo referenciado consisten en: utilizar tres operadores heurísticos cruzados, produciendo un flujo óptimo caracterizado por más información y, aplicar restricciones de doble vía para minimizar la distancia total recorrida por todos los agentes y, la distancia recorrida por cada uno de ellos. Los resultados garantizan que estas mejoras permiten reducir las distancias recorridas por cada vehículo y, en su defecto, por todo el conjunto, frente al uso de algoritmos genéticos clásicos. La solución presenta una alta aplicabilidad a enjambres de UAVs en los que se busque minimizar la distancia total recorrida por el enjambre y, con ello el tiempo de respuesta en un conjunto de tareas o misión.

El uso de algoritmos genéticos mejorados (Enhanced Genetic Algorithms (EGAs)) también aparecen en otros trabajos (191). En este caso, los autores presentan un método que combina los APF con los EGAs para poder planificar rutas de enjambres en entornos continuos. El algoritmo de APF genera todos los caminos posibles entre los puntos iniciales y finales, mientras que el algoritmo de EGA encuentra las rutas óptimas entre las distintas ubicaciones. En esta investigación, la EGA emplea cinco operadores cruzados y de mutación para mejorar los caminos iniciales. Además, el EGA incluye, dentro de la función objetivo, un parámetro para evitar posibles colisiones entre las rutas. Los resultados muestran la eficiencia de esta implementación en comparación con otros métodos clásicos o con otros algoritmos evolutivos. Así, no solo se es capaz de establecer rutas libres de colisiones, sino que la solución encontrada es la óptima.

Finalmente, aunque los GAs son los más empleados en el campo de la planificación de trayectorias, existen otras técnicas como la PSO, que también se emplean en trabajos recientes para solucionar este problema (23). En el artículo, se presenta esta técnica mejorada para poder establecer rutas de UAVs en entornos conocidos, estáticos y ricos en obstáculos. La mejora introducida consiste en una función de vibración que mejora las soluciones colisionadas en lugar de abandonarlas. Los resultados obtenidos muestran como este método es capaz de generar una solución y establecer múltiples rutas para un conjunto de UAVs en un terreno accidentado, como el que se muestra en la Figura 2.21.

2.3.4.2. Redes Neuronales Artificiales

Desde las últimas décadas del siglo pasado, las redes neuronales artificiales (ANNs) se han empleado para dar solución a un gran número de problemas y aplicaciones, siendo la navegación autónoma uno de ellos, para vehículos autónomos aéreos (192).

La idea de las ANNs es generar un panorama dinámico, acompañado de la máxima actividad neuronal entre las distintas capas de neuronas. Aplicado a la

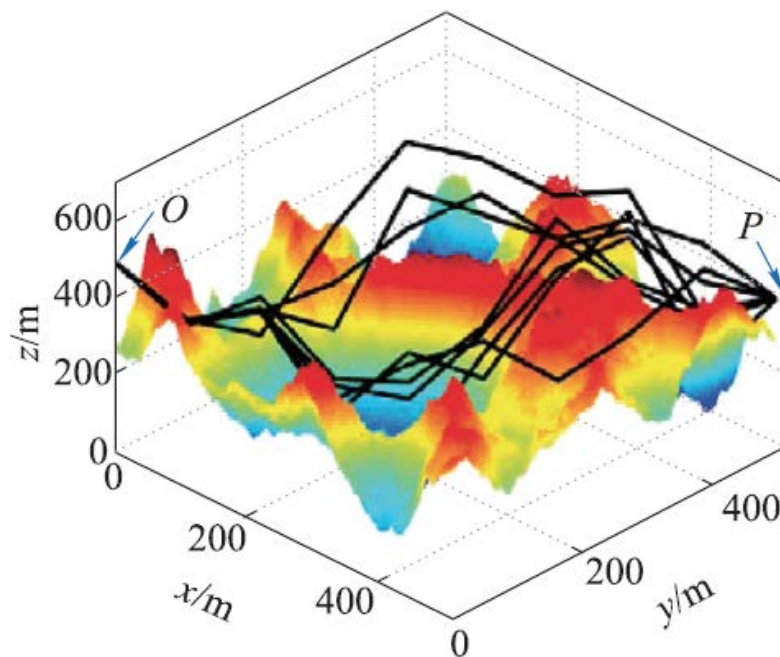


Figura 2.21: Planificación de rutas basada en PSO (23) ©2020 IEEE.

navegación autónoma, se introduce la dinámica de los vehículos en la red neuronal y se garantiza que la actividad neuronal se propague por el espacio libre, pudiendo navegar por rutas seguras sin supervisión.

Si bien en los últimos años se ha producido una expansión de estas técnicas, su principal desventaja es que, al ser un método bioinspirado, no se pueden estandarizar, es decir, no se pueden formar ni reglas ni modelos canónicos. Por lo tanto, el encontrar una solución óptima a un problema no garantiza que sea aplicable a otro problema similar aplicado en un entorno diferente (176). Las ANNs se encuentran englobadas dentro de los denominados métodos de AI y, como los anteriores, presenta diferentes técnicas, entre las que destaca el aprendizaje profundo (Deep Learning (DL)), el aprendizaje por refuerzo (RL) o, una combinación de ambas.

El RL está clasificado como un sub-campo perteneciente al aprendizaje automático (Machine Learning (ML)), pero es cierto que el RL incorpora, de forma nativa, una dimensión extra en las ecuaciones de aprendizaje, la cual habitualmente es el tiempo, provocando que el RL se sitúe como una técnica al margen del ML y, se aproxime a las técnicas de DL. Por un lado, RL utiliza muchos métodos establecidos en el aprendizaje supervisado, como son las redes neuronales profundas (Deep Neural Networks (DNNs)) para la aproximación de funciones, el descenso de gradientes estocásticos y la retropropagación para aprender la representación de los datos, pero es cierto que su aplicación suele llevarse a cabo de manera diferen-

te (193). Además, los recientes desarrollos en RL, combinando estas técnicas con el DL ha permitido resolver problemas mucho más desafiantes que antes, gracias en parte a los desarrollos de métodos y herramientas del DL.

El RL se caracteriza por poder tomar datos de un entorno dinámico con el principal objetivo de generar una secuencia de acciones que permitan alcanzar un resultado óptimo. Para ello, se establece un uso de recompensas que son generadas por el entorno en función de las acciones tomadas por el agente y que, a través de un sistema de realimentación, el agente puede valorar la acción tomada en función de dicha recompensa y actuar en consecuencia. De esta manera, el proceso se repite de manera cíclica hasta lograr el aprendizaje del agente.

Los trabajos recientes en esta área escenifican cómo han surgido líneas de investigación en este área que tratan de proponer soluciones al problema de la planificación de rutas y la navegación sin supervisión de vehículos aéreos autónomos.

La combinación de Redes Neuronales Convolucionales (Convolutional Neural Networks (CNNs)) junto con un algoritmo de RL llamado Deep Q-Network (DQN) se ha presentado como una solución al problema de la planificación de rutas para enjambres (24). La idea de combinar estos métodos es generar un análisis eficiente de la situación. Por un lado, las CNNs analizan la situación utilizando la información capturada por los agentes, mientras que el algoritmo DQN es el encargado de, a partir de la información sacada de las CNNs, generar las acciones para que los agentes del enjambre naveguen con un esquema como el que se muestra en la Figura 2.22. Los resultados obtenidos muestran cómo esta técnica elimina los problemas convencionales al hacer que las acciones elegidas procedan de la información recolectada por todos los UAVs, los cuales tienen una influencia mutua, a través de las CNNs. Además, los resultados muestran cómo este tipo de técnica es aplicable tanto en entornos estáticos como dinámicos, gracias a que los agentes comparten la memoria utilizada para el aprendizaje.

Buscando resolver el problema tanto de la planificación de rutas, como de la asignación de tareas, han surgido investigaciones basadas en ANN para buscar una solución óptima a dichos problemas (194). La técnica utilizada en este artículo se basa en un Gradiente de Política Determinista Profunda Multi-Agente (Multi-Agent Deep Deterministic Policy Gradient (MADDPG)), que pertenece al campo del RL de múltiples agentes. El procedimiento seguido consiste en emplear el algoritmo MADDPG para entrenar el sistema, resolviendo de manera simultánea tanto la asignación de objetivos como la planificación de rutas, siguiendo la estructura de recompensa correspondiente. Para garantizar la aplicación de este desarrollo en entornos dinámicos y en tiempo real, se utiliza una ANN simple en el sistema. Los resultados muestran que el algoritmo implementado es efectivo en varios escenarios y aplicable a un conjunto escalable de UAVs.

Existen modificaciones en los algoritmos de RL para poder ofrecer soluciones

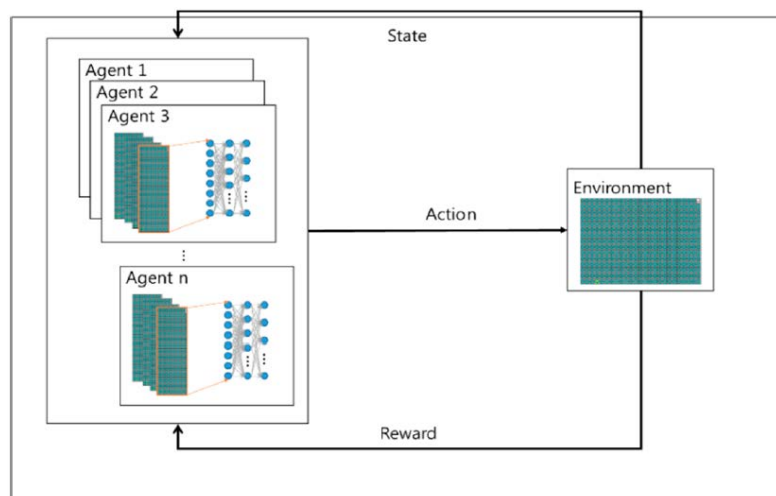


Figura 2.22: Esquema de método combinado de redes neuronales convolucionales y Deep Q-Learning (24).

al problema de la planificación sin tener una velocidad de aprendizaje lenta o pudiendo aprender en entornos completamente desconocidos (195). Para ello, se propone un algoritmo de RL multi-agente denominado Win or Learn Fast Policy Hill-Climbing (WoLF-PHC) cuya principal novedad es asegurar que cualquier agente tenga sus respuestas cuando el resto permanezca en un estado estacionario. De esta forma, se asemeja la actuación del agente a una estrategia determinista en la que no son necesarios conocimientos previos, como la función de transición o de recompensa, permitiendo la generación de rutas en entornos dinámicos.

Las técnicas de RL también han sido aplicadas a problemas de navegación sin supervisión en una formación determinada (196). Para ello, se ha implementado una arquitectura multi-capa para poder controlar la formación de varios vehículos mediante la planificación de movimientos. Esta capa múltiple se divide en dos subconjuntos con diferentes tareas. En primer lugar, un algoritmo de Q-Learning para determinar la configuración de los vehículos y, en segundo lugar, un algoritmo de Gradiente de Políticas de Refuerzo Profundo para llevar a cabo la planificación de rutas libres de colisiones. La idea de utilizar esta combinación es que el algoritmo de Q-Learning tiene un bajo coste y tiempo computacional, por lo que puede ser empleado para este tipo de tareas, mientras que el algoritmo de Gradiente de Políticas de Refuerzo Profundo, en paralelo, se encarga de tareas más complejas, como entrenar el sistema en nuevos entornos y descubrirlos por ensayo y error. Los resultados muestran cómo los agentes logran establecer rutas óptimas y desplegarse en el entorno con una formación determinada.

2.3.5. Conclusiones

En esta sección se han detallado aquellos métodos y algoritmos cuya implementación se ha dirigido en los últimos años a aportar soluciones a la planificación de rutas y trayectorias para enjambres de UAVs que deben trabajar de manera cooperativa. Los trabajos recogidos a lo largo de la misma muestran un amplio abanico de métodos posibles a adoptar para alcanzar una solución óptima a la generación de trayectorias seguras para un enjambre de UAVs que navegue sin supervisión por distintos entornos. Este conjunto de métodos son resumidos, junto con sus principales ventajas e inconvenientes, en las Tablas 2.1 y 2.2.

Tras el análisis de los diferentes métodos, se ha optado por el desarrollo e implementación de un método de planificación de trayectorias que permita a un enjambre escalable y heterogéneo de UAVs navegar, de manera coordinada y sin supervisión, por distintos entornos de cara a poder acometer labores claves en la lucha contra incendios, tanto forestales como urbanos, lo cual constituye el objetivo principal del presente trabajo. De cara a poder actuar con un número indeterminado de UAVs en entornos de grandes dimensiones, como puede ser un edificio de viviendas o una amplia extensión de terreno forestal, se debe orientar los métodos de planificación hacia el uso de algoritmos de exploración rápidos, que posean un coste computacional bajo y que permita obtener una solución al problema de generación de trayectorias escalable.

La necesidad de acometer una exploración rápida en grandes entornos y en problemas de dimensiones altas, como es el caso que concierne a este trabajo debido al uso de un número indeterminado de UAVs trabajando de manera coordinada, hace que la investigación se oriente hacia el uso de métodos basados en muestreo, los cuales se caracterizan por tener una baja complejidad computacional incluso para exploraciones de extensas áreas de terreno. Dentro de los métodos basados en muestreo, el hecho de tener que trabajar con UAVs hace que la planificación de rutas deba llevarse a cabo sobre un entorno en 3D, lo cual provoca que algoritmos como los basados en diagramas de Voronoi, cuya aplicación se ciñe a mapas en 2D sean descartados. En cuanto al resto de los métodos, estudios comparativos como el recogido en (197) muestra que los algoritmos basados en PRM son adecuados para los problemas de planificación de trayectorias desde el punto de vista del tiempo y la distancia recorrida. Además, el algoritmo de PRM cuenta con la ventaja de que la exploración se realiza una única vez y, el resultado de dicha exploración se aprovecha por el conjunto de los agentes que forman el enjambre. En caso de los algoritmos de RRTs la exploración se realiza de manera rápida, incluso en grandes entornos, pero debe llevarse a cabo una vez por cada uno de los UAVs que intervengan en la operación, lo cual limita la escalabilidad del método.

El principal inconveniente de este método es su pasividad y, por tanto, la necesidad de combinarlo con un método basado en grafo para generar rutas óptimas.

Por este motivo, y ante el análisis y resultados recogidos en el apartado de métodos basados en esta estructura matemática, se ha optado por emplear el algoritmo A* para completar el método obteniendo una herramienta de planificación de trayectorias aplicable a enjambres de UAVs de un número variable de agentes. De nuevo, estudios como (197) recogen que el algoritmo A* presenta buenos resultados en cuanto a la minimización de la distancia recorrida, aspecto clave en labores como la extinción de incendios, dónde el tiempo de respuesta es un factor clave del éxito. Con esta herramienta se podrán establecer un conjunto de trayectorias seguras en entornos de grandes dimensiones y diferentes características permitiendo desplegar una flota de UAVs, dotados con sistemas de detección y evasión de obstáculos y de toma de decisión para una navegación autónoma, en un corto espacio de tiempo.

En definitiva, tras el análisis de los métodos presentes en la literatura para la planificación de rutas y trayectorias orientadas al uso de UAVs y de los MRS, se ha optado por un método, cuya estructura se refleja en el diagrama de la Figura 2.23, formado por un algoritmo de PRM encargado de realizar la exploración del entorno, generando un grafo en 2D y 3D, el cual es empleado junto a una implementación basada en A* para la obtención de un conjunto de trayectorias que minimicen la distancia total recorrida por el enjambre.

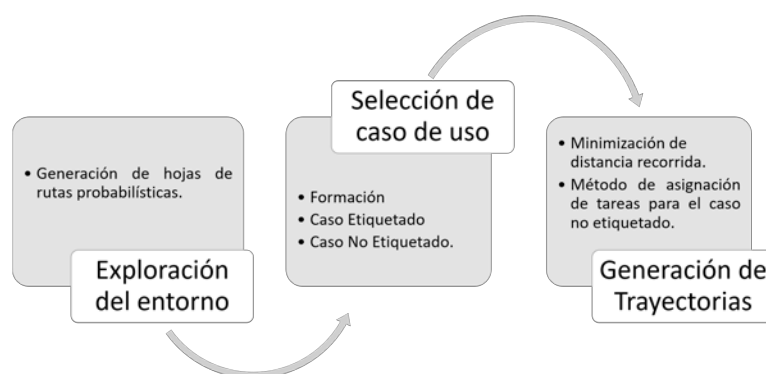


Figura 2.23: Diagrama método de planificación para enjambre de UAVs.

De esta forma, el aumento de los agentes participantes en la misión provocará la amortización del gasto computacional derivado de la exploración del entorno, la cual se realiza una única vez, generando un conjunto de rutas posibles. Dicho método se configura de tal forma que permite su utilización en diferentes casos de uso, los cuales se orientan hacia su aplicación al problema de los incendios, tanto forestales como urbanos, poniendo los desarrollos acometidos en línea con los objetivos de la tesis doctoral de disponer de un enjambre de UAVs capaz de acometer, de manera coordinada, tareas en el área de trabajo de un incendio.

Tabla 2.1: Métodos empleados para planificación de rutas con aplicabilidad a enjambres de UAVs (Parte 1).

Método	Algoritmo	Ventajas	Desventajas	Referencias
Descomposición basada en grafo	Dijkstra	<ul style="list-style-type: none"> ● Fácil implementación ● Solución Óptima ● Múltiples entornos 	<ul style="list-style-type: none"> - Alto coste computacional - Entornos estáticos 	(147; 148; 142) (149; 150)
	A*	<ul style="list-style-type: none"> ● Solución óptima ● Búsqueda rápida y guiada ● Implementación Online 	<ul style="list-style-type: none"> - Alto coste computacional - Grandes entornos - Entornos estáticos 	(151; 152) (153; 154)
	D*	<ul style="list-style-type: none"> ● Solución óptima ● Búsqueda rápida y guiada ● Entornos dinámicos 	<ul style="list-style-type: none"> - La función heurística utiliza distancias poco realistas 	(156)
Basado en muestreo	PRM	<ul style="list-style-type: none"> ● Solución reutilizable ● Rápida exploración de grandes entornos 	<ul style="list-style-type: none"> - Patrón en Zig-Zag - Método pasivos 	(159; 160; 161; 164) (162; 2; 5; 163)
	Voronoi	<ul style="list-style-type: none"> ● Implementación fácil ● Coste computacional bajo 	<ul style="list-style-type: none"> - Método pasivo - No convergencia - Entornos estáticos 	(15; 166; 168; 169) (167; 172; 171; 170)
	RRT	<ul style="list-style-type: none"> ● Bajo coste computacional ● Exploración rápida 	<ul style="list-style-type: none"> - Generación de rutas simple 	(173; 17; 174) (16; 175)
Basado en	APF	<ul style="list-style-type: none"> ● Baja complejidad de tiempo ● Implementación Online 	<ul style="list-style-type: none"> - Mínimo local 	(18; 177; 20) (178; 21)

Tabla 2.2: Métodos empleados para planificación de rutas con aplicabilidad a enjambres de UAVs (Parte 2).

Método	Algoritmo	Ventajas	Desventajas	Referencias
Basado en modelo matemático	MILP	<ul style="list-style-type: none"> ● Sistema completo y información del entorno 	<ul style="list-style-type: none"> – Alta complejidad temporal 	(179; 180)
	MIQP	<ul style="list-style-type: none"> ● Consideración del tiempo para la planificación de rutas 	<ul style="list-style-type: none"> – Resolución de modelos matemáticos complejos 	(182; 183) (22)
	OC	<ul style="list-style-type: none"> ● Solución considera el control de los sistemas 	<ul style="list-style-type: none"> – Soluciones no analíticas 	(184; 185) (186; 187)
Bio-inspirados	ANN	<ul style="list-style-type: none"> ● Entornos y sistemas infinitos 	<ul style="list-style-type: none"> – Soluciones no genéricas – El éxito depende de que las reglas y organismos estén correctamente parametrizados 	(24; 194) (74; 194) (195; 196)
	EA	<ul style="list-style-type: none"> ● Adecuado para problemas <i>NP-hard</i> y multi-objetivos 	<ul style="list-style-type: none"> – Alta complejidad temporal 	(148; 190) (191; 23)

2.4. Resumen

En este capítulo, se ha establecido una revisión de los avances y estado actual de los vehículos aéreos no tripulados, recogiendo las principales características de estas aeronaves, sus aplicaciones y, también, las limitaciones encontradas hasta la actualidad. Se ha estudiado qué aspectos han provocado el auge de este sector, las diferentes plataformas y soluciones tecnológicas englobadas dentro de una categoría tan amplia como son los drones aéreos y, los desafíos actuales a los que se enfrenta esta tecnología de cara a mejorar su aplicabilidad. Los nuevos avances y desarrollos deberán hacer frente a estos desafíos, aumentando la competitividad del sector y, manteniendo las expectativas de crecimiento del mismo en los próximos años. En línea con estos avances se recoge la actividad investigadora de la empresa Drone Hopper S.L, destinada a la fabricación y desarrollo de drones industriales con mejoras que permitan aumentar la autonomía de vuelo y la capacidad de carga.

Las secciones del presente capítulo se centran en el estudio de las metodologías actuales aplicadas a los enjambres robóticos, analizando aquellos conceptos que permiten situar esta tecnología dentro del campo tecnológico de los MRS. Dicho estudio se centra en analizar la aparición, avance y aplicabilidad de estos sistemas cooperativos. La posibilidad de contar con un sistema formado por un conjunto de UAVs capaces de actuar de manera coordinada y sin supervisión han demostrado que los MRS en general y, los enjambres de UAVs en particular constituyen, a día de hoy, una herramienta robusta, eficaz, autónoma y versátil para un amplio rango de aplicaciones y servicios, tanto del ámbito civil como del militar.

Por último, se ha examinado en su totalidad, los métodos relacionados con la planificación de trayectorias cuya solución es aplicable a enjambres formados por sistemas aéreos no tripulados que trabajen de manera cooperativa en una misma área, llevando a cabo una revisión completa de los mismos y, analizando las principales ventajas y desventajas de cada uno de ellos.

Capítulo 3

Planificación de Trayectorias para Enjambres de UAVs

Este capítulo presenta una solución robusta y escalable a la planificación de trayectorias para enjambres de UAVs, que permite acometer trabajos bajo diferentes configuraciones, de cara a la realización de labores de interés en la lucha contra el fuego, dando soporte y ayuda a los equipos de emergencias. ¹.

La planificación de trayectorias es considerada un aspecto crucial para lograr el trabajo autónomo de sistemas móviles no tripulados como es el caso de los UAVs. Como se analiza y estudia en el capítulo 2, a lo largo de los años se han

¹Publicaciones del autor relacionadas con el capítulo:

- Sosa San Frutos R., Al Kaff A., Hussein A., Madridano Á., Martín D., de la Escalera A. (2020) ROS-Based Architecture for Multiple Unmanned Vehicles (UXVs) Formation. In: Moreno-Díaz R., Pichler F., Quesada-Arencibia A. (eds) Computer Aided Systems Theory ? EUROCAST 2019. EUROCAST 2019. Lecture Notes in Computer Science, vol 12013. Springer, Cham. (3).
- Multi-Path Planning Method for UAVs Swarm Purposes, 2019 IEEE International Conference of Vehicular Electronics and Safety (ICVES) (5).
- Á. Madridano, A. Al-Kaff, D. M. Gómez and A. de la Escalera, "Multi-Path Planning Method for UAVs Swarm Purposes,"2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES), Cairo, Egypt, 2019, pp. 1-6, doi: 10.1109/ICVES.2019.8906381. (5).
- Al-Kaff, A.; Madridano, Á.; Campos, S.; García, F.; Martín, D.; de la Escalera, A. Emergency Support Unmanned Aerial Vehicle for Forest Fire Surveillance. *Electronics* 2020, 9, 260. (1).
- Madridano, Á., Al-Kaff, A., Martín, D., Escalera, A. D. L., & Armingol, J. M. (2019). Método de planificación de trayectorias múltiples para enjambre de UAVs. In *XL Jornadas de Automática* (pp. 741-748). Universidade da Coruña, Servizo de Publicacións (4).

desarrollado y explotado diferentes líneas de investigación en este campo. En el presente trabajo, la solución a la planificación de trayectorias está basada en la combinación de métodos de muestreo aleatorio, como es el algoritmo de PRM, con métodos basados en descomposición de grafo, en este caso el uso del algoritmo A*, para poder generar un conjunto de trayectorias seguras y óptimas para un enjambre escalable de UAVs.

A lo largo del capítulo se analiza, estudia y valida el uso de este tipo de métodos para diferentes actuaciones del enjambre como, por ejemplo, el movimiento coordinado de los drones para que cada UAV alcance una ubicación determinada, minimizando la distancia total de la solución global generada para el conjunto de UAVs o, la navegación coordinada de los UAVs para alcanzar una misma ubicación bajo una formación determinada, de tal manera que se produzca un despliegue eficaz de la flota de UAVs para acometer tareas de ayuda y soporte en la lucha contra incendios, tanto forestales como urbanos.

Los métodos de planificación de trayectorias propuestos se complementan con un algoritmo de suavizado de trayectorias que mejore el comportamiento dinámico de los UAVs durante la navegación autónoma, debido al inconveniente derivado del uso de PRM, como es la generación de rutas bajo un patrón de Zig-Zag (198).

3.1. Planificador de Trayectorias Global en 2D

En esta sección se analiza un método de planificación de trayectorias para enjambres de UAVs sobre información del entorno en 2D, aunque incluyendo aspectos que permiten su aplicación a problemas tridimensionales y, por tanto, a UAVs.

El método está basado en la combinación de diferentes algoritmos para tratar de generar una solución óptima al problema de planificación de trayectorias para un enjambre de UAVs, que debe navegar de manera autónoma hacia múltiples objetivos. Esta propuesta trata de aportar un método aplicable en entornos de diversa índole y grandes dimensiones y, para un número variable de UAVs, que permita, en un corto intervalo de tiempo, desplegar una flota de UAVs en un entorno del cual se dispone de cierta información preliminar.

El primero de los algoritmos se basa en un método de muestreo recogido en la literatura como es el PRM, el cual posee la ventaja de poder explorar entornos de grandes dimensiones en un corto periodo de tiempo y, generar un grafo con el conjunto total de posibles trayectorias. Esta primera fase se combina con una segunda fase en la que, a través de un algoritmo descomposición de grafos como el algoritmo A*, se establecen, de entre todas las soluciones posibles, aquellas que optimiza la distancia recorrida para cada uno de los UAV del enjambre.

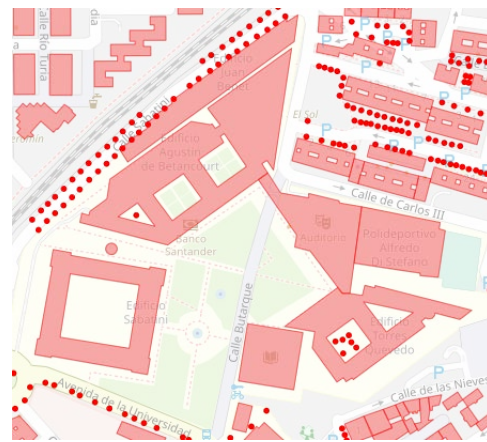
El desarrollo propuesto en 2D muestra que, a través de dicha combinación,

se puede obtener una solución global eficaz al problema de la planificación de trayectorias, para un enjambre de UAVs, formado por un número variable de drones. Junto a esta validación, la implementación de métodos en 2D se establece, en primer lugar, como un paso previo al desarrollo de un planificador global de trayectorias en 3D. En segundo lugar, de cara a la aplicación directa de este planificador en entornos abiertos, caracterizados por no poseer obstáculos solapados en a distintas alturas, como sí sucedería en el caso de un entorno urbano o un edificio de viviendas, dónde el solapamiento en alturas de diferentes obstáculos obliga a realizar la planificación global sobre un modelo tridimensional del entorno.

Esta segunda razón para establecer un planificador 2D para enjambres de UAVs se detalla haciendo uso de las Figuras 3.1. Esta figura muestra, por un lado, un plano en 2D de un entorno abierto correspondiente al campus universitario de la Escuela Politécnica Superior en Leganés, como el de la Figura 3.1a, obtenido a través de herramientas de software abierto como OpenStreetMap (199). Sobre esta imagen, se puede generar mapas, como el recogido en la Figura 3.1b, que permitan establecer obstáculos del entorno, tales como edificios o vegetación y, sobre los que poder establecer trayectorias para una navegación autónoma segura.



(a) Ejemplo de representación de entorno abierto.



(b) Ejemplo de mapa 2D de entorno abierto con diferenciación de obstáculos estáticos.

Figura 3.1: Representación 2D del entorno de la Escuela Politécnica Superior de Leganés.

A partir de esta información, se pueden obtener mapas de ocupación en 2D sobre los que llevar a cabo una planificación de trayectorias que permitan a los UAVs navegar de manera segura sobre este tipo de entornos. En este caso, la información tridimensional no es esencial, puesto que a través de este tipo de información se pueden establecer trayectorias con altitud fija que permitan al enjambre alcanzar

ubicaciones concretas en el entorno evitando los obstáculos presentes en el entorno, como puede ser el caso de actuación en incendios forestales o áreas abiertas como es este caso. Disponer de dicha información permite a los UAVs navegar de manera autónoma y segura a través de un conjunto de trayectorias como se muestra en la Figura 3.2. De esta forma, los agentes del enjambre pueden alcanzar ubicaciones de interés dentro del entorno sin necesidad de disponer de información 3D más completa, la cual si es necesaria a la hora de establecer navegaciones autónomas en áreas dónde se puede producir el solapamiento de obstáculos en altura, como puede ser el caso de acceder a un edificio de viviendas.

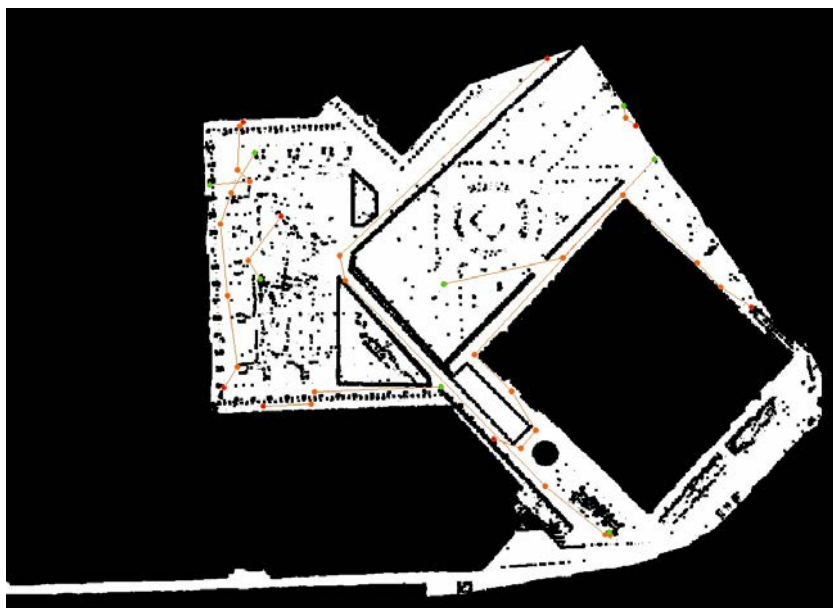


Figura 3.2: Solución obtenida por el planificador de trayectorias en 2D propuesto, actuando en un entorno abierto.

Si bien la solución presentada en esta sección es aplicable a distintas situaciones, las cuales poseen desarrollos diferentes y adicionales a la hora de generar las trayectorias óptimas y, cuyas modificaciones son detalladas posteriormente, el método de planificación en 2D comparte un esquema general para todos los casos formado por un conjunto de fases, como las mostradas en la Figura 3.3.

1. **Representación del entorno:** Para poder llevar a cabo una planificación global de trayectorias se hace necesaria información del entorno, de cara a poder determinar obstáculos estáticos presentes en el entorno. Sobre esta representación 2D del entorno se realiza, en fases posteriores, la generación de un grafo 2D con todas las posibles soluciones, mediante el uso de algoritmos de muestreo. Esta representación del entorno debe consistir en una imagen

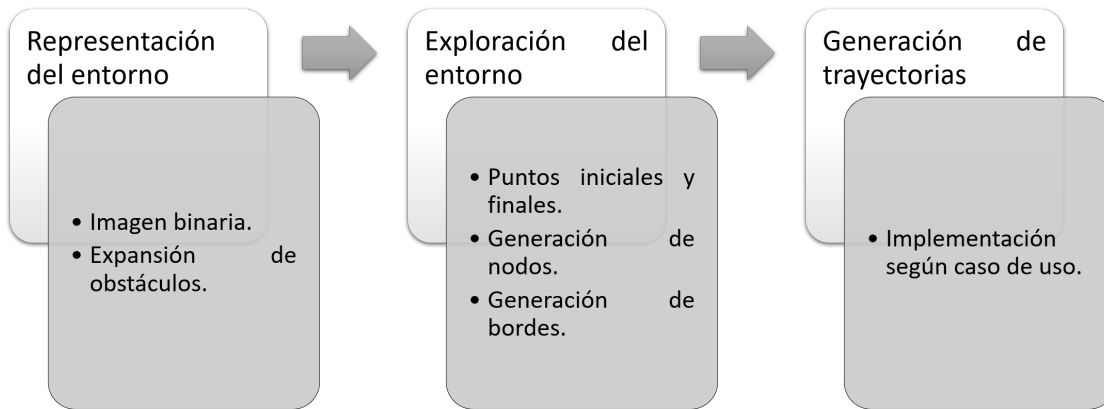


Figura 3.3: Diagrama de flujo para planificación de trayectorias en 2D.

binaria en la que los obstáculos se diferencien del espacio libre mediante los valores de los píxeles de dicha imagen.

Esta imagen digital se caracteriza por poseer dos valores posibles para cada uno de los píxeles (200). De esta manera, la imagen se reduce a valores 0 y 1, es decir, al color negro (valor 0 en escala de grises) y, al color blanco (valor 255 en escala de grises), para poder representar los obstáculos del entorno y, el espacio blanco, respectivamente, tal y como se observa en la Figura 3.4. Un aspecto importante a considerar en esta fase es la relación de *metros* por *pxel* ya que, a la hora de explorar el entorno, los nodos serán creados en una posición concreta de la imagen (x e y en píxeles), pero a la hora de generar la trayectoria será necesario convertir dichas ubicaciones en píxeles en ubicaciones reales en *metros* con respecto a la posición inicial de los UAVs.

Además, esta primera etapa incluye un procesamiento de la imagen que permita expandir los obstáculos, tal y como muestra la Figura 3.5, con el objetivo de aumentar, de manera virtual, el tamaño de los obstáculos, generando un margen de seguridad con respecto a los mismos. Este desarrollo permite, dentro del planificador global propuesto, considerar un tamaño de obstáculos superior al real, lo cual provoca que todos los nodos generados en el espacio libre se encuentren a una distancia de seguridad de los obstáculos, garantizando la navegación libre de colisiones de cada uno de los UAVs. Este es el

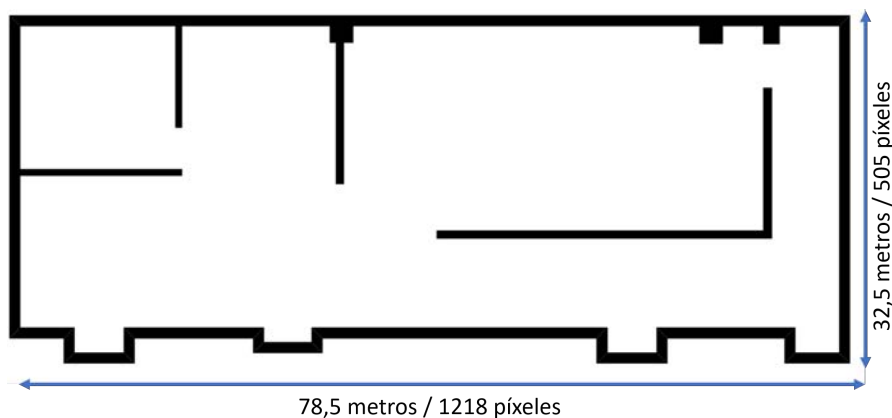


Figura 3.4: Ejemplo de imagen binaria del entorno en 2D.

caso de la ubicación inicial y final número 3, que a pesar de encontrarse en las proximidades de un obstáculo son alcanzables por los UAVs al haberse establecido considerando el margen de seguridad descrito. Este margen es palpable si se compara el tamaño de los obstáculos de la Figura 3.4 con el de la Figura 3.5.

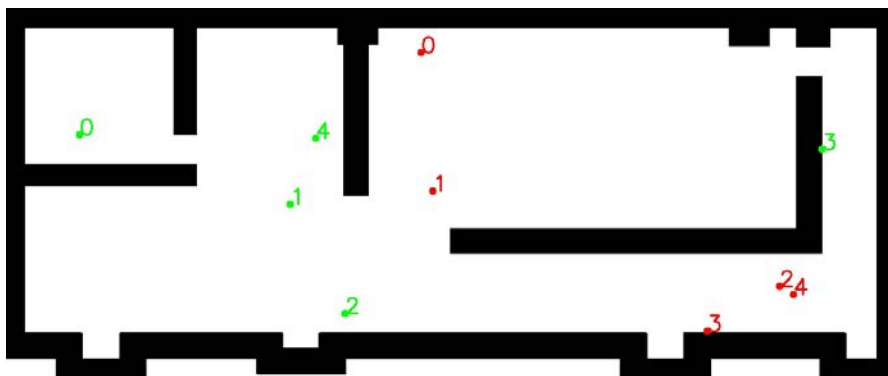


Figura 3.5: Representación del entorno con extensión de obstáculos (5) ©2019 IEEE.

Para realizar esta expansión virtual de los obstáculos, se recorre la imagen en busca de los bordes de los obstáculos, es decir, aquellos píxeles negros que contienen al menos un píxel blanco entre sus 8 píxeles vecinos. Si se produce este caso, se almacena el píxel y, tras recorrer toda la imagen y guardar todos los píxeles correspondientes a bordes de obstáculos, se genera un rectángulo alrededor de cada píxel con un tamaño determinado, dependiente de un parámetro configurable correspondiente al margen de seguridad que se desea

aplicar. Este valor vendrá determinado, principalmente, por el tamaño de los vehículos con los que se trabaje en cada momento.

2. **Exploración del entorno:** Este método de planificación de trayectorias se basa en un algoritmo de muestreo aleatorio, que permita realizar de manera rápida y completa una exploración del área de trabajo. De entre los métodos de muestreo recogidos en la literatura, se ha optado por implementar un método basado en el algoritmo PRM, dado que, en primer lugar, es uno de los métodos indicados para planificadores de ruta globales en entornos de grandes dimensiones y, en segundo lugar, la salida de dicho algoritmo es un grafo, con todas las posibles rutas, reutilizable para cada uno de los diferentes UAVs que forman el enjambre. De esta manera, el coste computacional derivado de la exploración y generación del conjunto conjunto de posibles soluciones se amortiza al emplear el grafo de salida tantas veces como agentes forman el enjambre, para la búsqueda de una solución óptima.

A pesar de ser un método pasivo y, por tanto, necesitar de otros métodos para la generación de una solución óptima, la posibilidad de ejecutar este algoritmo una sola vez generando un grafo a partir del cual obtener todas las trayectorias necesarias para el conjunto de vehículos que forman el enjambre, hace de este método una solución eficaz y, sobre todo, escalable, de cara a poder aumentar el número de agentes que forman el enjambre sin que el tiempo de procesamiento crezca en una proporción mayor o similar. En el caso de otros métodos de muestreo activos, cómo pueden ser los RRTs, sería necesario crear un árbol para cada UAV, estableciendo dos ramas (la que nace en el punto de inicio y la que nace en el punto de destino) para cada uno de ellos, lo que provocaría que el método no poseyera la escalabilidad de la solución aquí propuesta.

Por tanto, se ha enfocado este desarrollo sobre un algoritmo de PRM debido a sus ventajas comparado con otros algoritmos de planificación de trayectorias, como por ejemplo el uso único del algoritmo A*, o los ya comentados algoritmos de RRTs o los GA (197).

El algoritmo de PRM propuesto se estructura en varias fases. En primer lugar, en la exploración del entorno a través de la generación de nodos en ubicaciones aleatorias pertenecientes al espacio libre. Y, en segundo lugar, en la generación de bordes, libres de colisiones, que comuniquen los diferentes nodos entre sí. Analizando en detalle cada una de las fases:

- a) **Generación de nodos:** Consiste en un proceso de exploración aleatoria del entorno, en el que durante una primera fase se añaden los nodos correspondientes a las ubicaciones iniciales y las ubicaciones de destino,

las cuales vienen determinadas por el problema a resolver, y, por tanto, no corresponden a ubicaciones aleatorias.

Esta primera fase no consiste, únicamente, en agregar a la lista de nodos las ubicaciones iniciales y finales, sino que es necesario comprobar, sobre todo en el caso de las posiciones de finales, si estas se encuentran en espacio libre, o si, por el contrario, se ha introducido un objetivo inalcanzable para cualquiera de los UAVs del enjambre.

La segunda fase está orientada a la exploración del entorno, es decir, a la generación aleatoria de un número determinado de nodos sobre el espacio libre del área de trabajo. Para la creación aleatoria de los nodos se emplean las Ecuaciones 3.1 y 3.2, que permiten, a través del uso de librerías estándar de C++, generar un valor aleatorio dentro de unos máximos y mínimos (X_{max} , X_{min} , Y_{max} , Y_{min}) que vendrán definidos por el usuario en función del tamaño de la imagen. La librerías estándar de C++ posee una función denominada *rand()* que se encarga de devolver un valor aleatorio positivo entre 0 y 32767. Este valor máximo viene determinado por la librería a través de la macro constante *RAND_MAX*.

$$X = X_{min} + \frac{(X_{max} - X_{min}) * V_{Aleatorio}}{RAND_MAX + 1,0} \quad (3.1)$$

$$Y = Y_{min} + \frac{(Y_{max} - Y_{min}) * V_{Aleatorio}}{RAND_MAX + 1,0} \quad (3.2)$$

Una vez generadas las coordenadas aleatorias del nodo en píxeles, se comprueba si corresponden a espacio libre y, por tanto, si puede ser almacenado dicho nodo o si, por el contrario, el nodo corresponde a un obstáculo, motivo por el cual debe desecharse y repetirse el proceso de generación de nuevas coordenadas. Este proceso iterativo se repite hasta que se ha alcanzado el número máximo de nodos a generar, el cual se establece dentro del método como un valor parametrizable. En líneas generales, debe ser un valor de densidad más que un valor fijo de nodos, es decir, debe establecerse una densidad de nodos por unidad de superficie (*nodos/m²*), de tal manera que la exploración del entorno sea similar, aunque varíen las dimensiones del área de trabajo ya que, un número fijo de nodos puede ser insuficiente para la exploración de un entorno de grandes dimensiones y demasiado para dimensiones reducidas.

- b) **Generación de bordes:** La segunda fase, dentro de la exploración del entorno, consiste en establecer todas las rutas posibles libres de obstáculos que comuniquen dos nodos entre sí. Aunque el algoritmo clásico

contempla la comunicación total entre todos los nodos generados, 2 a 2, siempre que exista un camino libre posible, de cara a mejorar la eficacia del algoritmo, en el método propuesto se establece un rango de distancias para considerar que la ruta obtenida entre ambas ubicaciones es válida. De esta manera, introduciendo un valor mínimo de distancia, se evita añadir bordes que comuniquen dos nodos que se encuentran muy próximos entre sí y que no aporten una ruta alternativa interesante desde el punto de vista de la solución global y, introduciendo un valor máximo, se agiliza el proceso de creación de bordes, al evitar la comprobación de si existen dichos bordes entre dos nodos demasiado alejados, cuya probabilidad de comunicarse entre sí sin colisionar con un obstáculo es reducida, sobre todo en entornos estructurados, con una alta presencia de los mismos.

Estos valores de distancia mínima y máxima se configuran en función del tamaño total del mapa del entorno, de tal manera que, al igual que sucede con la densidad de nodos, se mantenga la relación de distancias, aunque el tamaño del área de trabajo a explorar se modifique. El procedimiento a seguir en esta fase consiste en recorrer todos los nodos generados en la fase anterior. En un proceso iterativo, se toma un nodo generado y se va calculando la distancia euclídea, según la Ecuación 3.3, a cada uno de los demás nodos almacenados, comprobando si dicha distancia se encuentra dentro del rango fijado anteriormente. En el caso afirmativo, se procede a comprobar que la línea recta que une dichos nodos se encuentra libre de obstáculos, en cuyo caso se guarda dicho borde para construir posteriormente el grafo final.

$$H(x_n, y_n) = \sqrt{((x_n - x_g)^2 + (y_n - y_g)^2)} \quad (3.3)$$

Este proceso iterativo se repite hasta que se ha comprobado la posible conectividad entre sí de todos los nodos generados. Para comprobar si un borde está libre de colisión, se realiza un procesamiento de la imagen a partir del cual se recorre la recta que une ambos nodos y, se comprueba si dicha recta contiene algún píxel de valor 0, es decir, si dicha recta atraviesa algún obstáculo o no. Además, se incorpora una restricción a la hora de crear el grafo, la cual consiste en impedir que dos nodos correspondientes a una ubicación inicial o final puedan conectarse entre sí, de cara a mejorar la navegación autónoma y coordinada del enjambre, al evitar posibles conflictos con algún UAV detenido en su posición de partida o sobre su ubicación de destino.

Por último, una vez terminada esta fase, se genera la salida final del método de exploración probabilístico, la cual consiste en un grafo con

todos los nodos y bordes creados, es decir, el conjunto de trayectorias posibles, como el que se muestra en la Figura 3.6.

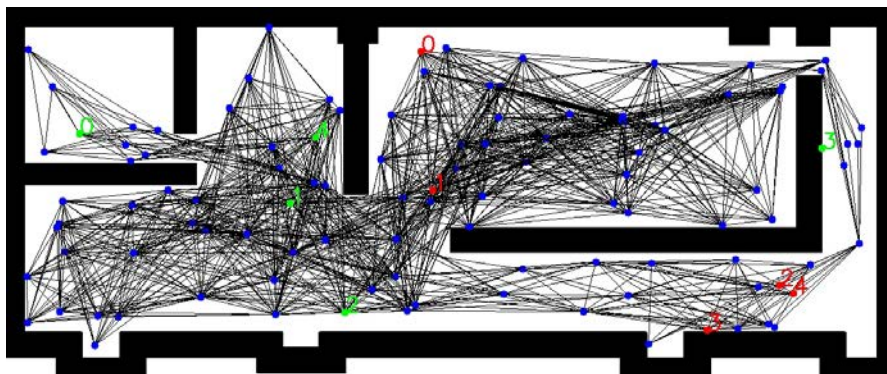


Figura 3.6: Mapa explorado con un algoritmo de PRM de 100 nodos (5) ©2019 IEEE.

3. **Generación de trayectorias:** Por último, la tercera fase del planificador global de trayectorias propuesto consiste en utilizar un algoritmo basado en descomposición de grafo para poder establecer el conjunto de trayectorias óptimas a partir del grafo generado como salida en la etapa anterior. Como se ha indicado previamente, esta etapa es implementada de diferentes formas en función del caso de uso o aplicación a la que se enfoque, pero la base del desarrollo es común para todos. Esta base consiste en emplear el algoritmo A^* para encontrar la manera más eficiente de alcanzar las ubicaciones de destino en cada uno de los casos a estudiar, y que serán detallados en las secciones 3.2.1, 3.2.3 y 3.2.2 de este capítulo. En definitiva esta fase recorre el grafo en busca de aquel conjunto de trayectorias que permite conectar una ubicación inicial con una final minimizando algún parámetro, como puede ser la distancia recorrida por cada UAV, tal y como muestra la Figura 3.7.

En este caso, en el que la planificación global se realiza sobre un entorno en 2D, es necesario, al trabajar con enjambres de UAVs, establecer una altitud de vuelo para las trayectorias óptimas generadas en 2D. Para ello, una vez el método ha generado la solución sobre el plano, se establece una altitud de vuelo para cada uno de los UAV, de tal manera que se garantiza la seguridad de la navegación. De esta forma, aunque dos o más UAVs se muevan hacia un mismo punto de ruta, se establece una diferencia de altitud, asegurando la navegación sin supervisión libre de colisiones.

Una vez se analice el método probabilístico propuesto para 3D y se detallen las implementaciones realizadas para adaptar dicha solución a los diferentes

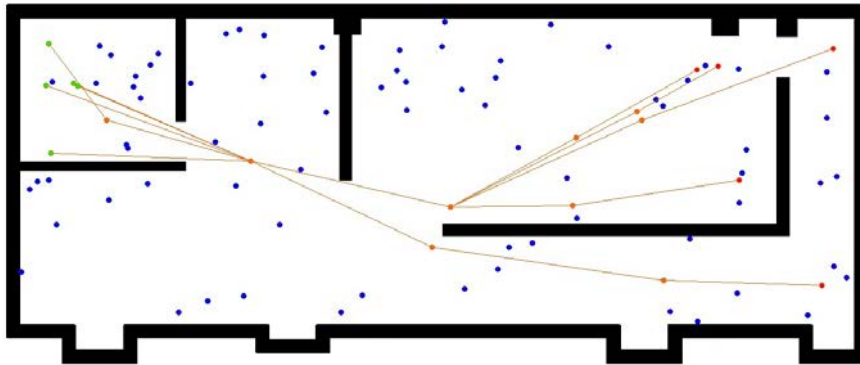


Figura 3.7: Solución óptima final para un enjambre de 5 UAVs (5) ©2019 IEEE.

casos de uso empleados, se establece una comparación y un análisis de los resultados de ambos métodos en la Sección 3.3.

3.2. Planificador de Trayectorias Global en 3D

En esta sección se analiza y detalla el método propuesto para la planificación de trayectorias en 3D para enjambres de UAVs basado en algoritmos de exploración probabilísticos y algoritmos basados en descomposición de grafos para poder generar una solución óptima y segura, permitiendo la navegación sin supervisión de cada uno de los UAVs que forman el enjambre. Como se detalla al comienzo de la Sección 3.1, los desarrollos establecidos previamente, en 2D, sirven como paso previo a la implementación de un método que permita obtener un planificador de trayectorias global a partir de información tridimensional del entorno. La necesidad de trabajar con información del entorno en 3D, como puede ser el caso de los incendios en entornos urbanos o edificios de viviendas, conlleva la introducción de cambios en el método de planificación presentado anteriormente. Dichas modificaciones son descritas a lo largo de la presente sección junto con las implementaciones abordadas para las diferentes aplicaciones en las que, el método propuesto, es aplicado permitiendo a un enjambre de UAVs acometer misiones de manera cooperativa.

En este caso se presenta un conjunto de algoritmos que permiten, a partir de mapas de ocupación en 3D, generar un conjunto de trayectorias óptimas para que, de forma autónoma, un conjunto de UAVs puedan alcanzar distintas localizaciones de interés dentro del área de trabajo. Para ello, como se detalla en esta sección, algoritmos basados en PRM y en Octomap (201) son empleados en conjunto para la exploración del entorno 3D y para establecer el conjunto óptimo de trayectorias,

que permita a cada UAV alcanzar un objetivo específico de la manera más eficiente posible.

La idea de este método, extendido del método de planificación de 2D, es establecer un planificador de trayectorias basado en información 3D que genere una solución óptima y escalable, de cara a ser aplicada en enjambres de UAVs, que puedan operar en diferentes entornos y bajo distintos planteamientos, como pueden ser: la actuación en un caso etiquetado, es decir, aquella situación en la que está preestablecida la relación UAV-tarea; la actuación en un caso no etiquetado, en el que se establece un conjunto de trayectorias para todos los UAVs que minimice la distancia total recorrida por el enjambre; o, por último, la posibilidad de alcanzar todo el enjambre una ubicación determinada bajo una formación concreta.

Como en el caso de la planificación en 2D, al trabajar con un algoritmo basado en muestreo pasivo, es necesario combinar la exploración probabilística del entorno con el algoritmo A* para poder establecer la solución óptima. Al igual que en la planificación en 2D, el método propuesto se conforma de un conjunto de fases, tal y como se sintetiza en la Figura 3.8. Como se puede observar, el esquema seguido es similar al caso de la planificación en 2D, pero con diferencias significativas, tanto en la primera fase de generación de mapa del entorno, como en la implementación del algoritmo de PRM para una exploración del área de en 3D.

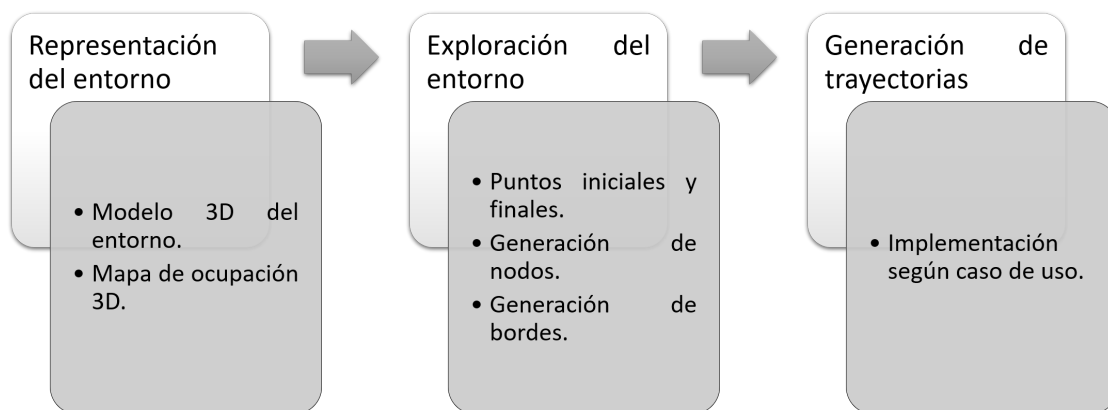


Figura 3.8: Diagrama de flujo para planificación de trayectorias en 3D.

1. **Representación del entorno:** La primera etapa de este método se enfoca a la necesidad de disponer información tridimensional del entorno sobre la

que poder realizar una planificación global de trayectorias para el conjunto del enjambre. A través de dicha representación se establecen los obstáculos presentes en el área sobre el cual se va a desempeñar la misión, lo cual constituye un paso clave a la hora de obtener una planificación de trayectorias eficiente en 3D (202).

La representación 3D del entorno sigue siendo un desafío a día de hoy, debido a que son varias las líneas de investigación que tratan de encontrar una solución que satisfaga este problema. En el método propuesto en este trabajo, se consigue establecer un conjunto de procedimientos que permiten obtener, a partir de un modelo 3D del entorno, un mapa de ocupación completo en 3D sobre el que generar una solución óptima para el despliegue rápido y eficaz de un enjambre de UAVs en áreas de emergencia en entornos urbanos. La aparición de herramientas de software abierto como OpenStreetMap (199) y Osm2Map (203) permiten obtener modelos 3D de ubicaciones reales.

La idea del método propuesto para obtener información 3D del entorno nace a raíz del repaso de los procedimientos presentes en la literatura actual para tratar de resolver este problema. Uno de los métodos presentes en la literatura, se asemeja a lo presentado en la sección anterior, y es lo que se conoce como una representación del entorno 2.5D, es decir, una representación en 2D en el plano del suelo e información parcial en dirección vertical. Esta solución, también conocida como superficie multi-nivel (204), presenta limitaciones derivadas de generar la tercera dimensión a partir de proyecciones, ya que, no sería posible establecer una ruta que tenga como punto inicial una ubicación debajo de un puente y, como punto final, un punto justo encima de dicho puente, ya que ambos comparten la misma proyección y no se pueden representar en un solo nivel transitable (202).

Otra representación en 3D, en este caso completa, es un mapa de transformación de distribuciones normales tridimensionales, es decir, una representación espacial compacta en la que el entorno se representa como un conjunto de distribuciones de probabilidad gaussianas (205). Presentan el inconveniente de que están indexados con *octrees*, cuyos nodos raíz deciden el tamaño del sub-espacio que pueden representar, lo que ocasiona que en ocasiones no se puedan expandir fácilmente y, solo puedan representar un sub-espacio con límites fijos (206).

Por último, se encuentra el tipo de representación empleado en el método propuesto en el presente documento, y que consiste en los mapas de ocupación en 3D, que permite la planificación de trayectorias en entornos 3D para vehículos autónomos tanto aéreos como submarinos (207; 208). Aunque este tipo de representación requiere de una demanda computacional mayor, es un

método extendido en el mundo de la robótica para llevar a cabo soluciones al problema de planificación en 3D.

Para los procedimientos propuestos, se utiliza información procedente de los *Octree*, los cuales se caracterizan por ser una estructura de datos jerárquica que subdivide el espacio en cubos o vóxeles hasta que se alcanza una determinada resolución (209; 210). Una de las propiedades de estos volúmenes es que permite conocer si están libres u ocupados y, con esta información, realizar la exploración del entorno.

Para obtener este mapa de ocupación del área de trabajo, se emplean bibliotecas de procesamiento de modelos 3D como son Binvox (211; 212) y Octomap (201). Así, a partir de un modelo 3D del entorno, se obtiene un mapa de ocupación sobre el que generar un grafo en 3D con todas las posibles rutas para alcanzar las ubicaciones de destino.

Tal y como se muestra en la Figura 3.9, el proceso seguido para obtener este mapa de ocupación u *Octree* consiste en convertir un modelo 3D del entorno a un archivo binvox, el cual es una voxelización binaria de dicho objeto en 3D, que contiene información sobre qué zonas están ocupadas y cuáles están libres. Luego, a través de la biblioteca Octomap, se genera un *Octree* que constituye la entrada para el algoritmo de exploración del entorno. Aunque en el procedimiento habitual se parte de un modelo 3D del entorno, la librería Octomap permite trabajar con nubes de puntos, por lo que otra forma de proceder, sería desplegar en el área de trabajo un conjunto de vehículos equipados con sensores específicos, como pueden ser cámaras estéreo o Light Detection and Ranging (LiDAR), que capturan información 3D del entorno, y mediante algoritmos de cobertura completa y fusión sensorial recopilar datos del entorno y generar un mapa de ocupación completo a partir de la información tridimensional recogida por los vehículos.

2. **Exploración del entorno 3D:** Una vez completada la primera etapa, obteniendo como salida un mapa de ocupación en 3D, se implementa un algoritmo probabilístico basado en PRM para realizar una exploración eficiente del área de trabajo, obteniendo sobre el mismo un conjunto de trayectorias posibles y, construyendo un grafo en 3D sobre el que aplicar, posteriormente, algoritmos de búsqueda óptima de trayectorias.

El método propuesto tiene como principal objetivo reducir el tiempo computacional derivado de la fase de exploración del entorno y de la generación del grafo 3D con el conjunto de todas las trayectorias posibles. Para ello, la implementación desarrollada se configura de tal manera que la fase de exploración se realice una única vez, independientemente del número de UAVs empleados

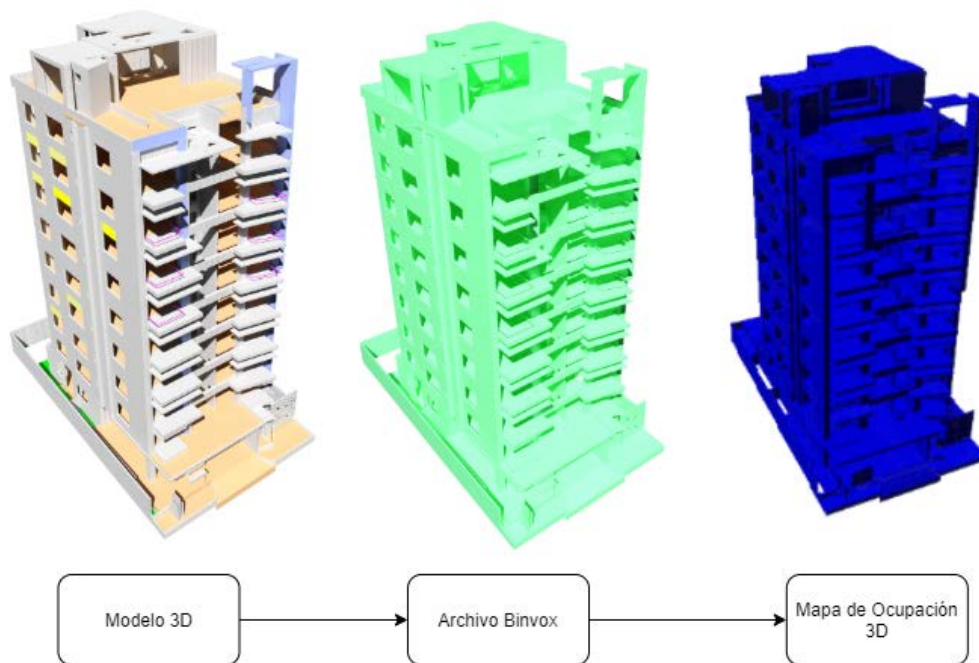


Figura 3.9: Diagrama del proceso de obtención de un mapa de ocupación 3D.

en el enjambre. Por tanto, la fase de búsqueda y construcción de todas las posibles trayectorias, la cual se caracteriza por ser computacionalmente costosa al trabajar con grandes cantidades de información tridimensional, se realiza una única vez, para, a continuación, mediante el uso de algoritmos basados en descomposición de grafos, encontrar la solución óptima al problema de la planificación de trayectorias.

El motivo para emplear un algoritmo probabilístico basado en PRM se debe, en primer lugar, a la capacidad para realizar exploraciones en situaciones de alta complejidad (grandes extensiones de áreas, alto número de UAVs empleados) con un tiempo de ejecución adecuado. En segundo lugar, a que, a diferencia de otros métodos de exploración probabilístico, en el método propuesto se introduce la ventaja de poder reutilizar el grafo 3D generado y, de esta manera, amortizar el coste computacional derivado de la exploración a medida que aumenta el número de UAVs.

En el método de exploración propuesto se introducen un conjunto de variables y parámetros que permiten implementar un método adaptativo, configurable y ventajoso para su uso en enjambres de UAVs. Dichos parámetros son detallados a continuación y, permiten al algoritmo de PRM propuesto trabajar eficientemente con mapas de ocupación 3D, es decir, con grandes cantidades de información sin aumentar de manera exponencial el tiempo

empleado para la exploración del entorno. La posibilidad de definir la densidad del grafo creado lleva a alcanzar un equilibrio óptimo entre el tiempo de cómputo y la optimización de la solución. Dicho equilibrio es importante ya que, si se trabaja con un alto número de nodos el tiempo computacional crece, pero la exploración del área es más completa, mientras que si, por el contrario, el número de nodos se reduce, se obtendrá una búsqueda de rutas más rápida a costa de reducir la calidad de la exploración, lo cual puede ocasionar que el muestreo sea incompleto y que, por tanto, no se pueda alcanzar alguna ubicación concreta, aunque si exista una ruta posible.

Los pasos seguidos para la exploración del entorno 3D se recogen en el Algoritmo 1 y son detallados a continuación. El primer paso dentro de esta etapa es definir un conjunto de variables requeridas por el algoritmo y, que introducen la ventaja de hacer del método propuesto una implementación altamente configurable, de cara a obtener una solución eficiente en términos de tiempo de cómputo y minimización de la distancia recorrida para un enjambre de UAVs.

- I_{pnts} : Archivo que contiene la posición inicial de los UAVs. En este caso, estas posiciones se establecen con respecto al origen de coordenadas del mapa y son fruto de una conversión de las coordenadas globales (*longitud*, *latitud* y *altitud*) a un sistema de referencia local, como puede ser la posición inicial de uno de los UAV. De esta manera, se establece un sistema de referencia común para todos los agentes implicados en la misión y, se obtiene la posición relativa entre los diferentes agentes.
- G_{pnts} : Archivo que contiene las ubicaciones de los objetivos a alcanzar por el enjambre. Al igual que en las posiciones iniciales, estas coordenadas aparecen en metros con respecto al origen de coordenadas del mapa. Dichas coordenadas son referenciadas en el algoritmo a un sistema local común, cuyo origen es la posición inicial de uno de los UAV.
- $X_{max}, X_{min}, Y_{max}, Y_{min}, Z_{max}, Z_{min}$: Conjunto de valores máximos y mínimos, en cada eje de coordenadas, que puede tomar como valor un nodo durante la generación aleatoria de los mismos. Estos valores se fijan considerando el tamaño del *Octree*.
- $Nodes$: Variable que permite parametrizar la densidad de nodos empleados para explorar el entorno. Es importante destacar que, como se ha incidido anteriormente, este valor debe establecer un equilibrio entre el tiempo de cálculo y una alta posibilidad de encontrar todas las soluciones posibles.

Algorithm 1: PRM en 3D.

```

Input: Mapa de ocupación 3D map
Output: Grafo en 3D (Figura 3.10).
1 Define: I_pnts Archivo de ubicaciones iniciales, G_pnts Archivo de
   ubicaciones objetivo, nodes vector de nodos, edges vector de bordes,
   RAND_MAX valor aleatorio máximo, Xmax Coordenada Máxima en X,
   Ymax Coordenada Máxima en Y, Xmin Coordenada Mínima en X, Ymin
   Coordenada Mínima en Y, nNodes Número de Nodos, octree Archivo Octree,
   Dmax Valor de Distancia Máxima y Dmin Valor de Distancia Mínima
2 begin
3   //Generación de nodos para exploración del entorno
4   nodes.push_back(I_pnts)
5   nodes.push_back(G_pnts)
6   Node_new()
7   while counter < nNodes do
8     Node_new.x = Xmin + ((Xmax - Xmin) · (rand() / (RAND_MAX +
9       1.0)))
10    Node_new.y = Ymin + ((Ymax - Ymin) · (rand() / (RAND_MAX +
11      1.0)))
12    Node_new.z = Zmin + ((Zmax - Zmin) · (rand() / (RAND_MAX +
13      1.0)))
14    //Comrpobar si el nodo existe y si está en espacio libre
15    OcTreeNode n = octree.search(Node_new)
16    if n then
17      bool occupied = octree.isNodeOccupied(n)
18      if !occupied then
19        nodes.push_back(Node_new)
20        counter++
21
22  //Generación de todas las rutas libres de colisiones posibles
23  for k = 0; k < nodes.size(); k++ do
24    for l = i+1; l < nodes.size(); l++ do
25      diff_X = nodes[k].x - nodes[l].x
26      diff_Y = nodes[k].y - nodes[l].y
27      diff_Z = nodes[k].z - nodes[l].z
28      euclideanDistance = sqrt(diff_X · diff_X + diff_Y · diff_Y
29        + diff_Z · diff_Z)
30      if euclideanDistance >= distMin y euclideanDistance <=
31        distMax then
32        //Comprobar que la ruta entre nodos está libre
33        vector<point3d> ray
34        point3d vec, endRay
35        vec.x() = -diff_X,    vec.y() = -diff_Y,    vec.z() =
36          -diff_Z
37        bool occupancyPath = octree.castRay(nodes[k], vec, endRay,
38          true, euclideanDistance)
39        if !occupancyPath then
40          Edge_new(nodes[k], nodes[l], euclideanDistance)
41          edges.push_back(Edge_new)
42          octree.computeRay(nodes[k], nodes[l], ray)
43
44  Dibujar grafo

```

- D_{max}, D_{min} : Para mejorar la eficiencia en la creación del grafo, se introduce como novedad en el método propuesto, estas variables que permiten configurar la longitud de los bordes generados, los cuales son posibles caminos entre dos ubicaciones, dentro de un rango de distancias dependiente de las dimensiones del área a explorar. Esta mejora permite, por un lado, eliminar posibles rutas entre dos nodos muy cercanos, lo cual no aportaría una ruta alternativa nueva, mientras que, por otro lado, introducir un valor de distancia máxima, origina que no se considera ni estudie la posibilidad de conectar dos nodos muy alejados cuya posibilidad de hacerlo a través de un borde libre de colisiones es baja. De esta manera, se reduce el número de iteraciones en las que se debe procesar información tridimensional del entorno para determinar si existe ese borde libre de colisiones entre dos nodos y, con ello, el tiempo de computación empleado para generar el grafo 3D.

Una vez parametrizado el algoritmo, se avanza a una segunda etapa, en la que el algoritmo de exploración del entorno se divide en un conjunto de implementaciones que permiten generar un conjunto finito de posibles rutas, dando como salida un grafo en 3D. Dichas implementaciones son:

- a) **Generación de nodos:** Tras comprobar que las ubicaciones de destino se encuentran dentro del espacio libre y añadir al grafo los nodos correspondientes a dichas ubicaciones y a las ubicaciones de partida de los `glspluav`, se inicia un proceso iterativo para la creación de nodos aleatorios sobre el entorno en 3D.

Para la creación aleatoria de los nodos se emplean las ecuaciones descritas en la planificación en 2D, añadiendo, para este caso, la coordenada Z . De nuevo, haciendo uso de librerías estándar de C++ y, de la función `rand()` y la macro constante `RAND_MAX`, se genera un valor aleatorio dentro de los valores máximos y mínimos definidos para cada una de las coordenadas.

Una vez generadas las coordenadas aleatorias del nodo, se comprueba si este nodo se encuentra dentro de un *vxel* libre del *Octree*. Dado que el mapa de ocupación es una estructura jerárquica en forma de árbol, en el que cada *vxel* está compuesto a su vez por 8 *vxeles* de menor tamaño hasta alcanzar la resolución mínima, es necesario buscar a cuál de ellos pertenece el nodo creado y, a continuación, comprobar si dicho *vxel* se encuentra libre u ocupado, mediante el uso de la librería *Octomap*. En caso de que el nodo se encuentre en espacio libre, se almacena como nodo perteneciente al grafo 3D y, se repite el proceso de manera iterativa hasta generar el número total de nodos.

- b)* **Generación de bordes:** El siguiente paso consiste en establecer todas las conexiones posibles entre los nodos aleatorios almacenados en el grafo. Para ello, el primer paso es verificar si la distancia euclídea entre los nodos se encuentra dentro del rango establecido. En caso de que se cumpla la condición, se realiza una comprobación de si el camino entre ambos nodos se encuentra libre de obstáculos mediante el uso de la librería Octomap, la cual posee funciones que permiten lanzar rayos desde un nodo a otro y comprobar si dicho rayo colisiona con algún obstáculo o no. Además, la función empleada considera otros aspectos que permite dar robustez a este proceso, como son poder comprobar si el final del rayo es el nodo de destino o, si la distancia recorrida por el rayo coincide con la distancia entre ambos nodos. Una vez realizada la verificación, y en caso de que dicho camino no posea obstáculos, el borde generado se almacena y añade al grafo final en 3D.

Este proceso se repite de manera iterativa hasta comprobar las conexiones, 2 a 2, entre todos los nodos generados en el paso anterior.

Completada la fase de exploración, el algoritmo basado en PRM genera un grafo en 3D que constituye la entrada para la siguiente fase, en la cual el uso del algoritmo A* permite la obtención de las trayectorias óptimas para que cada UAV del enjambre alcance su ubicación de destino. En la Figura 3.10 se puede observar un ejemplo de grafos en 3D para diferentes entornos de distintas dimensiones y características. En la Figura 3.10a se muestra la exploración 3D para una sola estancia, dónde los obstáculos se muestran en azul y el conjunto de posibles caminos en verde, mientras que en la Figura 3.10b, el escenario consiste en un edificio de grandes dimensiones, formado por 9 plantas y múltiples estancias, en el que los obstáculos se muestran en negro, mientras que el conjunto de caminos posibles aparecen en rojo.

3. **Generación de trayectorias:** Siguiendo el esquema presentado anteriormente, el método propuesto posee una tercera etapa destinada a la generación de una solución óptima, en cuanto a distancia total recorrida, para el problema de planificación de trayectorias en 3D. A partir del grafo 3D obtenido en la fase previa, se establecen cual es, para cada UAV, la ruta óptima.

Como en el caso del método de planificación en 2D, la base de la implementación es el uso del algoritmo A* para encontrar aquella ruta que minimiza la distancia recorrida por el UAV mientras navega hacia el objetivo. Esta fase tiene distintos procedimientos según sea la situación a la que se aplica el planificador global de trayectorias, tal y como se detalla a continuación en las Subsecciones 3.2.1, 3.2.3 y 3.2.2.

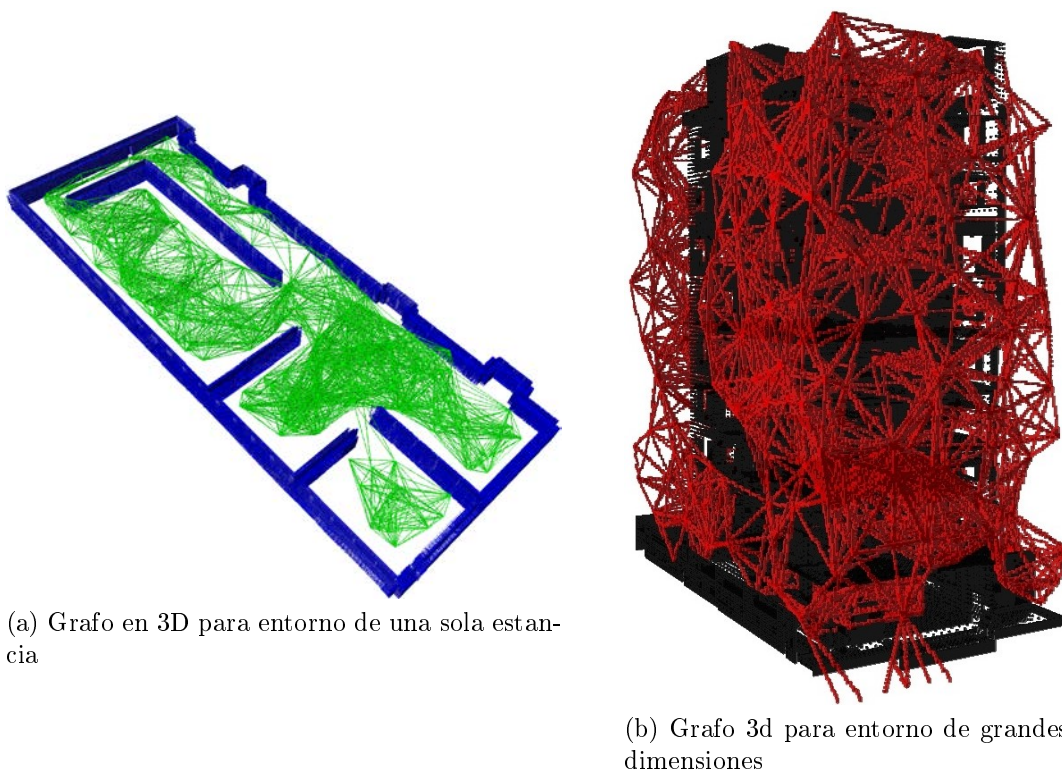


Figura 3.10: Salida generada por algoritmo de exploración de entornos en 3D.

3.2.1. Planificación de trayectorias para caso etiquetado

Se entiende por caso etiquetado aquella situación en la que se ha establecido una asignación de tareas previa a la planificación de trayectorias, es decir, se conoce la ubicación de destino de cada uno de los UAVs del enjambre y, por tanto, el punto inicial y final sobre el que el algoritmo A^* debe encontrar la ruta óptima.

En el caso de los enjambres homogéneos, en el que todos los UAVs poseen las mismas características, esta asignación puede ser secuencial, es decir, el primer objetivo puede ser asignado al primer UAV del enjambre, el segundo al UAV número 2 y, así sucesivamente, dado que todos los UAVs participantes son similares y no existe diferencia alguna entre que uno u otro alcance una ubicación determinada.

En el caso de los enjambres heterogéneos, en el que los UAVs participantes en la misión poseen diferentes características, se da la circunstancia de que esta diferenciación puede hacer que algún agente sea más adecuado para unas tareas que otros o, que posea cualidades que le permitan alcanzar una ubicación remota en menor tiempo o, simplemente poseas alguna propiedad que permita que un objetivo sea alcanzable para dicho UAV pero inalcanzable para el resto. En este caso, es necesario un algoritmo previo que considere todos estos condicionantes y

establezca qué UAV es el mejor posicionado para cubrir una determinada tarea. Este algoritmo se conoce como Multi-Robot Task Allocation (MRTA) (213) y, es el encargado de asignar la combinación óptima UAV-tarea considerando aspectos como la velocidad del UAV, la urgencia de la misión, la autonomía, la capacidad de carga o la adecuación de los sensores o carga de pago instalada en el UAV con la tarea a acometer (10). Por tanto, en esta situación es necesaria la combinación de una etapa previa basada en algoritmos de MRTA con métodos de planificación de trayectorias, para establecer los pares UAV-tarea óptimos y, a continuación, encontrar la forma más eficiente de alcanzar dicho objetivo.

Las implementaciones y validaciones recogidas en el presente trabajo se orientan al uso de enjambres de UAVs homogéneos, en el que todos los UAV cuentan con características similares. Por dicho motivo, cuando se habla de caso etiquetado se establece una situación en la que se lleva a cabo una asignación secuencial, en la que se establece que el UAV_1 acudirá a la tarea 1 y, así sucesivamente, hasta completar todas las tareas. Por tanto, el algoritmo A^* se encarga de, a través del grafo creado previamente, encontrar la ruta óptima en cuanto a distancia recorrida que permita conectar cada UAV con su destino.

3.2.2. Planificación de trayectorias para caso no etiquetado

El caso no etiquetado requiere más complejidad que el caso etiquetado, ya que, a diferencia del anterior, no existe una asignación previa de tareas o ubicaciones de destino para cada agente del enjambre. Por tanto, deben introducirse métodos adicionales para establecer dicha asignación de manera óptima. En el presente trabajo, la implementación se centra en encontrar una solución para cada UAV considerando la distancia total recorrida por el enjambre como aspecto clave a minimizar.

Por tanto, es necesario establecer un método que permita realizar una asignación entre dos categorías diferentes (UAV - ubicación de destino, para este caso), minimizando un factor común (distancia total recorrida por el enjambre). Para hacer frente a este problema, el método húngaro (214; 215; 216) permite minimizar aspectos en problemas de asignación, por lo que en este caso no etiquetado, se introduce una fase intermedia que encargada de realizar dicha asignación óptima mediante el empleo del método húngaro, para minimizar la distancia total recorrida por el enjambre. Por lo tanto, una vez aplicado el método, se produce una asignación de tareas y se obtiene un conjunto óptimo de trayectorias para que cada UAV alcance su objetivo.

Es importante considerar, en primer lugar, que para esta asignación se establece que el número de UAVs participantes en el enjambre es igual al número de ubicaciones distintas a alcanzar, es decir, que cada UAV acude a una única ubicación

diferente (N). En segundo lugar, y en relación con el primer punto, es necesario considerar que cuando el número de agentes del enjambre aumenta, la complejidad del problema a resolver crece exponencialmente, tal y como se detalla en la Figura 3.11. Dicha figura muestra que, para dos parejas UAV-tarea, el método húngaro debe procesar una matriz de 2×2 con cuatro posibles soluciones. Sin embargo, en el caso de que sean cinco parejas, existen 25 combinaciones posibles, es decir, el método húngaro trabaja y procesa N^2 posibles soluciones.

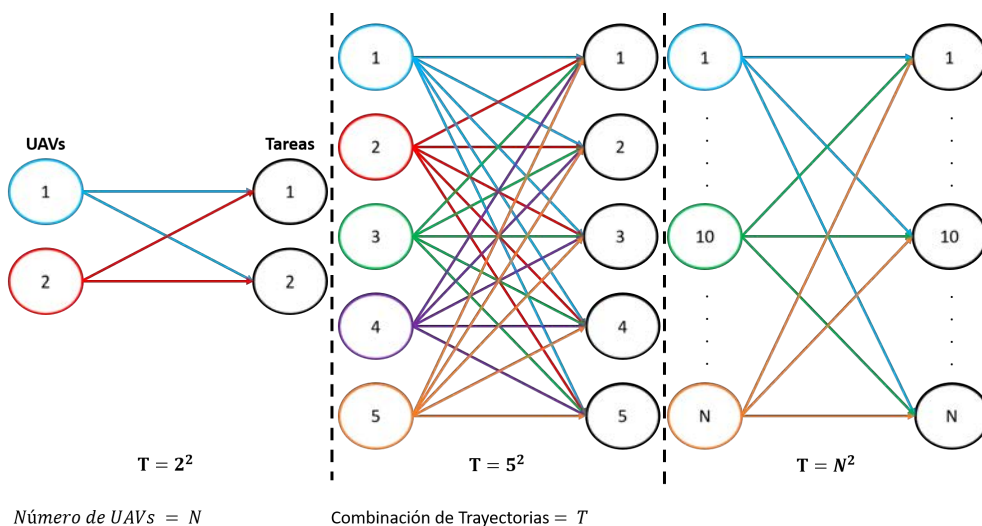


Figura 3.11: Método Húngaro

Para poder aplicar el método húngaro, minimizando la distancia total recorrida, es necesario implementar primero el algoritmo A* para obtener la ruta óptima de cada UAV a cada uno de los objetivos, si es que dicha ruta existe. Una vez se posee la matriz que relaciona todos los agentes con todos los destinos, se aplica el método húngaro y se obtiene la solución óptima que permite a cada UAV alcanzar una ubicación minimizando la distancia total recorrida por el enjambre. Es importante considerar que este método genera una solución óptima en la que todas las tareas son alcanzadas, es decir, aunque para un agente la distancia a un objetivo no sea la menor en comparación con el resto de objetivos, si esa pareja UAV-tarea es la única que permite alcanzar esa ubicación, es elegida por encima de otras ubicaciones con menor distancia a recorrer.

3.2.3. Planificación de trayectorias para formación de enjambres

Este caso de estudio se orienta a aquellas situaciones en las que todos los UAVs del enjambre navegan hacia una misma ubicación y, en concreto, deben alcanzar

dicho objetivo bajo una estructura predeterminada.

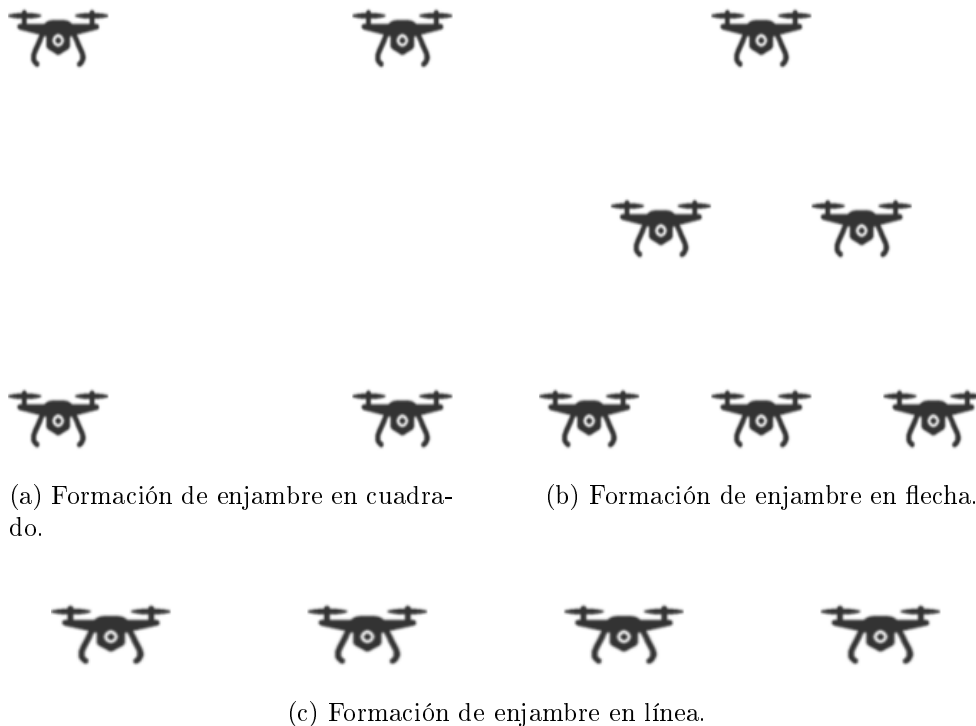
Este tipo de formación, cuyo término deriva del mundo animal, aparece como resultado de un comportamiento cooperativo de los miembros del sistema global, donde cada uno de los elementos mantiene una distancia y orientación específica con respecto al movimiento de los demás. Este comportamiento genera un conjunto de ventajas con respecto al uso de un sólo elemento, como son la capacidad de generar y modificar formaciones de forma autónoma y coordinada para adaptarse de una manera más adecuada al entorno en tiempo real (165).

En este caso, el desarrollo realizado a lo largo de esta investigación para poder establecer una solución a la planificación de trayectorias se divide en dos fases. Una primera fase donde se establece el tipo de formación que se desea alcanzar, junto con la posición específica de cada UAV dentro de la misma y, una segunda fase, en la que se genera el conjunto de trayectorias que permiten al enjambre alcanzar una ubicación bajo dicha formación, haciendo uso del algoritmo A*.

Buscando que esta auto-organización permita resolver tareas propias derivadas de la lucha contra el fuego con un mejor rendimiento, maximizando la cobertura espacial o minimizando el tiempo de respuesta de la misión, se proponen tres tipos de formaciones diferentes: línea, cuadrado o flecha, tal y como se muestra en la Figura 3.12. Aunque cada formación tiene aspectos particulares en su implementación, los cuales se detallan a continuación, todos poseen un conjunto de parámetros comunes que actúan como entradas del algoritmo y que son: el número de agentes disponibles para la formación, el tamaño de los agentes que permite establecer la separación entre UAVs dentro de la misma y, por último, la ubicación a la cual se desea llegar con dicha formación, a partir de la cual, se establecen las ubicaciones finales de todos los agentes del enjambre.

1. **Formación en línea:** El algoritmo implementado para establecer una formación en línea, sobre una ubicación concreta, contempla la posibilidad de que dicha formación sea tanto horizontal como vertical. La ubicación de interés marca la posición de uno de los agentes del enjambre y es, a partir de dicha ubicación, sobre la cual se calculan el resto de posiciones de los UAVs que forman parte de la formación.

Para ello, tal y como se muestra en el Algoritmo 2 se introduce un valor de separación (*sep*) entre agentes, el cual, como norma general, vendrá determinado por el tamaño de los UAVs, pero que puede ser parametrizado por otros aspectos a considerar, como puede ser características de la carga de pago instalada en cada uno de los agentes. En el caso del tamaño, un aspecto importante a considerar es si se trabaja con un enjambre homogéneo o heterogéneo. En el primer caso, en el que el tamaño de los agentes es similar, se puede establecer esta separación como un valor dependiente de dicho



(a) Formación de enjambre en cuadrado.

(b) Formación de enjambre en flecha.

(c) Formación de enjambre en línea.

Figura 3.12: Tipos de formaciones propuestas para vuelo en enjambre.

tamaño más un margen de seguridad adecuado al mismo. En el segundo caso, en el que el tamaño de cada UAV puede diferir, dicha separación vendrá determinada por el tamaño del agente más grande del enjambre. Este valor o, un valor proporcional del mismo es sumado o restado con respecto a la posición inicial para establecer el resto de posiciones del enjambre.

La implementación abordada en este trabajo y, resumida en el Algoritmo 2 permite que la posición objetivo se encuentra siempre en el centro de la formación y, que el resto de agentes del enjambre se vayan posicionando a un lado u otro de esta ubicación.

Una vez asignada está posición final como referencia para el resto de posiciones, se define el umbral de separación entre los mismos y se analiza si la línea a formar es horizontal o vertical. Acto seguido, se inicia un proceso iterativo que se repite tantas veces como UAVs formen el enjambre y, que tiene como objetivo, calcular y asignar a cada UAV su ubicación final dentro de la formación. En dicho proceso destaca el uso de la variable *orden*, la cual permite conocer si la posición a calcular se encuentra a la izquierda o a la derecha de la ubicación objetivo empleada como referencia. Dicha variable, se caracte-

Algorithm 2: Formación en línea

Input: Posición del objetivo O , Tipo de línea L_t , Número de agentes n_A ,
Tamaño del agente A_s

Output: Posiciones en la formación F_p

```

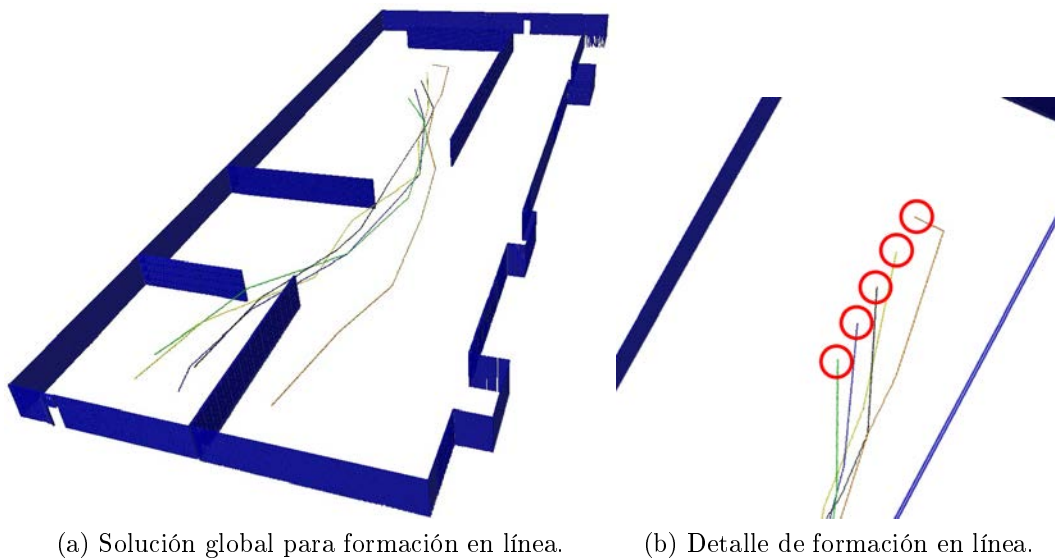
1 begin
2    $F_{p0} = O$ 
3    $sep = 3 \times A_s$ 
4   if  $L_t = vertical$  then
5     for  $nthCounter \leftarrow 2$  to  $n_A$  do
6        $F_{py} = O_{py}$ 
7        $F_{pz} = O_{pz}$ 
8        $orden = nthCounter / 2$ 
9       if  $nthCounter \% 2 == 0$  then
10         $F_{px} = O_{px} - orden \times sep$ 
11      else
12         $F_{px} = O_{px} + orden \times sep$ 
13    else
14      for  $nthCounter \leftarrow 2$  to  $n_A$  do
15         $F_{px} = O_{px}$ 
16         $F_{pz} = O_{pz}$ 
17         $orden = nthCounter / 2$ 
18        if  $nthCounter \% 2 == 0$  then
19           $F_{py} = O_{py} - orden \times sep$ 
20        else
21           $F_{py} = O_{py} + orden \times sep$ 

```

riza por ser siempre un entero obtenido a partir de la división del número de posición a calcular entre dos, es decir, una vez asignada la primera posición de la formación al objetivo marcado, las siguientes posiciones a cubrir y, para las cuales la variable $orden$ toma el valor igual a 1, se establecen con una separación sep a izquierda y derecha de la primera posición marcada. En el caso de que el enjambre cuente con más agentes, la variable $orden$ se toma el valor 2 para los dos siguientes agentes y, se establecen las posiciones cuarta y quinta de la formación, las cuales se caracterizan por encontrarse a una distancia de 2 veces sep a izquierda y derecha de la ubicación inicial. Este proceso se repite de manera iterativa hasta establecer todas las posiciones que ocuparán los UAVs que forman el enjambre.

Aunque la distancia con respecto a la posición inicial no es el único aspecto a estudiar ya que es, también, importante mantener el orden en el cual se va completando la formación. Por este motivo, se introduce una condición que permite conocer si la posición a calcular corresponde a una posición par o impar, de tal manera que se establezca a la izquierda o a la derecha del objetivo y, por tanto, se mantenga un orden de colocación.

Finalizado el proceso iterativo con el conjunto de posiciones finales del enjambre (F_p), la situación que se deriva puede ser tomar la vertiente de caso etiquetado, si se asigna cada posición de manera secuencial a cada agente o, por el contrario, si se desea obtener la solución más eficaz en cuanto a distancia total recorrida se puede considerar el problema como un caso no etiquetado y, asignar a cada UAV aquella posición dentro de la formación que minimice la distancia total recorrida por el enjambre para alcanzar dicha formación. En el presente trabajo se ha considerado que una vez calculadas todas las posiciones de la formación, el planificador de trayectorias se encuentra con un caso no etiquetado, de cara a optimizar la solución y minimizar la distancia total recorrida y, con ello, los tiempos de respuesta, aspecto clave en la lucha contra el fuego. Por tanto, a partir del procedimiento descrito en el caso no etiquetado se establece la ruta óptima para cada UAV del enjambre, generando un conjunto de trayectorias que permitan alcanzar la formación en línea deseada tal y como se muestra en la Figura 3.13.



(a) Solución global para formación en línea.

(b) Detalle de formación en línea.

Figura 3.13: Ejemplo de formación en línea horizontal.

2. **Formación en cuadrado:** En este caso, se establece un algoritmo que permita establecer una lista de posiciones para que los UAVs alcancen una formación en cuadrado. Para dicha situación y, a diferencia que la formación en línea, el posicionamiento de los agentes con respecto a la posición objetivo varía en ambos ejes del plano, de tal manera, que la ubicación inicial de destino se establece siempre como la esquina superior izquierda del cuadrado.

El proceso empleado para el cálculo de las posiciones de esta formación genera una secuencia que permite, en primer lugar, completar las cuatro esquinas del cuadrado y, en segundo lugar, ir rellenando los segmentos que unirían dos vértices consecutivos entre sí, de tal forma que, el interior del cuadrado siempre estaría libre de agentes, tal y como se puede ver en la Figura 3.14.

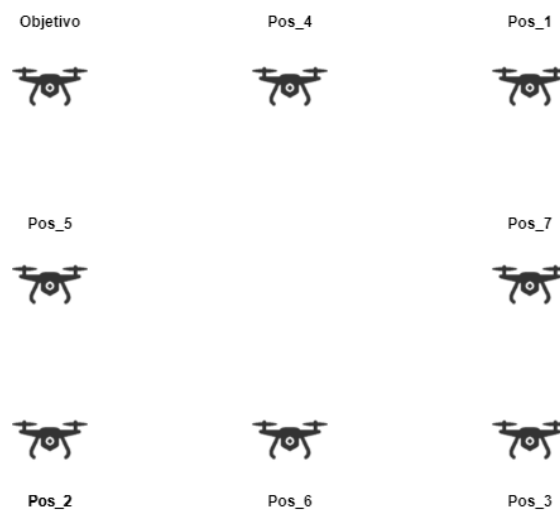


Figura 3.14: Orden de asignación de posiciones en la formación en cuadrado.

El algoritmo implementado debe, por tanto, calcular, en función del número de agentes, la distancia entre esquinas y, a continuación, las distancias intermedias de tal forma que sean equidistantes, es decir, que, como se muestra en la Figura 3.14, en caso de existir un agente entre las esquinas, este se encuentre en el centro del segmento. En caso de que haya dos agentes en cada segmento, la distancia entre la primera esquina y el primer agente intermedio debe ser la misma que desde este al segundo y, de este segundo agente a la segunda esquina. De esta forma, a medida que el número de agentes aumenta, la distancia entre ellos será equidistante y, los segmentos que forman los bordes del cuadrado quedarán divididos a partes iguales. El método empleado se describe de manera completa en el Algoritmo 3.

El primer paso consiste en definir y establecer los parámetros que definen

la formación en cuadrado en función del número de agentes y del tamaño máximo de los mismos. De esta forma, haciendo uso de la ecuación 3.4, se calcula el número de centros ($nCentros$), o lo que es lo mismo, de agentes que se incluyen en cada uno de los segmentos. Para ello, se toma el entero superior resultante de la operación recogida en dicha ecuación, la cual considera el número de agentes y las esquinas y segmentos que forman el polígono regular a formar, que para este caso concreto se establece en 4, pero que permite extrapolar esta función a futuras formaciones.

$$nCentros = \text{ceil}\left(\frac{n_A - nEsquinas}{nLineas}\right) \quad (3.4)$$

En el caso de que el enjambre esté formado por 4 o menos agentes, el número de centros es 0 y, por tanto, solo es necesario establecer la posición de las esquinas del cuadrado que, en este caso, se ha parametrizado en función del tamaño máximo de UAV. En caso de que el enjambre posea más de 4 agentes y, por tanto, exista al menos un centro en uno de los segmentos, la distancia entre vértices se amplía considerando este aspecto, mediante el cálculo de la variable $disEsquinas$. Una vez definida la distancia entre los vértices del cuadrado, se calcula la distancia entre los agentes ubicados en los segmentos ($disPos$) que establecen el perímetro del cuadrado.

El resto del Algoritmo 3 recoge el cálculo del conjunto de posiciones de los UAVs en la formación. Para ello, se inicia un proceso iterativo que concluye cuando se han establecido todas las posiciones objetivo de los agentes del enjambre. Como se ha indicado, el proceso de composición de la formación se realiza de tal manera que en primer lugar se establezcan y calculen las posiciones relativas al resto de vértices que forman el cuadrado. Una vez completado este paso, y en caso de que el enjambre posea más de 4 UAVs, se procede a ir estableciendo los centros en cada uno de los segmentos.

Para ello, y al igual que en la formación en línea, es importante definir la variable *orden* que considere el punto en el que se encuentra la formación, es decir, conocer cuál es la siguiente posición a calcular y, si se ha establecido un agente en cada uno de los cuatro vértices del cuadrado, momento en el cual se debe iniciar una nueva vuelta donde la siguiente ubicación es un centro del primer segmento. De esta forma, se puede conocer cuál es la posición relativa a la formación que se pretende calcular en cada iteración e ir completando la formación hasta completar el cálculo de posiciones para todos los UAVs del enjambre.

Algorithm 3: Formación en cuadrado**Input:** Posición objetivo O , Número de agentes n_A , Tamaño agente A_s **Output:** Posiciones en la formación F_p

```

1 begin
2    $F_{p0} = O$ 
3    $nEsquinas = nLineas = 4$ 
4    $nCentros = \text{ceil}(\frac{n_A - nEsquinas}{nLineas})$ 
5   if  $nCentros \neq 0$  then
6      $disEsquinas = (4 \times A_s \times nCentros) + A_s$ 
7   else
8      $disEsquinas = 4 \times A_s$ 
9    $disPos = disEsquinas / nCentros + 1$ 
10  for  $nthCounter \leftarrow 2$  to  $n_A$  do
11    if  $nthCounter < 4$  then
12      if  $nthCounter == 1$  then
13         $F_{px} = O_{px} + disEsquinas$ 
14         $F_{py} = O_{py}$ 
15         $F_{pz} = O_{pz}$ 
16      else
17        if  $nthCounter == 2$  then
18           $F_{px} = O_{px}$ 
19           $F_{py} = O_{py} - disEsquinas$ 
20           $F_{pz} = O_{pz}$ 
21        else
22           $F_{px} = O_{px} + disEsquinas$ 
23           $F_{py} = O_{py} - disEsquinas$ 
24           $F_{pz} = O_{pz}$ 
25      else
26         $orden = nthCounter / 4$ 
27        if  $nthCounter == 4 \times orden$  then
28           $F_{px} = O_{px} + orden \times disPos$ 
29           $F_{py} = O_{py}$ 
30           $F_{pz} = O_{pz}$ 
31        else
32          if  $nthCounter == 4 \times orden + 1$  then
33             $F_{px} = O_{px}$ 
34             $F_{py} = O_{py} - orden \times disPos$ 
35             $F_{pz} = O_{pz}$ 
36          else
37            if  $nthCounter == 4 \times orden + 2$  then
38               $F_{px} = O_{px} + orden \times disPos$ 
39               $F_{py} = O_{py} - disEsquinas$ 
40               $F_{pz} = O_{pz}$ 
41            else
42               $F_{px} = O_{px} + disEsquinas$ 
43               $F_{py} = O_{py} - orden \times disPos$ 
44               $F_{pz} = O_{pz}$ 

```

Una vez calculadas y generadas todas las posiciones objetivo para la formación del enjambre, se sigue el procedimiento del caso no etiquetado y, combinando el algoritmo A* junto con el método húngaro, se obtiene la solución óptima a la combinación de trayectorias que minimiza la distancia total recorrida por el enjambre, como la que se muestra en la Figura 3.15.

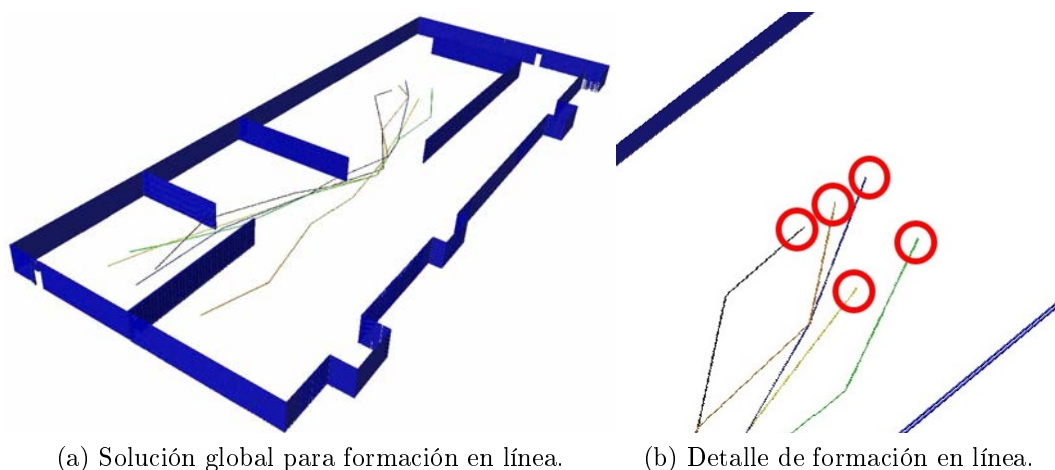


Figura 3.15: Ejemplo de formación en cuadrado.

3. **Formación en flecha:** Este tipo de formación sigue una metodología similar a las anteriores, tomando idénticas entradas que la formación en cuadrado y generando como resultado la lista de posiciones finales de dicha estructura. Esta formación puede ser dividida en filas dentro de la cuales se establece un número n de agentes, los cuales van incrementando en 1 en cada nueva fila, teniendo, por tanto, un agente en la fila 1, dos agentes en la fila 2 y, así sucesivamente. La secuencia que representa el número total de agentes en el enjambre dependiendo del número de fila se recoge en la ecuación 3.5. Dicha ecuación se puede reorganizar y obtener, a través de la fórmula cuadrática, el número de filas que forman la formación en función del número de agentes.

$$N_A = \frac{row \times (row + 1)}{2} \quad (3.5)$$

El Algoritmo 4 muestra la metodología seguida para obtener el conjunto de posiciones dentro de la formación. En primer lugar, se establece un parámetro de separación entre posiciones, el cual viene determinado por el tamaño del máximo de UAV. Acto seguido, se calcula el número de filas (n_{Filas}) que forman la formación final en flecha en función del número de agentes, organizando y resolviendo la ecuación cuadrática 3.5.

Algorithm 4: Formación en flecha**Input:** Posición objetivo O , Número de agentes n_A , Tamaño agente A_s **Output:** Posiciones en la formación F_p

```

1 begin
2    $F_{p0} = O$ 
3    $sep_x = sep_y = 1,5 \times A_s$ 
4    $nFilas = \frac{-1 + \sqrt{1 + 8 \times n_A}}{2}$ 
5    $aux = 1$ 
6   for  $nthCounter \leftarrow 1$  to  $nFilas$  do
7     for  $nthCounter2 \leftarrow 0$  to  $i \lll aux < n_A$  do
8       if  $nthCounter2 == 0$  then
9          $F_{px} = F_{px}Previo - sep_x/2$ 
10         $F_{py} = F_{py}Previo - sep_y$ 
11         $F_{pz} = O_{pz}$ 
12       else
13          $F_{px} = F_{px}Previo + sep_x$ 
14          $F_{py} = F_{py}Previo$ 
15          $F_{pz} = O_{pz}$ 
16        $a++$ 

```

Una vez el número de fila es obtenido, se va recorriendo cada una de ellas y estableciendo la posición de cada uno de los UAV en dicha fila, sabiendo que el número total de agentes por fila es igual al número de fila en el que se encuentra. De esta manera, sólo es necesario comprobar cuando se cambia de fila y cuál es el primer agente de dicha fila ya que, en dicho caso, su posición varía tanto en el eje x como en el eje y , mientras que en el resto de posiciones de la fila, solo se establecen modificaciones en la coordenada x . A diferencia de las formaciones anteriores, el cálculo se va efectuando con respecto a las posiciones previas calculadas y no en relación con el objetivo inicial.

En el caso de que la nueva posición a calcular corresponda con el primer agente de la fila, su posición en el eje x se establece restando la mitad de la separación fijada en este eje, mientras que en el eje y se resta la separación completa fijada para dicho eje. En el caso del resto de posiciones de la fila, se añade la separación fijada en el eje x con respecto a la posición previa, pero se mantiene la posición en y , pues la formación de la flecha o pirámide solo se contempla de manera vertical y no horizontal.

Una vez completadas todas las filas de la flecha, se continúa el proceso im-

plementado en el resto de formaciones y que consiste en aplicar el algoritmo A* junto al método húngaro para encontrar la combinación de trayectorias que minimiza la distancia total recorrida por el enjambre, como ocurre con el método no etiquetado. La solución generada para este tipo de formación se muestra en la Figura 3.16, dónde se recoge un mapa 3D con las diferentes trayectorias de cada uno de los UAVs.

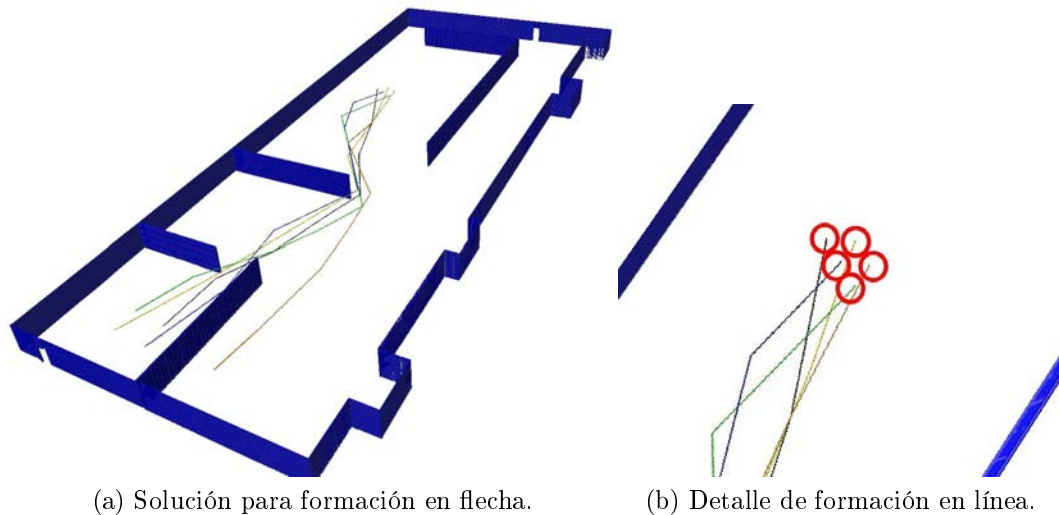


Figura 3.16: Ejemplo de formación en cuadrado.

En esta sección se han presentado y analizado un conjunto de procedimientos que permiten establecer una solución óptima a la planificación de trayectorias, aplicable sobre un entorno en 2D y 3D, para que un enjambre de UAVs pueda desplegarse en el área de trabajo bajo una formación concreta y, posteriormente, poder realizar una navegación coordinada bajo dicha estructura.

Aunque si bien es cierto que, el tipo y cantidad de formaciones que un enjambre de UAVs puede alcanzar es muy extenso y diversificado, en el presente trabajo se han establecido estos tres tipos de formaciones concretas al orientarse la aplicación del enjambre a labores y trabajos relacionados con la lucha contra el fuego y, en concreto, a posibles tareas de extinción de los mismos.

Las labores de extinción de incendios forestales suelen estar ligadas a atajar y extinguir combustiones que se propagan, habitualmente, de manera incontrolada sobre una superficie vegetal y abierta (217). Es importante destacar que los incendios forestales pueden generarse bajo un conjunto de diversas formas geométricas, entre las que destacan algunas formas regulares como los círculos o las elipses, y cuyas propiedades vienen determinadas por aspectos como las condiciones meteorológicas, la orografía del terreno o la variación en el tipo de combustible (218; 219).

En el caso de los incendios circulares, el frente del incendio se despliega en todas las direcciones desde la zona central hacia el exterior (220), por lo que la formación en cuadrado, con un tamaño y perímetro superior al diámetro del círculo de fuego, tal y como se muestra en la Figura 3.17, permitiría actuar sobre el fuego, evitando su propagación y ayudando al control y reducción del mismo.

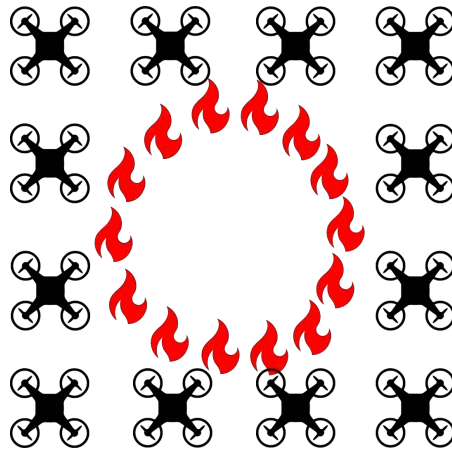


Figura 3.17: Esquema de actuación en formación cuadrada frente a un incendio con forma circular.

En el caso de los incendios con forma elíptica, se caracterizan porque la propagación se ve afectada por un viento con una dirección predominante (220). En este caso, al avanzar el fuego hacia una dirección concreta, disponer de una formación en flecha permite actuar de manera incisiva sobre la zona de crecimiento y avance del fuego. En este caso, el tamaño de la flecha debería establecerse en función del diámetro menor de la elipse para, tal y como se muestra en la Figura 3.18, poder actuar sobre toda el área afectada.

Por último, aunque los incendios no presenten una forma regular, sí que presentan en común distintas partes, entre las que destaca el frente de llama. Este frente es la parte perimetral del incendio que se encuentra afectada por la velocidad del viento y la orografía del terreno, por lo que es la zona con mayor velocidad de propagación y generación de calor (220). Por este motivo, se considera la zona del incendio que ocasiona mayores daños y, su contención es un aspecto crucial para el control del incendio. Una formación en línea permite actuar sobre dicho frente de manera eficaz si se disponen los UAVs de tal forma que la longitud de la línea de formación sea similar a la longitud del frente de llama, tal y como se muestra en la Figura 3.19.

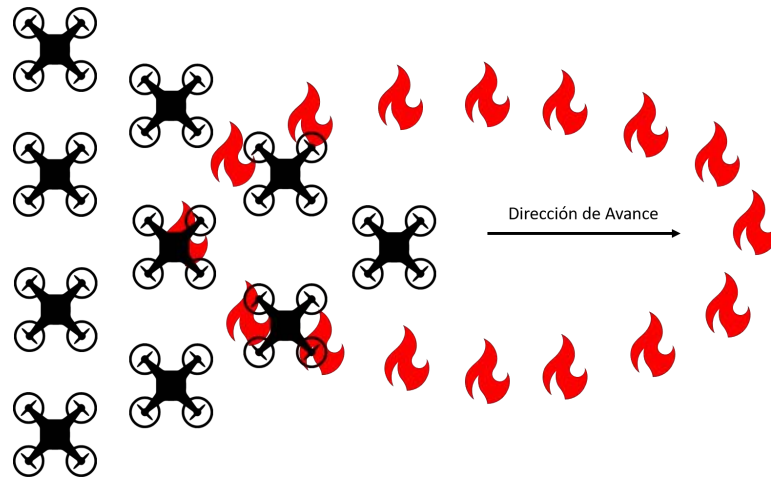


Figura 3.18: Esquema de actuación en formación tipo flecha frente a un incendio con forma de elipse.

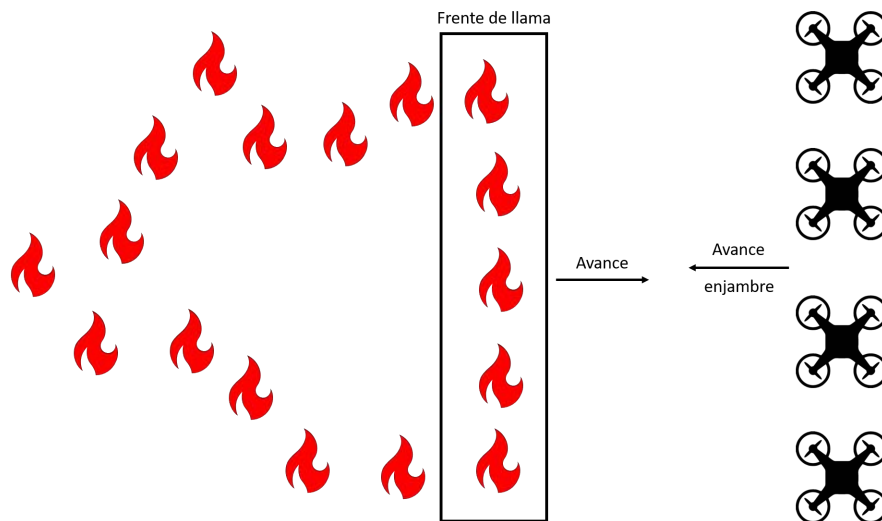


Figura 3.19: Esquema de actuación en formación en línea contra un frente de llama.

3.3. Simulación y Validación del Método de Planificación de Trayectorias

En esta sección se analizan los resultados obtenidos en los métodos de planificación de trayectorias para enjambres de UAVs descritos en las secciones previas tanto para información bidimensional como para información tridimensional del entorno. Este análisis no está basado únicamente en aspectos cuantitativos, sino

que los resultados obtenidos han sido testados en entornos de simulación, sobre los que se ha implementado una arquitectura de enjambre escalable, versátil y configurable, que permita validar los algoritmos implementados.

3.3.1. Arquitectura de software

En el campo de los vehículos autónomos y, especialmente, en el área de los UAVs, las simulaciones constituyen un aspecto crucial para el testado y validación de algoritmos implementados, debido a que cualquier mínimo error en los métodos desarrollados puede conducir a la pérdida o fallos en el control de los UAVs y, con ello, la pérdida del vehículo, incluyendo los sensores o elementos embarcados en el mismo, lo que constituiría un incidente con grandes pérdidas económicas.

Por este motivo, se ha diseñado una arquitectura de software sobre el framework de Robot Operating System (ROS) (221) y el entorno de simulación Gazebo (222) que permita la simulación de diferentes misiones del enjambre buscando un alto grado de similitud a la implementación de los métodos desarrollados en la plataforma real SkyOnyx (223; 224). La arquitectura de software presentada en este apartado es empleada para la prueba y validación del conjunto de algoritmos y métodos recogidos en esta tesis.

Para ello, bajo el marco de ROS, se ha establecido una arquitectura centralizada, en la que un nodo central se encarga del control, intercambio de información y toma de decisiones de cada uno de los UAVs que constituyen el enjambre. El uso de ROS está ligado a su capacidad para implementar y desarrollar software colaborativo y portable, que permita la generación de arquitecturas de comunicación robustas y escalables.

Estas características no constituyen los únicos motivos para la elección y uso de dicho software, sino que también se encuentra ligado a la implementación de dichos métodos en plataformas aéreas reales, las cuales se encuentra equipadas con la controladora de vuelo Pixhawk (225). Esta controladora se basa en software libre con el objetivo de extender este autopiloto entre instituciones académicas, el mercado de los UAVs destinados al ocio y entretenimiento y, proyectos industriales de bajo costo. Dicho autopiloto nace como una controladora de vuelo orientada al uso de UAVs en modo autónomo mediante el uso de sensores embarcados y el intercambio de información a través del protocolo de comunicación Micro Air Vehicle link (MavLink). MavLink constituye un protocolo de comunicación para sistemas no tripulados que especifica un conjunto de mensajes a intercambiar entre dichos sistemas y las GCSs, con útiles funciones para controlar las misiones de dichos sistemas y su integración en Internet (226; 227). El control a través de dicho autopiloto hace de ROS una herramienta de software apropiada como puente de enlace entre los métodos de planificación de trayectorias y las plataformas aéreas,

al disponer de importantes desarrollos, tanto en la adquisición e intercambio de información de sensores, como en los protocolos de comunicación MavLink, como es el caso de MAVROS, una interfaz del protocolo MavLink basado en ROS.

Como muestra la Figura 3.20, se ha implementado un nodo central basado en ROS que permite, en primer lugar, establecer un enlace entre la solución generada por los métodos de planificación de trayectorias en 2D y 3D y los UAVs (tanto simulados como reales) y, en segundo lugar, proporcionar un algoritmo de control de posición y seguimiento de dichas trayectorias de cada uno de los UAVs a través de la interfaz *MAVROS*. Las entradas al nodo central corresponden a una lista de identificadores para cada uno de los UAVs, la posición inicial de cada UAV con respecto a un sistema de referencia común para todo el enjambre y un archivo con el conjunto de puntos de ruta que constituyen cada trayectoria de los UAVs.

Una vez establecida la comunicación con los UAVs y cargados los parámetros de entrada descritos previamente, el nodo central se encarga de cambiar el modo de vuelo de cada uno de los UAVs a modo *Offboard*, el cual se vincula al comportamiento autónomo del UAV, de ejecutar la instrucción de armado de motores y comenzar con el envío de los puntos de ruta a seguir. Mediante el intercambio de información con cada uno de los autopilotos embarcados en los diferentes UAVs, el nodo recibe la posición GPS de cada UAV y realiza un seguimiento de la trayectoria comprobando si dicho UAV ha alcanzado cada punto de ruta o si se ha completado la ruta prefijada alcanzando la ubicación de destino. Tal y como se observa en la Figura 3.20, el nodo generado es escalable y, mediante un archivo de configuración, se puede establecer en conjunto de parámetros necesarios para agregar agentes al enjambre.

En cuanto al entorno de simulación, se ha optado por integrar los métodos propuestos sobre *Gazebo*, el cual constituye un entorno de simulación que permite crear entornos en 3D para múltiples vehículos basándose en el uso de código abierto y, simular sistemas de control con una alta fidelidad. Otro aspecto considerado para establecer *Gazebo* como entorno de simulación es su compatibilidad con el autopiloto Pixhawk y, en concreto, con el Firmware *PX4*, el cual constituye un software de control de vuelo en código abierto para vehículos no tripulados. *PX4* proporciona un estándar para dar soporte de hardware y software para UAVs, a través de una arquitectura modular y extensible basada en puertos (228).

3.3.2. Validación de la planificación de trayectorias en 2D

El método de planificación de trayectorias en 2D, basado en PRM, para un enjambre de UAVs ha sido implementado satisfactoriamente sobre la arquitectura de software propuesta anteriormente, realizando simulaciones para los diferentes casos de uso, comprobando que en todas ellas los UAVs son capaces de completar

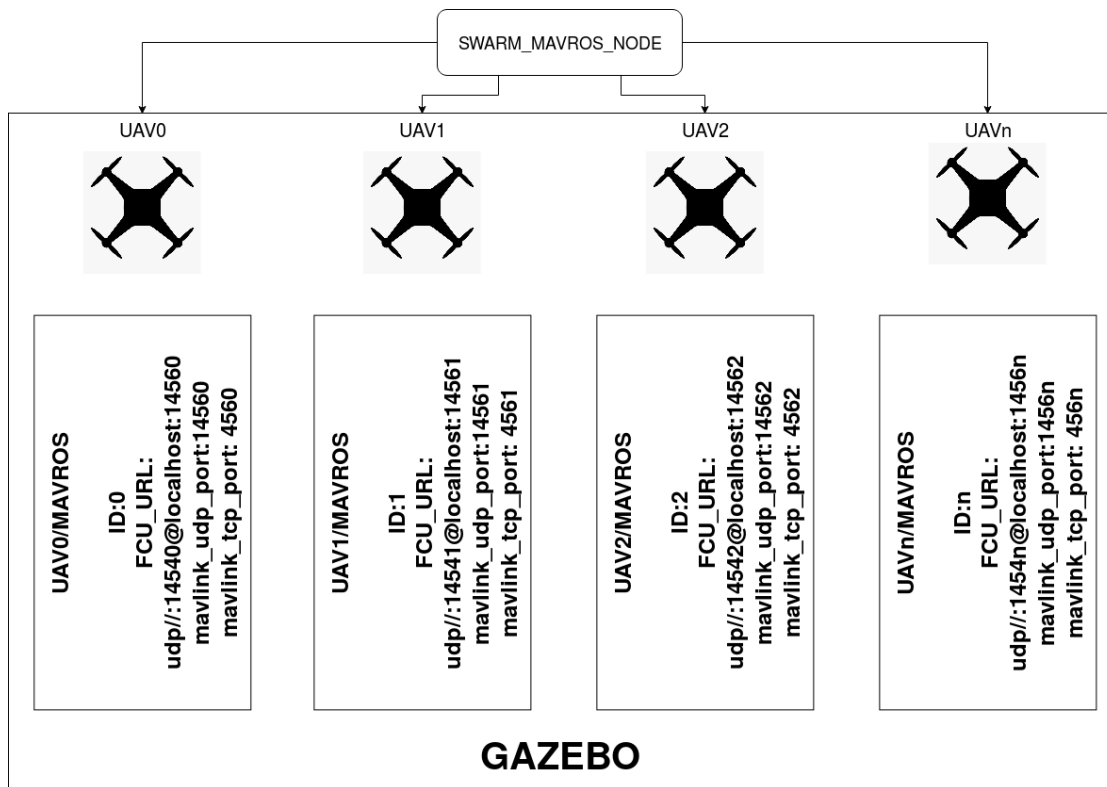


Figura 3.20: Esquema de arquitectura de software implementada.

satisfactoriamente la misión, navegando sin supervisión, sin colisionar con obstáculos presentes en el entorno, ni estáticos ni correspondientes al resto de UAVs del enjambre. Para validar la planificación en 2D sobre entornos urbanos, se ha empelado un entorno estructurado formado por múltiples estancias y cuyas dimensiones se establecen en 78.5 m de largo y 32.5 m de ancho, lo que supone un área a explorar de 2551.25 m^2 , como el descrito en la sección 3.1.

Uno de los aspectos considerados para basar el método de planificación de trayectorias en algoritmos de PRM es conseguir una solución escalable, en la cual el aumento del número de UAVs no implique un aumento crucial en el tiempo de cómputo empleado para la generación de una solución óptima.

La Tabla 3.1 recoge el conjunto de datos que permite analizar como varía el tiempo de cómputo en función de la cantidad de UAVs participantes en la misión, para una exploración constante y completa, entendiéndose por constante que a lo largo de las simulaciones no se varía el número de nodos creados y por completa que el valor de nodos creados asegura la obtención de una solución óptima para todos los casos propuestos indiferentemente del número de UAVs empleados. Se puede observar, tal y como se analiza a continuación, que el comportamiento es

muy diferenciado según la situación a tratar.

Tabla 3.1: Tiempo de cómputo en función del número de UAVs.

Número de UAVs	Caso Etiquetado(s)	Caso Etiquetado / UAVs	Caso No Etiquetado (s)	Caso No Etiquetado / UAVs
1	0.0166	0.0166	0.0172	0.0172
3	0.0187	0.0062	0.0190	0.0063
5	0.0198	0.0040	0.2213	0.0443
10	0.0244	0.0024	0.0377	0.0038
25	0.0476	0.0019	0.2784	0.0111
50	0.1122	0.0022	1.9635	0.0393
75	0.2282	0.0030	7.9334	0.1058
100	0.3746	0.0037	20.5284	0.2053

En el caso no etiquetado, la necesidad de introducir un algoritmo de asignación de objetivos que minimice la distancia total recorrida por el enjambre provoca que, a medida que el número de UAVs aumenta el tiempo empleado en encontrar la solución final óptima crezca siguiendo una distribución polinómica de grado 2, tal y como se muestra en la Figura 3.21. Aunque este crecimiento provoca que, para enjambres compuestos por un alto número de UAVs, el tiempo de cómputo se establezca en el orden de magnitud de segundos, es importante resaltar que el tiempo empleado por UAV se mantiene en un orden de magnitud inferior y, para el caso máximo de 100 UAVs la solución alcanzada es capaz de generar una ruta óptima cada 0,2s.

Junto a este aspecto, resaltar también que el método húngaro trabaja con un número posible de soluciones que aumenta de manera exponencial con el aumento del número de UAVs, por lo que se puede considerar que la respuesta del método sigue siendo adecuada, pues el crecimiento del tiempo de cómputo empleado en el mismo es menor en comparación con el aumento de las posibles soluciones con las que trabaja el algoritmo.

En el caso etiquetado, se observa como los valores referidos al tiempo de cómputo se mantienen dentro en un orden de magnitud de décimas de segundo incluso para enjambres de grandes dimensiones, como sería el caso de un enjambre de más 50 UAVs o superior. Si se analiza el crecimiento en el tiempo de cómputo total para el conjunto de UAVs, se observa que aunque, dicho crecimiento sigue una distribución cuadrática, tal y como se recoge en la Figura 3.22, si se traslada el análisis a el tiempo empleado en generar la solución óptima por cada uno de los UAVs, se concluye que el tiempo empleado se mantiene dentro de un rango acotado de tiempo incluso para enjambres de grandes dimensiones, es decir, se está ante una solución escalable, en la que el tiempo de cómputo total empleado en generar la solución se va a amortizando a medida que el número de UAVs aumenta.

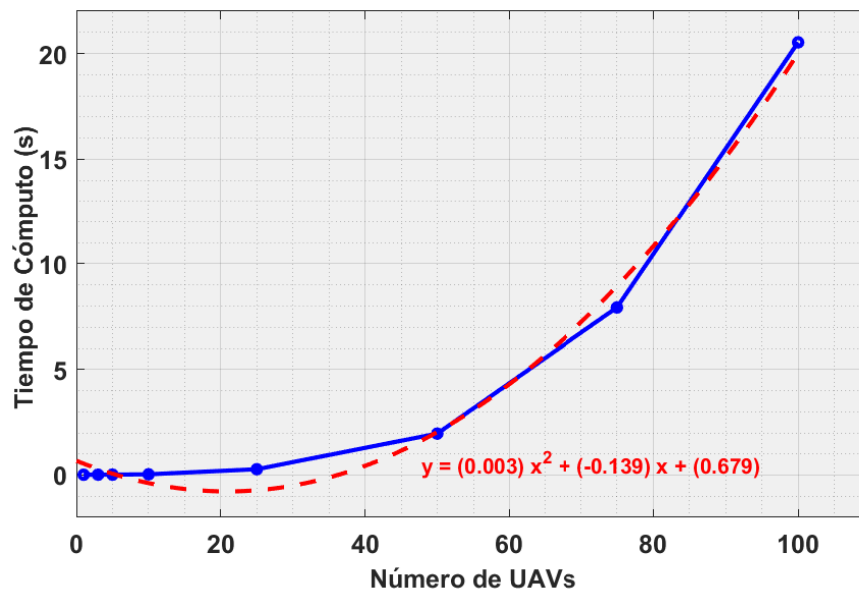


Figura 3.21: Tiempo de cómputo frente al número de UAVs para el caso no etiquetado.

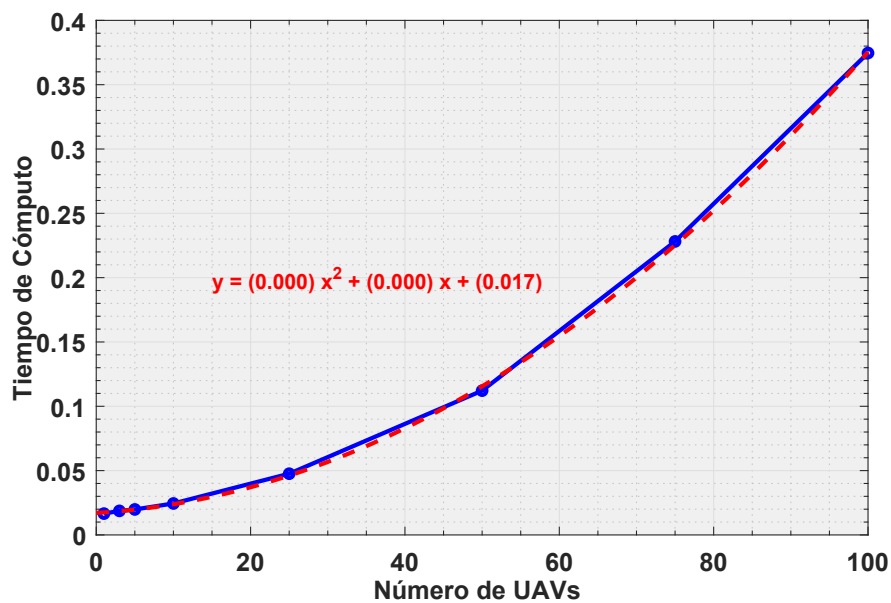


Figura 3.22: Tiempo de cómputo frente al número de UAVs para el caso etiquetado.

Si se toma el tiempo de cómputo derivado de la generación de una solución óptima como único parámetro de comparación entre ambas situaciones, la conclusión

sería clara debido al crecimiento del mismo a medida que el número de UAVs que forman el enjambre aumenta, pero antes establecer conclusiones sobre el caso no etiquetado, y su comparación con el caso etiquetado, es conveniente analizar cuál es el efecto de este método de asignación y, en qué porcentaje se reduce la distancia total recorrida por el enjambre cuando se establece esta situación en comparación con la asignación secuencial objetivos-UAVs empleada en el caso etiquetado. La Figura 3.23 muestra cómo, para la misma situación, el caso no etiquetado es capaz de reducir de manera considerable la distancia total recorrida por el enjambre a medida que aumenta el número de UAVs participantes en la misión.

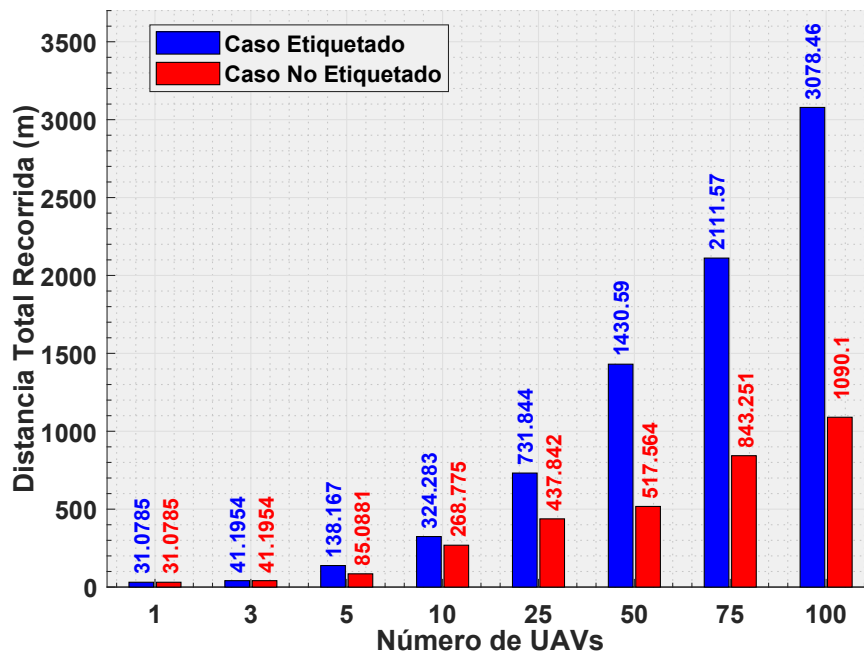


Figura 3.23: Distancia total recorrida para el caso etiquetado y no etiquetado en 2D en función del número de UAVs.

Aunque para enjambres compuestos por un número de agentes bajo, como por ejemplo el caso de 3 UAVs la distancia total recorrida no presenta ninguna reducción, a partir de 5 UAVs la distancia total recorrida por el enjambre se reduce una media del 47%. Si se analiza la implementación del método húngaro para enjambres de grandes dimensiones, como en el sistema formado por 50 o más UAVs, la reducción de la distancia recorrida se incrementa hasta alcanzar valores por encima del 60%. Si, además, se considera que el movimiento de los UAVs se realiza mediante un comportamiento dinámico estable en el que la velocidad de crucero es constante durante la navegación entre puntos de ruta, esta reducción de la distancia total recorrida se transforma en una reducción considerable del tiempo

3.3. Simulación y Validación del Método de Planificación de Trayectorias 11

de vuelo empleado por todo el sistema para completar una misión, es decir, en una respuesta más rápida y eficaz en operaciones críticas como pueden ser la lucha contra los incendios.

Uniendo los dos análisis realizados hasta ahora, aunque el método de planificación de trayectorias para el caso no etiquetado podría perder cierta escalabilidad en cuanto al orden de magnitud en el que se obtiene una respuesta óptima, la realidad es que para un caso extremo de 100 UAVs el método propuesto sería capaz de generar cada 0,2 s una ruta óptima en la que la distancia recorrida por el enjambre y, con ello los tiempos de actuación, se reduzcan hasta un 60 %.

Hasta ahora se ha analizado el método de planificación de trayectorias en su conjunto, pero como se ha descrito en las Secciones 3.1 y 3.2, este método está conformado por dos fases bien diferenciadas. Por un lado, la exploración del entorno y, por otro, la generación de una solución óptima a partir del grafo generado en la exploración. Como se ha indicado con anterioridad, es importante establecer un equilibrio entre el número de nodos generados durante la exploración y el tiempo de cómputo empleado en este proceso para generar el grafo resultante, debido a que los algoritmos basados en PRM no son completos y, por tanto, pueden no encontrar una solución, aunque esta exista debido a una exploración incompleta. Por esta razón, la Tabla 3.2 recoge la variación del tiempo de cómputo empleado, únicamente, en generar el grafo en función del número de nodos empleados para la exploración junto al dato de si con esa cantidad de nodos se ha conseguido obtener una solución completa o, si por el contrario, la exploración es insuficiente y quedan objetivos sin alcanzar. De cara a poder valorar este último aspecto, se ha trabajado con un enjambre fijo de 10 UAVs y, para comprobar si existe solución posible se ha implementado el caso no etiquetado, de tal manera que, se consideren todas las posibles combinaciones entre UAVs y objetivos, aumentando así la probabilidad de encontrar una solución completa es mayor.

Los resultados muestran cómo, para este entorno, no es necesaria una densidad de nodos alta de cara a conseguir que se encuentre una solución completa. También, a través de dicha información se pone de manifiesto como a medida que se aumenta el número de nodos el tiempo empleado en la construcción del grafo aumenta, por lo que es necesario establecer una densidad de nodos equilibrada.

Tabla 3.2: Comparativa tiempo de cómputo para generación de grafo en función del número de nodos.

nNodos	10	25	50	75	100	125	150	175	200
Tiempo de Cómputo (s)	0.005477	0.007522	0.013962	0.02122	0.032744	0.04318	0.058322	0.08157	0.095268
Solución Completa	No	Si	Sí	Sí	Sí	Sí	Sí	Sí	Sí

Por último, en cuanto a los resultados cualitativos derivados de la implementación y simulación del método sobre la arquitectura de software propuesta, se concluye que el método propuesto permite navegar, de manera segura y sin supervisión, a un enjambre escalable de UAVs, atravesando un entorno estructurado formado por distintas estancias.

3.3.3. Validación de la planificación de trayectorias en 3D

El método de planificación de trayectorias en 3D propuesto, basado en el algoritmo de PRM para enjambres de UAVs, ha sido implementado de manera satisfactoria sobre la arquitectura de software descrita en la Sección 3.3.1, realizando simulaciones para diferentes casos de uso y sobre distintos escenarios.

Para el análisis cuantitativo del método de planificación en 3D se ha optado por emplear un entorno de trabajo de grandes dimensiones, que permita comprobar la aplicabilidad del método a casos de uso como pueden ser los incendios en áreas urbanas. En concreto, realizar un trabajo de implementación, prueba y validación sobre un edificio de nueve plantas con múltiples estancias en cada una de ellas, como el mostrado en la Figura 3.24 y, cuyas dimensiones son 41 *m* de largo, 24 *m* de ancho y 49 *m* de alto, es decir, un área de exploración de 48216 *m*³.

El primero de los aspectos a analizar es si se mantiene la escalabilidad del método propuesto considerando el tiempo de cómputo empleado en la obtención de una solución óptima en función del número de UAVs que participan en el enjambre. Los resultados obtenidos son recogidos en la Tabla 3.3, y en ellos se puede observar como a medida que el número de UAVs aumenta la respuesta de ambos casos de estudio es diferente al igual que ocurría para el caso en 2D.

En el caso etiquetado, se puede observar como a medida que aumenta el número de UAVs participantes en el enjambre el tiempo computacional va aumentando progresivamente, pero si se centra el estudio en el tiempo invertido en generar una ruta óptima por cada UAV se puede apreciar como este tiempo por agente se mantiene dentro de unos rangos determinados e incluso, va descendiendo a medida que aumenta el número de UAVs. Esta respuesta del método se deriva de la amortización del tiempo empleado en generar el grafo en 3D, demostrando la escalabilidad del método propuesto.

En el caso no etiquetado, la introducción del método húngaro provoca un aumento cuadrático del tiempo de cómputo a medida que crece el número de UAVs en el enjambre. Sin embargo, este resultado puede ser considerado satisfactorio debido a que este crecimiento es más suavizado si se compara con el aumento del número de posibles soluciones con las que trabaja el método húngaro, el cual crece exponencialmente con cada UAV nuevo añadido al enjambre. Además, si como en el caso etiquetado, se analiza el tiempo de cómputo de manera individualizada, es

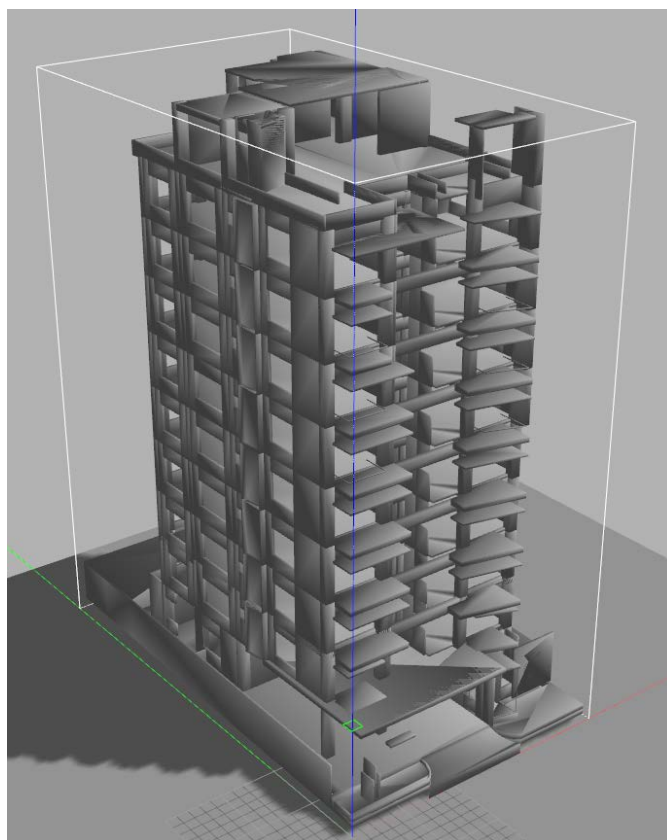


Figura 3.24: Entorno de simulación para validación de método de planificación de trayectorias en 3D.

decir, cuanto tiempo se emplea para generar cada una de las soluciones óptimas, el crecimiento se estabiliza y, se mantiene constante a medida que el número de agentes en el enjambre aumenta.

Para poder comparar por completo ambos casos y establecer consideraciones acerca del aumento del tiempo de cómputo derivado de la implementación del método húngaro, es necesario analizar, en el caso de las trayectorias en 3D, como se reduce la distancia total recorrida por el enjambre en caso de realizarse una asignación correlativa de UAVs y objetivos o, de asignarse una relación UAV-objetivo que minimice la distancia total recorrida. La Figura 3.25 muestra como a medida que se aumenta el número de UAVs, el efecto causado por el método húngaro se hace más visible, logrando, incluso para enjambres de pequeñas dimensiones, una reducción de la distancia total recorrida cercana al 50 %.

A diferencia del caso de planificación en 2D, en este caso, el método húngaro tiene un efecto importante incluso para enjambres formados por 2 miembros. Esto se debe a que, en primer lugar, se considera una tercera dimensión en el problema y,

Tabla 3.3: Tiempo de cómputo en función del número de UAVs para 3D.

Número de UAVs	Caso Etiquetado (s)	Tiempo Etiquetado / UAVs	Caso No Etiquetado (s)	Tiempo No Etiquetado / UAVs
1	0.7577	0.7577	0.7505	0.7505
2	0.7262	0.3631	0.7505	0.3753
3	0.7326	0.2442	0.7754	0.2585
4	0.7467	0.1867	0.7972	0.1993
5	0.7787	0.1557	0.9400	0.1880
6	0.7874	0.1312	1.0150	0.1692
7	0.8108	0.1158	1.1335	0.1619
8	0.8158	0.1020	1.2802	0.1600
9	0.8416	0.0935	1.5747	0.1750
10	0.8898	0.0890	1.8076	0.1808
11	0.8829	0.0803	1.9092	0.1736
12	0.8927	0.0744	2.0933	0.1744
13	0.9164	0.0705	2.3556	0.1812
14	0.9138	0.0653	2.7646	0.1975
15	0.9110	0.0607	2.9362	0.1957

en segundo lugar, a que se trabaja con un entorno simulado de grandes dimensiones en el que las distancias en cada una de las rutas son mayores y, por tanto, la necesidad de optimizar este aspecto se hace más necesaria.

En entornos de altas dimensiones, la exploración del entorno se convierte en un aspecto clave para poder, en primer lugar, encontrar una solución completa en la cual se alcancen todos los objetivos y, en segundo lugar, encontrar un equilibrio entre el número de nodos y el tiempo de cómputo que permita maximizar la optimización de la distancia recorrida haciendo uso del algoritmo A*, tanto en el caso etiquetado como en el no etiquetado. La Figura 3.26 recoge como varía el tiempo de cómputo total empleado en la generación del grafo 3D en función del número de nodos empleados en la exploración y se observa como de necesario es establecer un equilibrio que permita obtener soluciones completas controlando el tiempo de cómputo derivado de dicho proceso, puesto que el gasto computacional crece cuadráticamente a medida que se aumenta el número de nodos. Dado que la generación de nodos es aleatoria y solo se incluyen en el grafo aquellos nodos que se encuentran en el espacio libre, se ha optado por realizar, para un mismo número de nodos, 10 simulaciones y calcular el valor medio del tiempo empleado en cada una de ellas, siendo dicho tiempo medio el que aparece en la Figura 3.26.

Para poder encontrar ese equilibrio, es necesario estudiar para que valor de

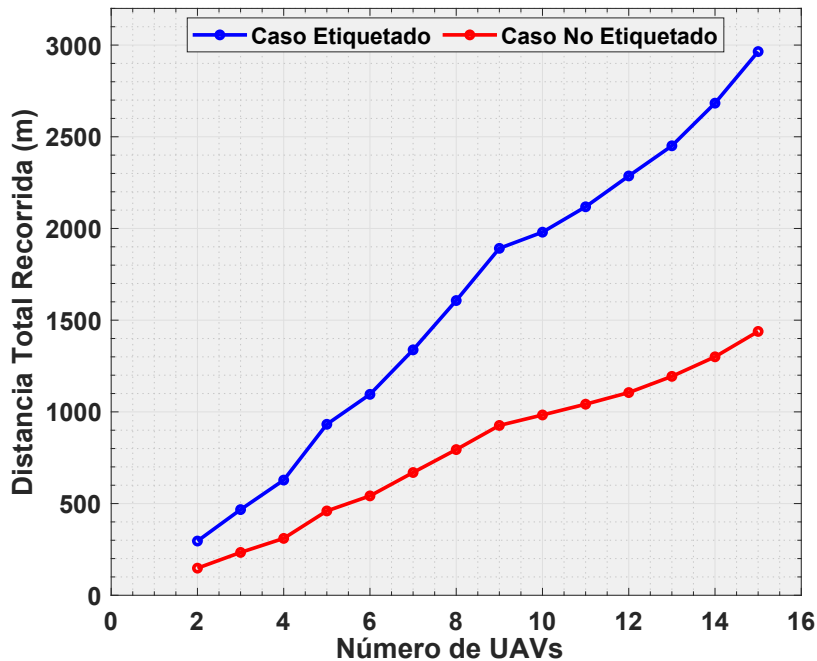


Figura 3.25: Distancia total recorrida por el enjambre para planificación en 3D en función del número de UAVs.

Los nodos creados se obtiene una solución completa para ambos casos de estudio. La Figura 3.27 ayuda a establecer ese valor de equilibrio, que permita establecer un valor de densidad (nodos/m^3) que aplicar a futuros entornos. Para poder realizar este estudio, se ha fijado un tamaño de enjambre de 10 UAVs y se han realizado 10 simulaciones para el mismo número de nodos comprobando en cada una de ellas si existe una solución completa o no. De estas simulaciones se obtiene que, para el entorno empleado, en el caso etiquetado son necesarios 2300 nodos para poder encontrar una solución completa, es decir, que en el 100 % de las veces se han alcanzado todas las ubicaciones establecidas. Mientras que, para el caso no etiquetado, son necesarios 2000 nodos para poder alcanzar una solución posible el 100 % de las veces. Considerando el volumen del entorno y el número de nodos mínimo para el cual, en ambos casos, se obtiene siempre una solución completa, se puede establecer que es necesaria una densidad de $0.04 \text{ nodos}/\text{m}^3$.

Unificando el conjunto de datos analizados hasta este momento, se concluye que, para una aplicación real de emergencias en entorno urbano, con un edificio de múltiples plantas se podría encontrar un conjunto de trayectorias óptimas para un enjambre de 10 UAVs en un intervalo de tiempo de 0.7 s a 1.2s, según se trabaje en un caso etiquetado o no etiquetado.

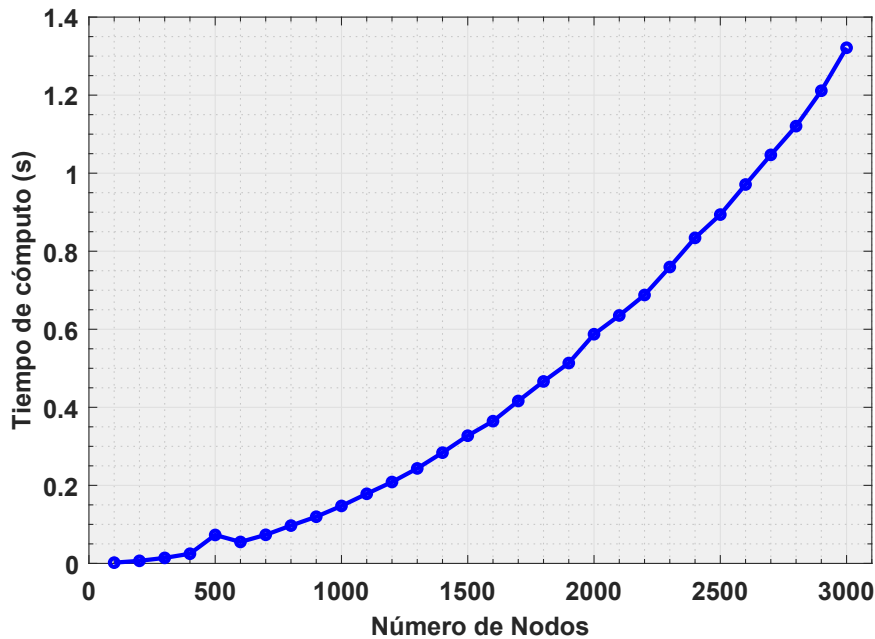


Figura 3.26: Tiempo de cómputo de generación de grafo 3D en función del número de nodos.

En entornos de grandes dimensiones se podría pensar que una exploración densa ayudaría a reducir la distancia total recorrida por el entorno al poder generar un grafo más completo y, con ello, un mayor abanico de bordes a combinar de cara a encontrar una trayectoria en la que se minimice la distancia recorrida por cada UAV. Ante esta posibilidad, se ha analizado como afecta el número de nodos empleados en la exploración a la distancia total recorrida por el enjambre siendo el resultado de dicho estudio el recogido en la Figura 3.28. En ella, se observa que esta suposición no es correcta y que, a pesar de aumentar el número de nodos y, con ello, obtener un grafo más denso, la distancia total recorrida por el enjambre se mantiene constante tanto para el caso etiquetado como para el caso no etiquetado. Por tanto, es un aspecto crucial encontrar el equilibrio entre nodos y probabilidad de éxito, pues realizar exploraciones más densas no conllevan ninguna mejora en el método de planificación.

Por último, los métodos propuestos incluyen un pre-procesado del entorno previo a la exploración del mismo en el que a través de un parámetro se genera un margen de seguridad alrededor de los obstáculos, al igual que sucede con el método de planificación de trayectorias en 2D. Este margen de seguridad está relacionado con el tamaño máximo de los UAVs empleadas en el enjambre y, puede ayudar a determinar que tareas son accesibles o no en función del tamaño del UAV. El aumento de dicho margen de seguridad conlleva a que ciertas tareas que en un

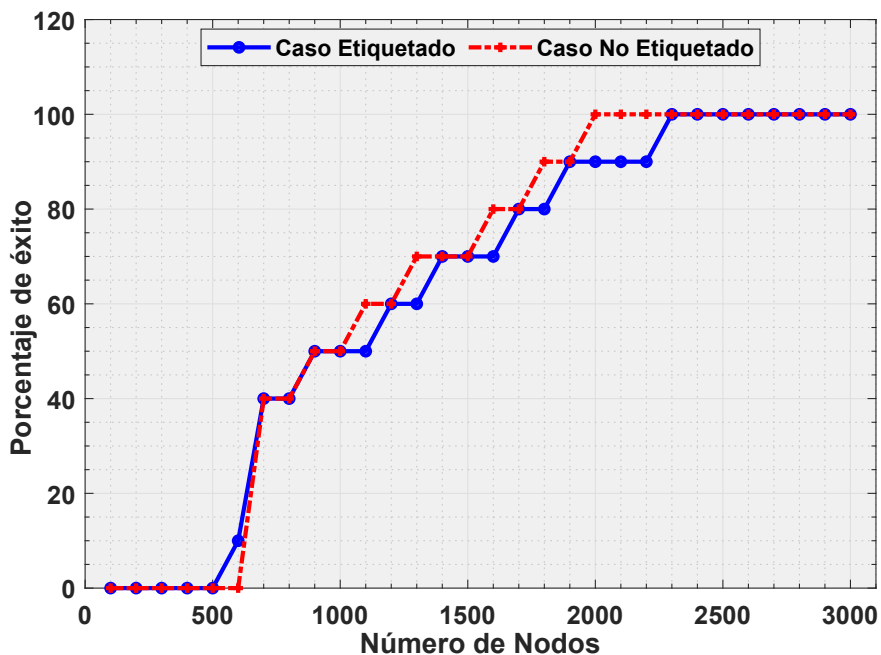


Figura 3.27: Porcentaje de éxito del planificador de trayectorias en 3D en función del número de nodos.

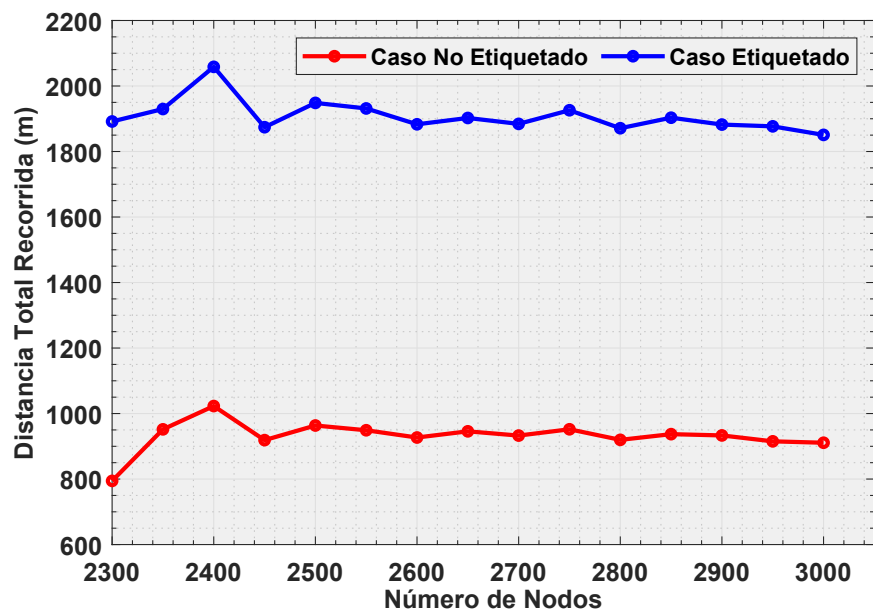


Figura 3.28: Distancia total recorrida para trayectorias 3D en función del número de nodos.

principio son alcanzables, se conviertan, para la misma situación, en lugares inaccesibles para el nuevo margen de seguridad. Por tanto, la modificación de este parámetro permite establecer cuál es el tamaño máximo de UAV para acceder a dicha ubicación. La Tabla 3.4 presenta un conjunto de 10 tareas y si son accesibles o no en función del tamaño del UAV. Los resultados mostrados en dicha tabla indican *si* en el caso de que al menos un UAV sea capaz de alcanzar dicha tarea y, un *no* si ninguno de ellos es capaz de alcanzar dicho objetivo. De esta forma se introduce un parámetro de robustez al método propuesto y, permitiría, en el desarrollo real de una misión, comprobar previamente si algún objetivo es alcanzable o no en función del tamaño de los UAVs de los que se dispone.

Tabla 3.4: Alcanzabilidad de las tareas considerando el tamaño de los UAVs.

Tamaño de UAV (m)	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
0.5	S	S	S	S	S	S	S	S	S	S
1.0	S	S	S	S	S	S	S	S	S	S
1.5	S	S	S	S	S	S	S	S	S	S
2.0	S	S	S	N	S	S	S	S	S	S
2.5	S	S	S	N	S	S	S	N	S	S
3.0	S	S	N	N	S	S	S	N	S	S
3.5	N	S	N	N	S	N	N	N	S	N
4.0	N	N	N	N	S	N	N	N	N	N
4.5	N	N	N	N	Y	N	N	N	N	N
5.0	N	N	N	N	N	N	N	N	N	N

T= Tarea, S=Si, N=No.

Sumado al análisis cuantitativo del método de planificación de trayectorias en 3D, se encuentra el análisis cualitativo derivado de la prueba y validación de los algoritmos en simulaciones. Así pues, se han realizado simulaciones en el entorno de simulación de *Gazebo*, para diferentes situaciones y configuraciones y, obteniendo en todas ellas un resultado satisfactorio, en el cual todos los UAVs son capaces de navegar, sin supervisión y de manera segura, a sus objetivos y regresar al punto de inicio sin colisionar entre ellos ni con el entorno.

3.4. Suavizado de Trayectorias

Como se recoge en el estado del arte de estas metodologías, los algoritmos de muestreo basados en PRM presentan el inconveniente de generar trayectorias con un patrón de Zig-Zag, es decir, la consecución de bordes para establecer una ruta completa da como resultado una trayectoria quebrada que va en contra de un

comportamiento dinámico atenuado y constante y, que genera cambios bruscos en las dinámicas de los vehículos. Este resultado que, en primer lugar, está asociado a un comportamiento dinámico no adecuado para realizar tareas como la captura de información a través de sistemas ópticos, o para llevar a cabo maniobras de gran precisión y que, en segundo lugar, puede provocar un mantenimiento prematuro de ciertos componentes de los UAVs, se soluciona mediante un post-procesado que permita suavizar las trayectorias generadas.

De entre las soluciones recogidas en la literatura para corregir el problema de las trayectorias en Zig-Zag, se encuentran los métodos basados en Línea de Vista o Line-of-Sight (LOS), que tratan de reducir el número de nodos o puntos de ruta que forman la trayectoria preestablecida. Este post-procesado permite simplificar la ruta propuesta de manera simple y computacionalmente ligera.

Estos algoritmos consisten en que para cada uno de los nodos introducidos en la trayectoria generada y, partiendo de la posición inicial del UAV, se comprueba si existe una conexión directa entre dicha posición inicial y el objetivo que se desea alcanzar. De esta forma, se traza un segmento entre ambos puntos y se verifica si dicho segmento está libre de colisiones. Si esta situación no se produce, se realiza un proceso iterativo en el que se comprueba el segmento de unión entre el punto inicial y cada uno de los nodos de la ruta prefijada, pero en retroceso, es decir, desde el punto de ruta de destino hacia el nodo inicial. En caso de encontrar un segmento libre de obstáculos, se establece este borde como bueno y se eliminan todos aquellos bordes y nodos intermedios que formaban parte de la trayectoria preestablecida. A continuación, se repite todo el proceso, pero en lugar de buscar un segmento que una el punto inicial con el final se toma como punto de partida el nodo intermedio ya conectado a la posición inicial y, se repite el proceso de búsqueda de segmento libre de obstáculos partiendo del a ubicación de destino y retrocediendo por los puntos de ruta establecidos hasta encontrar el segmento libre de colisiones.

El algoritmo desarrollado se ha aplicado tanto a las soluciones generadas por la planificación en 2D, como a las soluciones generadas en un entorno en 3D. Es importante resaltar que, en el caso del suavizado de trayectorias sobre 3D, el tiempo de cómputo empleado por el algoritmo es superior ante la necesidad de trabajar con grandes cantidades de información como son los mapas de ocupación en 3D. El proceso de suavizado de trayectorias requiere un proceso iterativo, en el para cada ruta, se debe comprobar si existe segmentos libres de colisiones entre nodos no consecutivos para poder eliminar bordes y nodos intermedios, implementando un procedimiento similar al empleado para generar los bordes del grafo 3D.

3.4.1. Suavizado de trayectorias para planificación en 2D

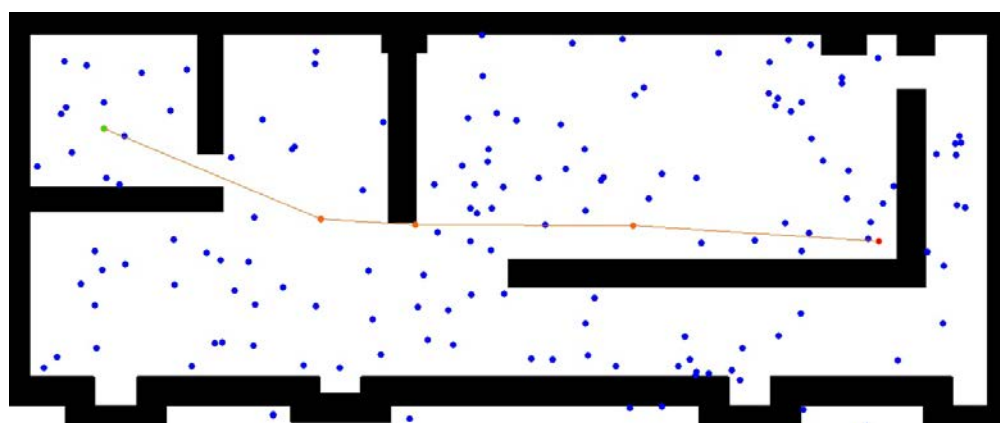
Si se analizan los resultados para una exploración y planificación en 2D, se puede observar cómo empleando un tiempo de cómputo ligeramente superior, se consigue reducir el número de puntos de ruta introducidos en las trayectorias y, con ello, reducir en una pequeña proporción la distancia total recorrida por el enjambre.

En la Figura 3.29 se muestra el resultado cualitativo de dicho algoritmo para una única trayectoria, de cara a poder observar con mayor claridad como el algoritmo elimina aquellos nodos intermedios innecesarios, al ser posible conectar un nodo con otro más alejado sin pasar por dicho punto intermedio. La Figura 3.29a recoge sobre un plano 2D cuál es la trayectoria original, mientras que la Figura 3.29b muestra el proceso de eliminación de nodos intermedios y, con ello, una trayectoria que reduce el efecto Zig-Zag comentado anteriormente.

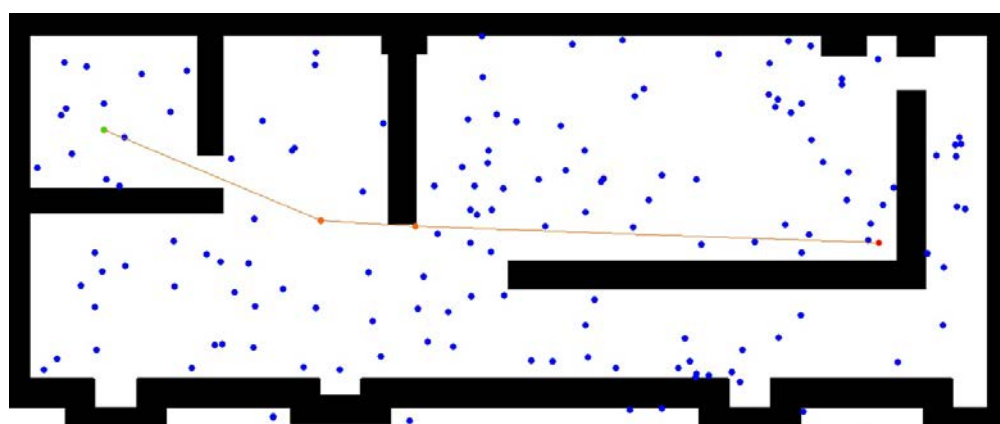
Analizando el algoritmo desde un punto de vista cuantitativo, el número de puntos de ruta generados en el conjunto de las trayectorias, el tiempo computacional derivado de la implementación de dicho método o la distancia total recorrida por el enjambre en función del tipo de trayectoria empleada son aspectos a tratar. Para ello, se ha decidido comparar la solución generada por el planificador de trayectorias en 2D para ambas situaciones, considerando únicamente el caso etiquetado, es decir, en el que la asignación objetivo-UAV es correlativa al orden de registro de las aeronaves y las ubicaciones finales. Los experimentos se han realizado considerando un enjambre de 5 UAVs y, puesto que la exploración del entorno es aleatoria, se ha establecido 5 intentos diferentes sobre un mismo escenario para poder analizar los datos de manera más precisa.

El primero de los aspectos a considerar son el número de puntos de ruta generados en cada una de las trayectorias y, cómo con el uso del algoritmo de suavizado se consigue reducir, en la mayoría de las ocasiones, el número de puntos de ruta intervinientes en cada una de las trayectorias de los UAVs, tal y como se muestra en la Tabla 3.5

Aunque de manera individual la reducción de puntos de ruta puede parecer pequeña, si se analiza su impacto sobre el conjunto de todas las trayectorias generadas para el enjambre se observa que el global es mayor, tal y como se observa en la Figura 3.30. De los datos recogidos en dicha figura se extrae que, para algunos casos la reducción del puntos de ruta es superior al 20 % y que, la media se sitúa muy cerca del mismo, con un valor de 19,41 %, lo cual para un escenario reducido como el de la Figura 3.29 no es un número de puntos de ruta significativo, pero que para entornos de grandes dimensiones, una reducción del 20 % en los nodos intermedios a alcanzar puede suponer una mejora significativa tanto en el comportamiento dinámico como en el tiempo total empleado por el UAV en cubrir dicha trayectoria. Si, además, se considera que en la ruta de cada UAV uno de los



(a) Planificación de trayectoria para caso 2D sin suavizar.



(b) Mapa 2D con trayectoria suavizada.

Figura 3.29: Ejemplo de suavizado de trayectorias en 2D.

puntos de ruta corresponde a la ubicación de destino, la cual va a estar presente siempre, y se analiza la reducción de puntos de ruta considerando solo los nodos intermedios, el porcentaje de reducción se eleva a 23,85% de todos los posibles nodos a eliminar, ya que dicho punto de ruta correspondiente al destino forma parte de todas las trayectorias.

Otro de los aspectos a analizar es si esta reducción de puntos de ruta se traduce en una reducción de la distancia total recorrida por el enjambre y que, suponiendo un comportamiento dinámico estable en el que los UAVs navegan a una velocidad prácticamente constante, implica una reducción en el tiempo de respuesta. En este caso, para la planificación en 2D propuesta, la reducción es prácticamente insignificante, tal y como se observa en la Figura 3.31 para la totalidad de los experimentos.

Esto es debido a varios motivos: el primero de ellos, es que la distancia total

Tabla 3.5: Comparativa de puntos de ruta introducidos en las trayectorias originales y suavizadas en 2D para cada UAV del enjambre.

Trayectorias	Test 1		Test 2		Test 3		Test 4		Test 5	
	O	S	O	S	O	S	O	S	O	S
UAV_1	5	4	5	3	6	4	6	4	5	4
UAV_2	6	4	6	5	6	5	5	4	6	5
UAV_3	6	4	7	5	6	5	5	5	5	5
UAV_4	5	4	6	5	6	5	5	4	6	5
UAV_5	4	4	4	3	5	4	4	4	4	4

O=Trayectoria Original, S=Trayectoria Suavizada.

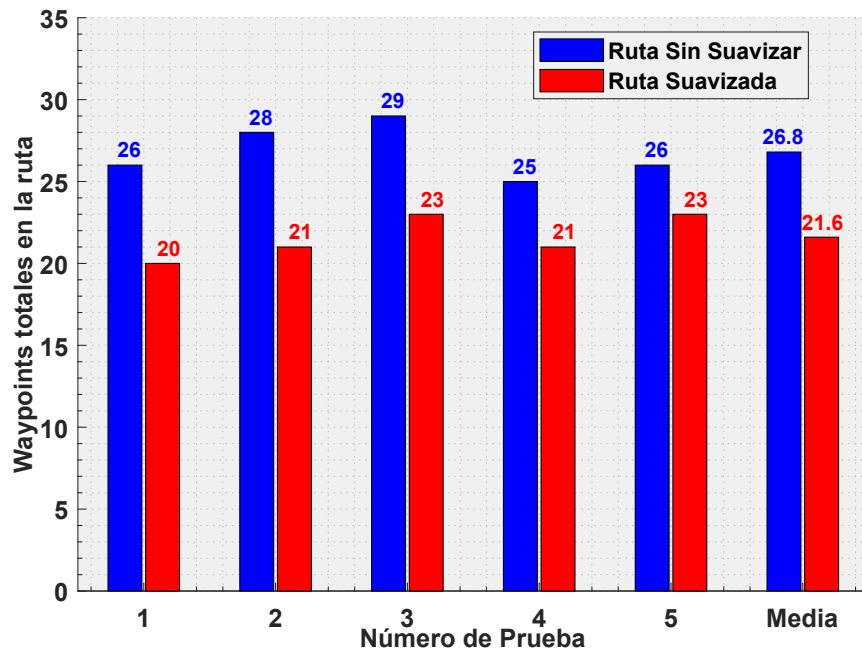


Figura 3.30: Comparativa del total de waypoints para trayectorias sin suavizar y trayectorias suavizadas en 2D para un enjambre de 5 UAVs.

recorrida expresada en la Figura 3.31 considera únicamente la distancias y movimiento en el plano XY y, por tanto, no considera aspectos como la distancia recorrida durante la fase de despegue, ni posibles variaciones de altura, constituyendo este aspecto como la clave de esta falta de reducción de distancia; y, en segundo lugar, la densidad del grafo construido y los parámetros de creación del mismo se encuentran bien ajustados dado que el número de puntos de ruta introducidos originalmente no es demasiado alto, por lo que las rutas iniciales generadas

se acercan o son la solución óptima en términos de distancia mínima recorrida y comportamiento dinámico.

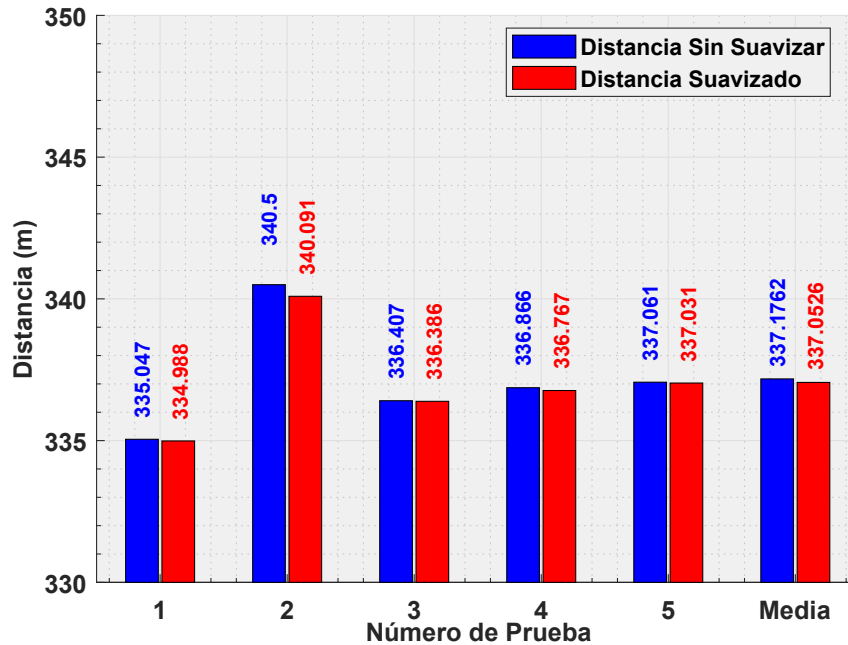


Figura 3.31: Comparativa de la distancia total recorrida por un enjambre de 5 UAVs para trayectorias sin suavizar y trayectorias suavizadas en 2D.

Por último, el otro parámetro analizado con respecto a la implementación de este algoritmo es el tiempo de cómputo adicional que genera su uso para obtener una solución óptima, no solo en distancia recorrida sino en comportamiento dinámico del sistema.

Para los casos analizados, se observa en la Figura 3.32, que el uso de este algoritmo conlleva un aumento del tiempo de cómputo de un 6,62% de media, lo cual lleva a concluir que, dada la reducción del porcentaje de puntos de ruta totales y, con ello la mejora dinámica del movimiento del enjambre y la posible reducción del tiempo de navegación hacia el objetivo, este algoritmo podría ser empleado en una etapa de post-procesado para obtener una solución más óptima y completa a costa de un ligero aumento en el tiempo de cómputo empleado en obtener dicha solución.

3.4.2. Suavizado de trayectorias para planificación en 3D

En cuanto a la aplicación del algoritmo de suavizado a trayectorias en 3D, al incluir la coordenada Z y trabajar sobre todo el espacio posible, el resultado

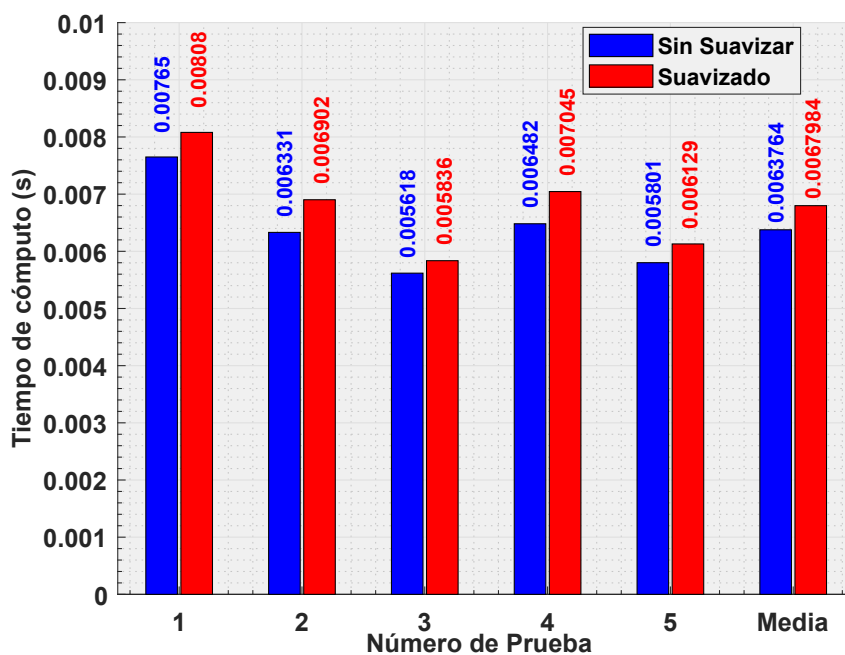
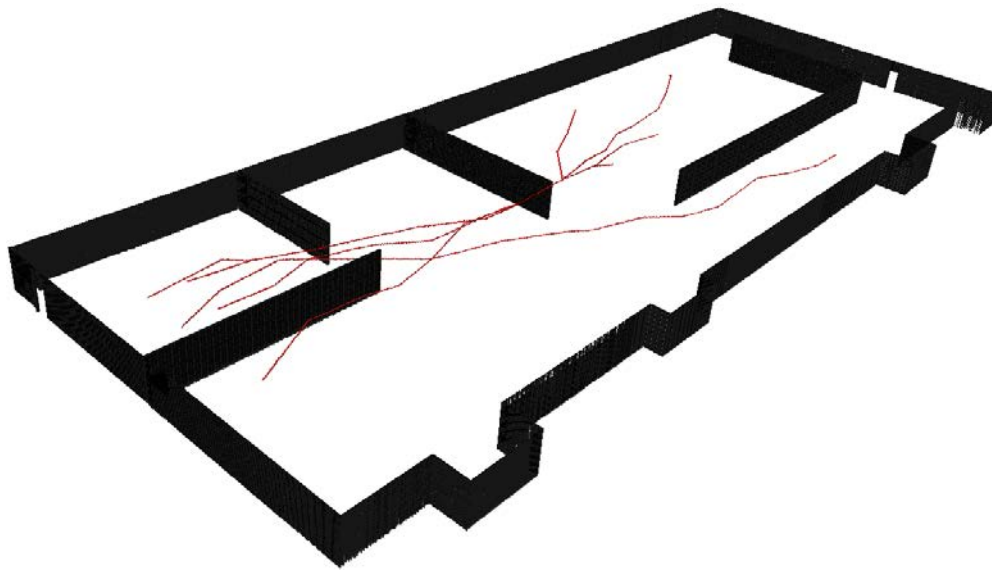


Figura 3.32: Comparativa del tiempo de cómputo empleado para la generación de trayectorias sin suavizar y suavizadas en 2D para un enjambre de 5 UAVs.

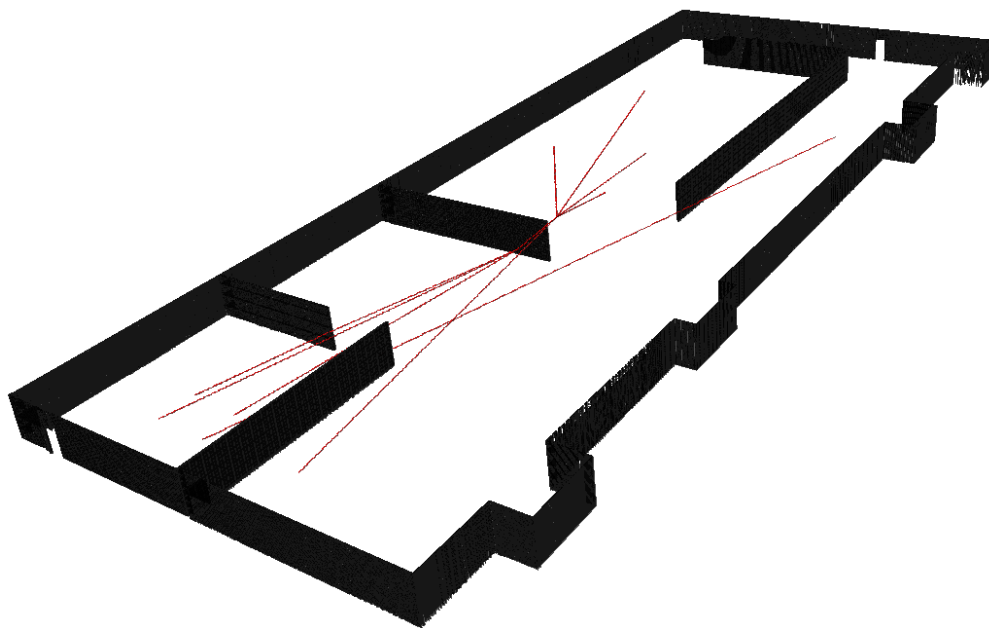
cuantitativo del algoritmo es más evidente, siendo el suavizado de trayectorias más ostensible en las trayectorias en 3D que en las trayectorias en 2D, tal y como se puede observar al comparar la Figura 3.33a, la cual contiene la solución original generada para un enjambre de 5 UAVs, considerando únicamente la distancia recorrida como parámetro a optimizar, con la Figura 3.33b, en la que se muestra la solución suavizada de dichas trayectorias, añadiendo a la optimización de distancia un mejor comportamiento dinámico del sistema. La comparativa muestra como el cambio en las trayectorias es bastante significativo, eliminando el patrón en Zig-Zag y, generando trayectorias en las que predominan largos trayectos en línea recta, mejorando con ello, también, los tiempos de respuesta de actuación del enjambre.

Al igual que en la planificación 2D, se recoge un análisis cuantitativo del uso de este algoritmo como suavizado de trayectorias con patrón Zig-Zag analizando su impacto en aspectos como la cantidad de puntos de ruta incluidos en la solución final, la distancia total recorrida por el enjambre o el tiempo de cómputo requerido para generar trayectorias que mejoren el comportamiento dinámico de los diferentes UAVs.

En cuanto al número de puntos de ruta que forman parte de cada una de las trayectorias, en el caso en 3D se puede observar, tal y como se recoge en la



(a) Planificación de trayectoria para caso 3D sin suavizar.



(b) Mapa 3D con trayectoria suavizada.

Figura 3.33: Ejemplo de suavizado de trayectorias en 3D.

Tabla 3.6, que para cada uno de los UAVs participantes la reducción llevada a cabo es considerable, lo cual pone de manifiesto la necesidad de algoritmos de post-procesado para la eliminación de nodos intermedios.

Tabla 3.6: Comparativa de puntos de ruta introducidos en las trayectorias originales y suavizadas en 3D para cada agente del enjambre.

Trayectorias	Test 1		Test 2		Test 3		Test 4		Test 5	
	O	S	O	S	O	S	O	S	O	S
UAV_1	15	3	15	4	15	3	15	3	16	3
UAV_2	13	3	13	3	13	3	13	3	13	3
UAV_3	15	4	15	3	15	3	15	3	14	4
UAV_4	12	3	11	4	12	4	13	3	13	3
UAV_5	11	3	11	3	10	3	11	4	10	3

O=Trayectoria Original, S=Trayectoria Suavizada.

De manera individual la reducción de puntos de ruta es considerable, pero si se analiza su impacto sobre el conjunto de trayectorias creadas para el enjambre, se observa que el impacto global es muy significativo, tal y como se observa en la Figura 3.34, de la que se puede extraer que para algunos casos la reducción alcanzan el 76 % de los puntos de ruta y, que la media se sitúa en el 75,38 % de nodos eliminados, lo cual sin duda supone una mejora significativa tanto desde el punto de vista del comportamiento dinámico de los UAVs, como del tiempo total invertido en la navegación, ya que, a la hora de seguir las trayectorias prefijadas, la velocidad del UAV se reduce al aproximarse a un punto de ruta de cara a ganar precisión en el paso por el mismo. Además, al igual que en el caso 2D, se debe considerar que del conjunto de nodos que forman cada una de las trayectorias, uno corresponde a la ubicación del objetivo y, este nodo nunca puede ser eliminado. Por tanto, obviando este nodo y considerando el porcentaje de reducción sobre los que sí pueden ser eliminados, se obtiene que con el algoritmo propuesto se consigue una reducción de puntos de ruta intermedios de un 81,57 %.

El siguiente aspecto a analizar es como la reducción de nodos intermedios afecta o no a la reducción de la distancia total recorrida por el conjunto de UAVs y, con ello, una reducción también del tiempo empleado para alcanzar el destino. A diferencia del caso en 2D, en este caso la reducción es algo más palpable, tal y como se muestra en la Figura 3.35, debido principalmente a la consideración total del entorno y no únicamente del plano X-Y y, también, a que al planificar sobre una dimensión más, el grafo generado no se encuentra tan bien ajustado como en el caso anterior, por lo que los efectos del suavizado son más evidentes. Aunque esta reducción se sitúa en todos los casos por encima de los 5 *m*, su valor porcentual es bajo, situado en un 3 % de media, para un enjambre de 5 UAVs.

Por último, es imprescindible analizar cómo afecta el uso de este algoritmo al tiempo de cómputo empleado para generar dicha solución óptima, que mejore el comportamiento dinámico de los UAVs. Y, es en este punto donde el algoritmo

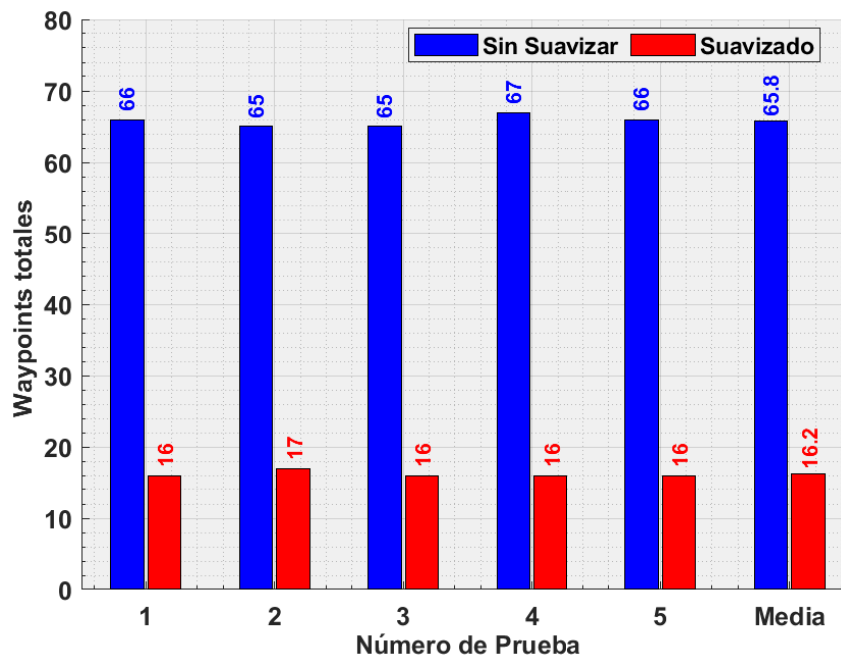


Figura 3.34: Comparativa del total de waypoints para trayectorias sin suavizar y trayectorias suavizadas en 3D para un enjambre de 5 UAVs.

propuesto presenta su principal inconveniente, tal y como se muestra en la Figura 3.36. Al emplear este algoritmo sobre un entorno 3D de grandes dimensiones, el tiempo empleado en recorrer las rutas originales, analizar que nodos pueden ser eliminados y, proponer una solución suavizada aumenta hasta situarse por encima de 1 *segundo*. Este aumento considerable del tiempo de cómputo se debe a la necesidad de tener que trabajar con mapas de ocupación en 3D, los cuales poseen información útil y detallada del entorno, pero que provocan que el uso y acceso de manera repetida a dicha información ralentice algoritmos como el propuesto en esta sección. A diferencia del algoritmo de planificación de trayectorias propuesto, en el cual se procesa el mapa de ocupación una única vez para generar el grafo, en el caso del método de suavizado de trayectorias se debe acceder a la información tridimensional del mapa de ocupación cada vez que se desea comprobar si el segmento que une dos nodos se encuentra libre de obstáculos o no.

Este resultado elimina el uso de este algoritmo para aplicaciones que requieran un cálculo y suavizado dinámico de las trayectorias en tiempo real, pero no limita su uso para una etapa de post-procesado que permita generar una solución óptima con mejor respuesta dinámica de los UAVs al eliminar el patrón Zig-Zag de las trayectorias originales obtenidas a partir de la solución global construida por los algoritmos de planificación de trayectorias en 3D basados en PRM. Junto a esta

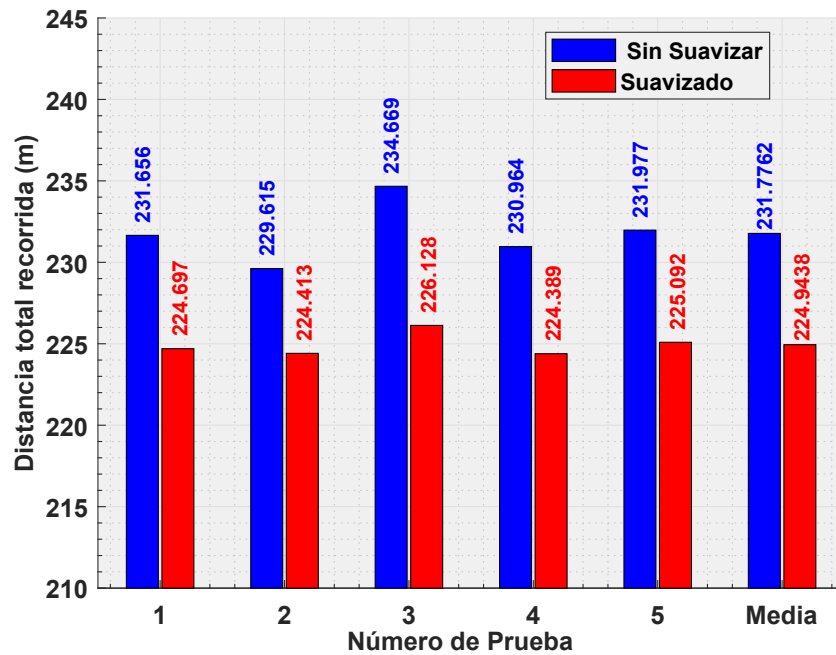


Figura 3.35: Comparativa de la distancia total recorrida por un enjambre de 5 UAVs para trayectorias sin suavizar y trayectorias suavizadas en 3D.

razón es importante considerar que el suavizado de trayectoria y, por tanto, la eliminación de puntos de ruta intermedios mejora los tiempos de respuesta del enjambre y genera una solución más eficiente en cuanto a distancia recorrida y tiempo de ejecución de la misión.

3.5. Conclusiones

Este capítulo recoge los desarrollos e implementaciones realizadas en el campo de la planificación de trayectorias para enjambres de UAVs, tanto a partir de exploraciones del entorno basadas en información 2D como en 3D.

Se propone un método de planificación de trayectorias en 2D basado en PRM para múltiples trayectorias, de modo que los diferentes UAVs participantes en la misión puedan alcanzar distintos objetivos. Además, el método ha sido testado y validado para su aplicación en 3 casos de uso diferentes como son: aquella situación en la cual todos los UAVs navegan hacia un mismo objetivo de cara a adquirir una formación concreta sobre dicha ubicación; la situación en la que cada uno de los UAVs alcanzan un objetivo previamente establecido; y, por último, aquella situación en la cual cada uno de los UAVs del enjambre va a un objetivo determinado,

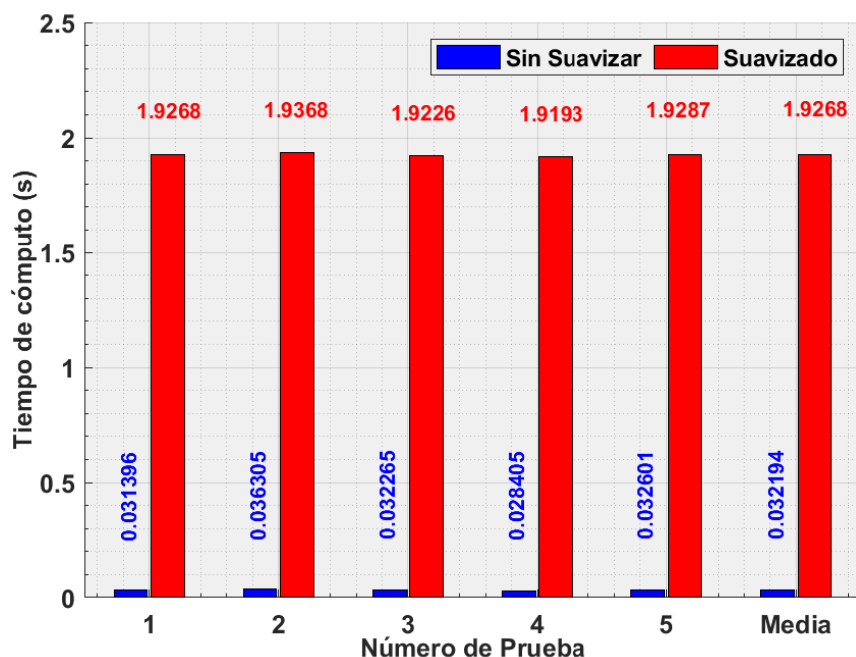


Figura 3.36: Comparativa del tiempo de cómputo empleado para la generación de trayectorias sin suavizar y suavizadas en 3D para un enjambre de 5 UAVs.

de tal manera que, la distancia total recorrida por el enjambre se minimice.

En el caso del vuelo en formación, se han generado 3 tipos distintos de formación de cara a adecuar la navegación a labores de extinción de incendios en entornos forestales y, mejorar así la eficacia de estos medios en la lucha contra el fuego.

Los resultados recogidos en el caso del planificador de trayectorias en 2D demuestran que la exploración del entorno y generación del grafo, a través del método propuesto basado en PRM, está justificado dado que el gasto computacional derivado de dicho proceso se amortiza a medida que las dimensiones del enjambre aumentan y, por tanto, se establece como una herramienta eficaz para resolver problemas de planificación de trayectorias en enjambres escalables. Junto a ello, se han validado los resultados obtenidos integrándolos en el marco de ROS y en el simulador *Gazebo*, como paso previo a su implementación en plataformas reales.

Para el método de planificación de trayectorias en 3D para enjambres de UAVs, se han introducido una serie de modificaciones sobre el método de 2D para su adaptación a problemas de una dimensión mayor. Por tanto, a partir de un método basado en PRM se ha generado un planificador de trayectorias sobre información del entorno en 3D, cuyos resultados avalan la escalabilidad del método de cara a emplearlo en enjambres de UAVs. En este caso, se incluye también su adaptación al marco de ROS y *Gazebo* para llevar a cabo simulaciones que permitan la validación

del método como paso previo a la realización de pruebas reales.

Ambos métodos presentan una segunda etapa basada en el algoritmo A^* , que permite generar una solución óptima en la que se minimiza la distancia recorrida por cada uno de los UAVs para alcanzar su objetivo. Además, se incluye, para el caso no etiquetado, una implementación del método húngaro, el cual permite obtener una asignación UAV-Objetivo que minimice la distancia total recorrida por el enjambre.

Por último, se ha desarrollado un método de suavizado de trayectorias que permita eliminar la principal desventaja de los métodos de planificación basados en PRM, como es la generación de trayectorias bajo un patrón de Zig-Zag. Este desarrollo de suavizado de trayectorias se basa en algoritmos de línea de vista para eliminar puntos de ruta intermedios en las trayectorias y, junto a la eliminación del patrón en Zig-Zag, mejorar el comportamiento dinámico de los UAVs durante la navegación sin supervisión.

Capítulo 4

Arquitectura de Navegación Autónoma para Entornos Dinámicos

Para poder implementar una arquitectura robusta y versátil que permita a un enjambre completar de manera autónoma y coordinada una misión es necesaria la combinación de un conjunto de capas de desarrollo que otorguen al enjambre de las herramientas necesarias para poder navegar de manera segura por el entorno. En el Capítulo 3 se recoge un capa de dicha arquitectura que permite establecer la solución a uno de los aspectos claves para lograr dicha autonomía en la navegación y que, no es otro que establecer una planificación global inicial de trayectorias seguras que permitan a los UAVs evitar colisiones con objetos estáticos presentes en el entorno, pero de cara a generar robustez y aumentar la aplicabilidad del sistema de múltiples UAVs propuesto se hace necesaria la introducción de capas al alto nivel que se basen en la recopilación e interpretación de información del entorno de cara a establecer acciones de control que permitan la navegación autónoma segura de cada uno de los UAV.¹

Un aspecto clave para la realización autónoma de misiones por parte de los enjambres de UAVs es la capacidad de detectar y evitar colisiones con obstáculos presentes en el entorno, tanto estáticos como dinámicos. En el caso de los enjambres de UAVs es evidente que el problema de la detección y evasión de obstáculos no se reduce únicamente a los objetos estáticos presentes en el entorno, sino que en el caso de los enjambres es inherente la presencia de objetos dinámicos correspondientes al resto de agentes que forman el enjambre. Por este motivo, se hace necesario

¹Publicaciones del autor relacionadas con el capítulo:

- Madridano, Á., Al-Kaff, A., Flores, P., Martín, D., & de la Escalera, A. (2020, September). Obstacle Avoidance Manager for UAVs Swarm. In 2020 International Conference on Unmanned Aircraft Systems (ICUAS) (pp. 815-821). IEEE. (6)

establecer, dentro de la arquitectura propuesta, una capa encargada de localizar y establecer la posición relativa entre los UAVs del enjambre y, generar una salida que permita llevar a cabo maniobras y acciones de control que eviten las colisiones entre miembros de un mismo enjambre, tal y como se esquematiza en la Figura 4.1, en la que cada UAV del enjambre navega sin supervisión hacia un objetivo concreto a través de una trayectoria previamente fijada, con los métodos y algoritmos descritos a lo largo del Capítulo 3.

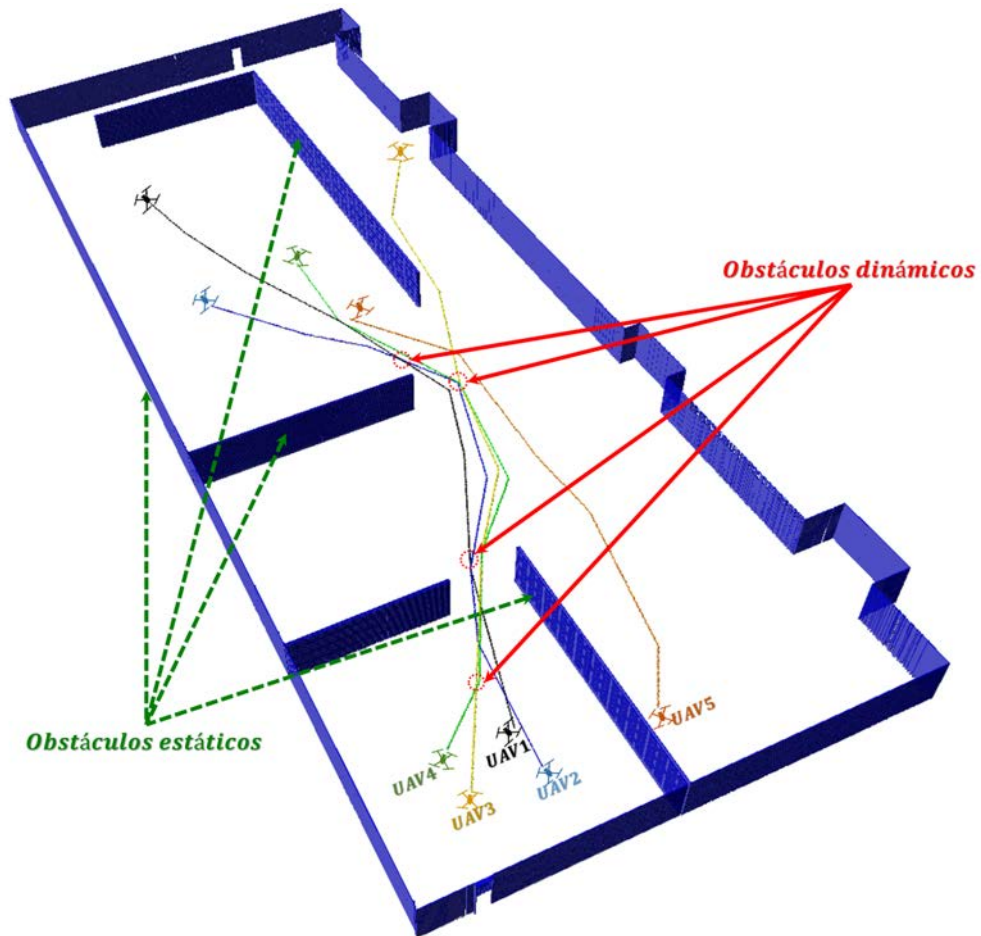


Figura 4.1: Problemática de la navegación autónoma en enjambre (6) ©2020 IEEE.

Sin embargo, el desarrollo de métodos destinados a detectar y evitar colisiones con obstáculos dinámicos sigue siendo un tema problemático al que se le busca una solución de cara al uso compartido futuro del espacio aéreo por parte de vehículos aéreos tripulados y vehículos aéreos no tripulados. Tanto es así, que desde la Organización de Aviación Civil Internacional (OACI) se trabaja activamente en la búsqueda de soluciones tecnológicas que, en primer lugar, garanticen la seguridad

en las operaciones de los UAVs en el espacio aéreo compartido y, en segundo lugar, proporcionen a estos vehículos capacidades similares a las de las aeronaves tripuladas por humanos a través de los sentidos del piloto (229).

Para resolver esta problemática, han surgido líneas de investigación con enfoques distintos que tratan de generar soluciones tecnológicas a la detección y evasión de obstáculos para enjambres de UAVs basadas en campos potenciales artificiales (20; 230; 231), en algoritmos dependientes de la velocidad del obstáculo (232; 233), en algoritmos bio-inspirados (234; 235), métodos basados en muestreo (236; 237; 238; 239), en algoritmos de aprendizaje por refuerzo y aprendizaje profundo (121) y, por último, en algoritmos de colisión basados en probabilidad (240).

En línea con estas investigaciones, a lo largo del presente capítulo se presentan un conjunto de métodos basados en diferentes tecnologías que permitan dotar a la arquitectura propuesta para navegación autónoma de UAVs en enjambre de un sistema de detección y evasión de obstáculos robusto y redundante en entornos dinámicos.

4.1. Sistema de Evasión de Obstáculos para Enjambres de UAVs.

La primera de las soluciones tecnológicas propuesta para afrontar el problema de la navegación autónoma de un enjambre de UAVs es establecer un sistema de gestión de detección y evasión de obstáculos formado por dos capas basadas en diferentes métodos. De esta forma, se dota a los UAVs del enjambre de una arquitectura robusta y redundante para la detección y evasión de colisiones en tiempo real.

La combinación de capas con diferente tecnología permite al enjambre de UAVs llevar a cabo maniobras de manera coordinada y centralizada y, en caso de pérdida de comunicaciones o enlace entre los UAVs que forman dicho enjambre, poder llevar a cabo una navegación segura sin supervisión de manera descentralizada, en la que cada UAV sea capaz de tomar sus propias decisiones y alcanzar una ubicación concreta sin colisionar con los objetos presentes en el entorno. Por tanto, tal y como se recoge en la Figura 4.2, el sistema de gestión de detección y evasión de obstáculos se encuentra dividido en dos capas o sub-sistemas: el primero de ellos consiste en un método basado en la detección, localización y posicionamiento relativo del conjunto de los UAVs del enjambre y, se encarga de actuar sobre la velocidad de crucero de los UAVs para evitar colisiones entre sí; el segundo de ellos, se basa en métodos de muestreo probabilísticos basados en PRM que permita, a través de la información 3D obtenida del entorno, generar una ruta alternativa que

permita evitar colisiones con objetos detectados en la trayectoria de cada uno de los UAVs.

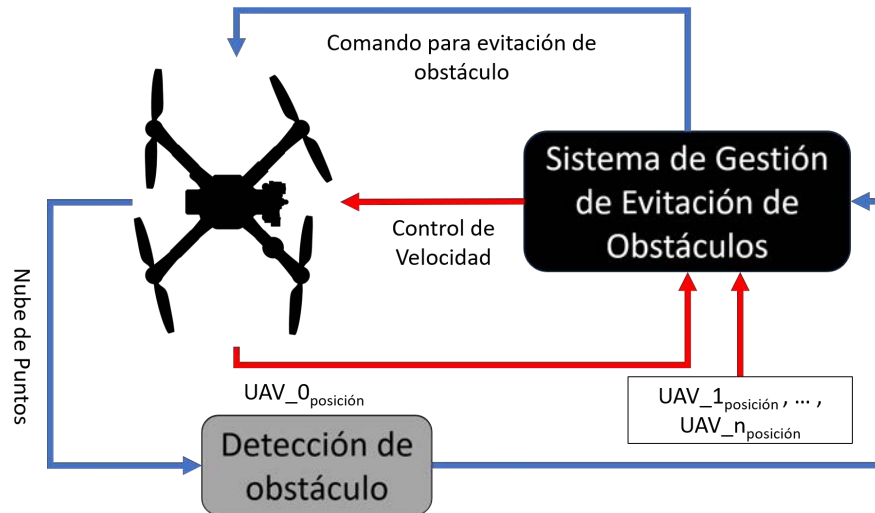


Figura 4.2: Sistema redundante de gestión de evasión de obstáculos.

A continuación, en el Apartado 4.1.1 y en el Apartado 4.1.2, se detallan y analizan cada una de las capas desarrolladas para proveer a los UAVs de un sistema de detección y evasión de obstáculos robusto en el que, la primera capa se establece sobre una arquitectura centralizada encargada de recoger el conjunto de posiciones de los UAVs del enjambre y realizar acciones de control sobre la velocidad de los mismos para evitar colisiones entre sí. Mientras que la segunda de las capas se dispone sobre una arquitectura descentralizada, en la que cada uno de los UAVs captura información 3D del entorno y realiza maniobras individuales de evasión de obstáculos en el caso de detectar una posible colisión con cualquier obstáculo, estático o dinámico, presente en el entorno.

4.1.1. Sistema de navegación basado en control de velocidad de crucero

Esta primera capa se establece como un desarrollo necesario para la navegación sin supervisión del enjambre a través de trayectorias prefijadas generadas mediante el uso de los métodos de planificación en 3D descritos en la Sección 3.2. Esta solución establece rutas seguras que consideran obstáculos estáticos presentes en el entorno, pero no consideran los posibles conflictos originados por coincidencia de puntos de ruta intermedios coincidentes a lo largo de las diferentes trayectorias de los UAVs del enjambre.

Por este motivo, se implementa una capa basada en una arquitectura centralizada enfocada a la modificación de los perfiles de velocidad de los UAVs a lo largo de las trayectorias prefijadas para permitir un movimiento seguro y, ante todo, coordinado de todo el enjambre de UAVs dentro de una misma área de trabajo. Esta capa se asienta sobre un control centralizado del enjambre, el cual se basa en las posiciones 3D de todos los UAVs, para llevar a cabo un control sobre las velocidades de desplazamientos de cada uno de los agentes de cara a evitar posibles colisiones cuando 2 o más UAVs se aproximen a una misma ubicación. Así pues, cuando el sistema detecta que al menos dos UAVs se encuentran próximos entre sí, aplica a cada uno de ellos un perfil de velocidad diferente, de tal manera que cada uno de los UAVs atraviese esa misma ubicación en distinto momento de tiempo, evitando de esta manera la colisión. Mientras que la navegación sin supervisión sea segura y, ningún UAV del enjambre se aproxime a otro por debajo de un margen de seguridad preestablecido, el sistema no actúa y cada UAV se desplaza a su velocidad de crucero máxima, determinada por la configuración del autopiloto.

Basado en la posición GPS de todos los UAVs, el nodo central se encarga de verificar la distancia euclídea entre los miembros del enjambre y, aplicar correcciones en la velocidad de crucero si la distancia entre 2 o más UAVs se reduce por debajo de un umbral predeterminado. La manera de modificar las velocidades de crucero de los UAVs es aplicando un factor de reducción, el cual se encuentra parametrizado y varía en función del identificador (*ID*) de cada UAV dentro del enjambre, reduciendo la velocidad de navegación de cada uno de los UAVs en distinta proporción.

Como se recoge en el Algoritmo 5, el primer paso consiste en establecer la configuración del enjambre, asignando a cada UAV un identificador dentro del sistema junto con una trayectoria prefijada para navegar sin supervisión por el entorno. Completada la configuración, el método implementado comienza un proceso iterativo en el que se realiza un seguimiento de la trayectoria de cada uno de los UAVs, comprobando la posición en cada instante de tiempo y, comparándola con las ubicaciones intermedias fijadas a lo largo de la trayectoria para verificar que la navegación autónoma de cada UAV está siguiendo la ruta preestablecida. Durante este proceso iterativo, no solo se va comprobando la trayectoria seguida por cada UAV, sino que se establece un segundo proceso encargado de establecer la posición relativa de cada UAV con respecto al resto de UAVs del enjambre, aplicando factores reductores de velocidad en el caso de que se cumplan las condiciones determinadas previamente para ello.

El factor de reducción de velocidad se calcula de acuerdo a la Ecuación 4.1 y se aplica a la velocidad de crucero del UAV indicado en el instante de tiempo $t+1$, tal y como se refleja en la Ecuación 4.2.

Algorithm 5: Control de velocidad para enjambres de UAVs

Input: *pose_sub* posición de los UAVs, *agent_ID* vector de ID de los agentes, *agent_paths* Trayectoria prefijada para cada UAV.

Output: *velocity_pub* Velocidad del UAVs.

1 **Define:** *Wp_radius* Radio para considerar un punto de ruta alcanzado, *Safety_thr* Distancia de seguridad entre UAVs, *agents* vector de agentes del enjambre, *waypoints_i* Vector con los puntos de ruta de cada trayectoria para cada UAV, *velocity_reducer* Parámetro encargado de reducir la velocidad nominal, *nWaypoint* Número de punto de ruta actual

2 **begin**

3 //Establecer configuración inicial del enjambre

4 **for** $i = 0; i < agent_ID.size(); i++$ **do**

5 | $agents[i] = \text{new Agent}(agent_ID[i], agent_paths[i])$

6 | $waypoints_i \leftarrow \text{generate_waypoints}(agent_paths[i])$

7 | $velocity_reducer = 1$

8 //Control del enjambre basado en la velocidad.

9 **for** $j = 0; j < agents.size(); j++$ **do**

10 | //Calculo de la distancia absoluta al próximo waypoint.

11 | $Abs_Goal_distance_j = waypoints_j[nWaypoint_j] - pose_sub_j$

12 | //Comprobar la distancia del UAV con el resto de agentes del enjambre.

13 | **for** $h = j+1; h < agents.size(); h++$ **do**

14 | $euclidean \leftarrow \text{distance_euclidean}(pose_sub_j, pose_sub_h)$

15 | **if** $euclidean < Safety_distance$ **then**

16 | | $Velocity_reducer = (h+1)*3.0$

17 | $speed_pub_j = nominal_velocity / velocity_reducer$

18 | //Comprobar si el punto de ruta actual ha sido alcanzado.

19 | **if** $Abs_Goal_distance_j < Wp_radius$ **then**

20 | **if** $nWaypoint_j < waypoints_j.size()$ **then**

21 | | //Cargar próximo punto de ruta.

22 | | $nWaypoint_j = nWaypoint_j + 1$

23 | **else**

24 | | $speed_pub_j = 0.0$

$$F_{reductor} = (ID_{UAV} + 1) \times 3, 0 \quad (4.1)$$

$$Vel_{t+1}(UAV) = \frac{Vel_t(UAV)}{F_{reductor}} \quad (4.2)$$

Como se observa en la Ecuación 4.1, el factor reductor de la velocidad varía en función del identificador del UAV, al que se le suma el valor de 1 debido a que al primer vehículo del enjambre se le asigna el ID 0. En cuanto al factor multiplicativo de 3, se establece como un valor experimental que incrementa dicho factor para hacer significativa la reducción para el caso más laxo, es decir, para el primer UAV del enjambre. La idea de establecer una dependencia entre el factor de reducción y el ID se basa en dos aspectos, los cuáles son detallados a continuación.

El primero de ellos se basa en que si la reducción aplicada a cada UAV fuera similar, no se puede asegurar la efectividad del método implementado para evitar colisiones entre los dos objetos dinámicos, ya que la reducción de velocidad siempre se aplica a $n - 1$ UAVs, siendo n el número de vehículos implicados en el conflicto, es decir, si son 2 los UAVs en conflicto uno reduce la velocidad y el otro continúa con su movimiento habitual, pero en caso de que sean 3 o más los UAVs implicados, uno mantendría constante su velocidad evitando la colisión, pero los otros 2 UAVs podrían colisionar si la reducción de la velocidad fuera de la misma magnitud. El segundo de los aspectos está relacionado con la explicación del primer aspecto y es que, la reducción siempre se aplica a $n - 1$ vehículos envueltos en el conflicto, estableciendo siempre la comparación desde el UAV de menor ID hasta el UAV de mayor ID . Para entender mejor este aspecto se incluyen la Figura 4.3 y la Figura 4.4.

En la Figura 4.3 se muestra la actuación del método propuesto sobre el UAV₁ de un enjambre compuesto por 5 agentes, en la que se puede observar, en la parte superior la distancia de dicho UAV al resto de agentes del enjambre y, en la parte inferior la aplicación del factor de reducción cuando se cumple la condición de que el UAV₁ se encuentre a menos de 3.5 m , valor del umbral de seguridad, de distancia del resto.

En la Figura 4.4, se muestra la misma actuación, pero en este caso para el UAV₄. En este caso, la posición de dicho UAV se compara con el resto de agentes con los cuales no se ha comparado aún, que para un enjambre de 5 UAVs, dicha comparación se limita al UAV número 5. De esta forma, cuando el UAV₄ se encuentra a menos de 3.5 m de cualquiera de los agentes con un identificador menor son estos los que reducen su velocidad para evitar la colisión, mientras que en el caso de la comparativa de posiciones con el UAV₅, es el agente identificado con el número 4 el que reduce su velocidad cuando se produce una situación de inseguridad.

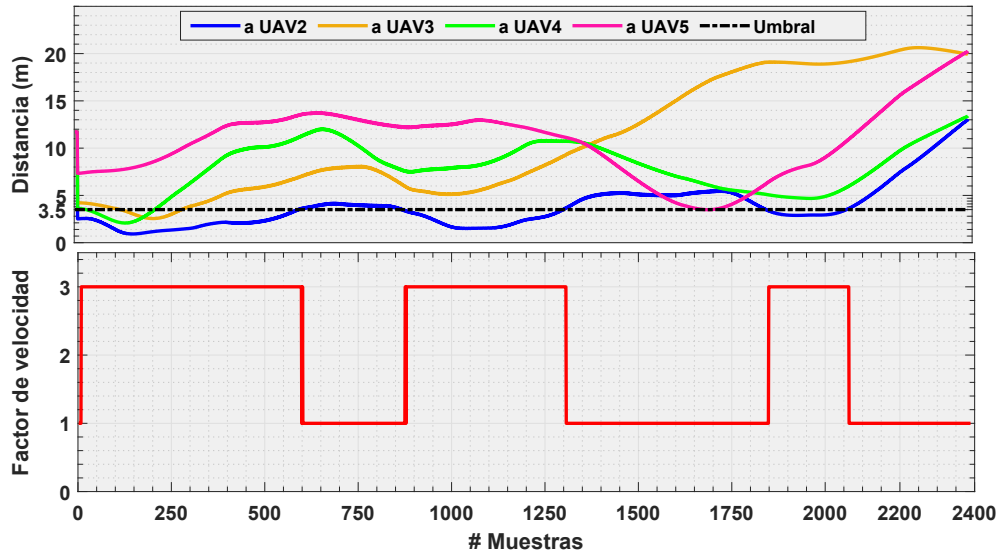


Figura 4.3: Distancias relativas y aplicación de factor de reducción para el UAV₁ (6)
 ©2020 IEEE.

Esta forma de actuar permite, no solo garantizar la navegación segura sin supervisión, sino que permite una actuación más coordinada del enjambre al aplicar factores de reducción más altos a aquellos UAVs que se comparan con menos vehículos y que, por tanto, tienen menos probabilidad de aplicar el factor reductor de la velocidad y, aplicando un factor reductor de menor orden de magnitud a aquellos UAVs que al poseer un identificador más bajo se comparan con mayor cantidad de agentes y, por tanto, poseen más probabilidad de que el sistema de control actúe sobre su velocidad de crucero.

La implementación de este método ha sido testada y validada sobre la arquitectura de software presentada y descrita en la Sección 3.3.1 para un enjambre de 5 UAVs, capaces de navegar sin supervisión y completar una misión sin colisiones gracias a la aplicación combinada del método de planificación en 3D y el sistema de evasión de colisiones, presentado en esta sección, para enjambres de UAVs basado en el control de velocidad. Junto con esta validación cualitativa, que permite comprobar, en un entorno de simulación, la aplicabilidad de los algoritmos propuestos hasta el momento, se encuentran la Figura 4.3, la Figura 4.4, la Figura 4.5 y la Figura 4.6 que permiten analizar desde un punto de vista cuantitativo el funcionamiento de dicho sistema sobre el enjambre.

En cada una de estas figuras aparecen reflejados, por un lado, las distancias relativas entre los diferentes UAVs del enjambre y, por otro lado, en qué momento el factor de reducción es aplicado sobre el UAV en cuestión. Se ha de añadir que,

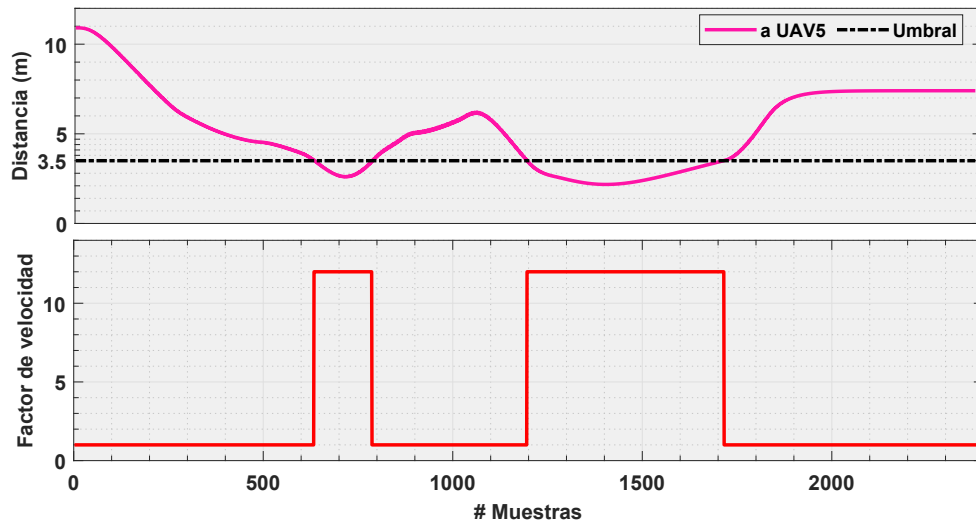


Figura 4.4: Distancias relativas y aplicación de factor de reducción para el UAV₄ (6)
 ©2020 IEEE.

para todos las figuras, los datos empleados son datos extraídos de una simulación sobre el entorno de Gazebo para un enjambre de UAVs que navega empleando el firmware de *PX4*, de tal manera que, estos datos simulados son similares a los que se obtendrían de cada uno de los autopilotos embarcados en plataformas reales, siendo la posición GPS de cada uno de los UAVs el dato de entrada al algoritmo que permite calcular la distancia relativa entre UAVs, para determinar si es necesario o no aplicar el control de velocidad. Para la obtención de la posición relativa de un UAV con respecto a otro se lleva a cabo una transformación de coordenadas desde el sistema de coordenadas geográficas, expresadas en Longitud y Latitud, al sistema de coordenadas Universal Transverse Mercator (UTM), en el cual las magnitudes se expresan en *metros*, facilitando el cálculo de la posición relativa de unos agentes con respecto al resto.

Para el caso de la Figura 4.5, se observa cual sería la actuación del método sobre el segundo UAV del enjambre. En este caso, y siguiendo la Ecuación 4.1, el factor de reducción introducido es de 6, al ser el UAV₂ representado con el *ID* número 1. Si se observan los datos recogidos a lo largo de la simulación, se puede apreciar que, el número de veces que el sistema tiene que intervenir es superior al número de veces que interviene en UAVs de mayor *ID*.

En la Figura 4.6, se recogen los datos obtenidos de la comparación del UAV₃ con aquellos agentes del enjambre que poseen un identificar mayor. En este caso, el factor de reducción es empleado en dos momentos concretos debido a la proximidad de dicho UAV con el UAV₄ por debajo del umbral establecido.

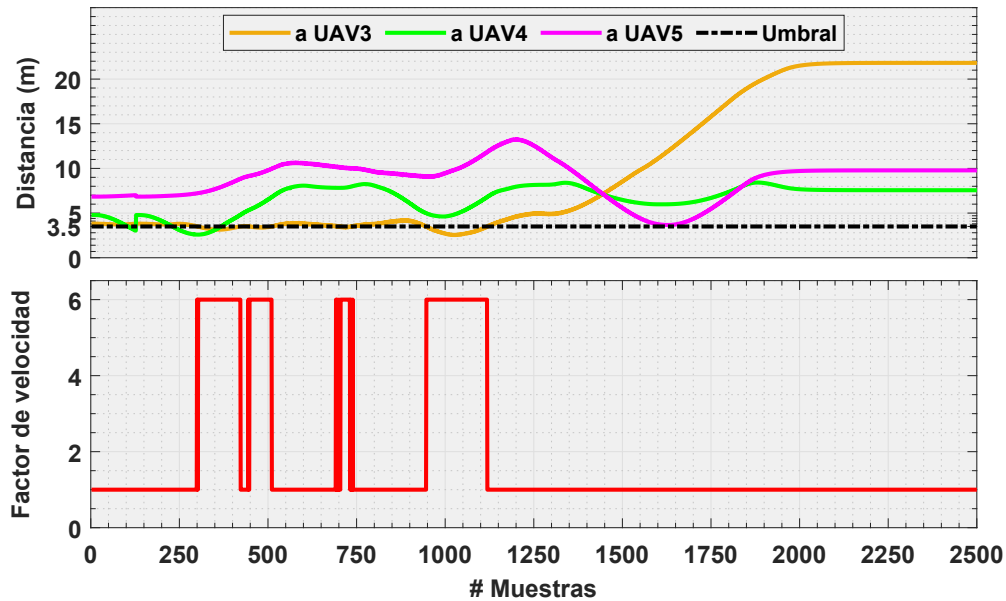


Figura 4.5: Distancias relativas y aplicación de factor de reducción para el UAV₂ (6)
 ©2020 IEEE.

Del análisis de todas las gráficas se pueden extraer que el método implementado sobre el enjambre tiene un funcionamiento acorde a lo esperado y, permite, mediante modificaciones en tiempo real sobre las velocidades de los UAVs, garantizar la navegación segura y sin supervisión del enjambre en un entorno dinámico. Es importante puntualizar que, durante la simulación, la tasa de actualización de información se sitúa en 0,2 s, es decir, que con una frecuencia de 50 Hz el sistema es capaz de introducir acciones de control sobre los UAVs en tiempo real.

4.1.2. Sistema de detección y evasión de colisiones basado en percepción del entorno

Esta segunda capa se establece como un desarrollo complementario para la arquitectura de navegación autónoma del enjambre, tomando como punto de partida de la solución generada por el planificador de trayectorias propuesto en el Capítulo 3, es decir, contando con un conjunto de trayectorias prefijadas como entrada a la capa. La combinación de estos métodos de planificación de trayectorias con el algoritmo de control de velocidad permite al enjambre navegar sin supervisión evitando colisiones con objetos estáticos incluidos en el mapa de ocupación y, a su vez, evitando colisiones entre los propios UAVs que forman parte del enjambre. A pesar de esta combinación, se hace necesario implementar, dentro de la arqui-

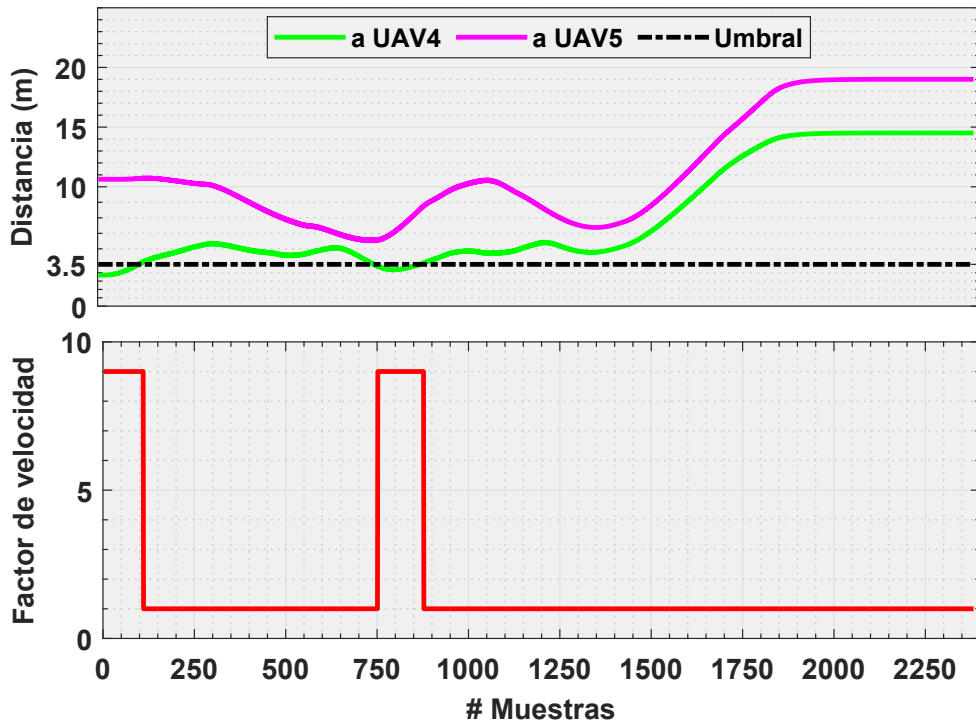


Figura 4.6: Distancias relativas y aplicación de factor de reducción para el UAV₃ (6)
©2020 IEEE.

tectura propuesta, métodos que permitan detectar objetos adicionales al modelo 3D y obstáculos dinámicos, de cara a obtener un sistema de navegación autónoma seguro y robusto.

A diferencia de la primera capa, este sistema se basa en una arquitectura descentralizada que actúa sobre el lazo de control de cada UAV de manera individual, complementando la información e instrucciones introducidas a través de la capa previamente descrita o, permitiendo la navegación autónoma del UAV en caso de pérdidas de las comunicaciones. La centralización de la primera capa provoca que una pérdida de las comunicaciones, entre cualquiera de los agentes y el nodo central, implique un riesgo para la navegación segura del enjambre y, de ahí, que se establezca la presente capa bajo un control descentralizado para la consecución de un sistema de gestión de evasión de obstáculos global, redundante y robusto.

El método implementado en esta segunda capa se basa en la percepción 3D del entorno para permitir la detección y evasión de los obstáculos, tanto estáticos como dinámicos, del entorno. Y es que, tanto los métodos de planificación 2D como los métodos de planificación en 3D trabajan con información estructural del entorno, como pueden ser datos relacionados con la ejecución de proyectos como

son los planos y modelo 3D de los edificios construidos vacíos, siendo necesario, por lo tanto, un sistema de percepción del entorno capaz de captar información del mismo, de cara a poder navegar sin colisiones en áreas dinámicas.

Para llevar a cabo la percepción 3D del entorno, es importante considerar que, al tratarse de plataformas aéreas, los sensores empleados deben poseer un conjunto de características y propiedades, en términos de tamaño, peso y consumo, que facilite su implantación en los UAVs. Esta percepción 3D del entorno se puede obtener a partir de dispositivos basados en tecnología LiDAR o en cámaras estéreo.

Un sensor tipo LiDAR consiste en un dispositivo que permite obtener datos de distancia en 3D a partir de un emisor láser y la medida de tiempo de retraso entre la emisión del pulso y la detección de la señal reflejada en dicho objeto. Aunque en un principio, esta tecnología está ligada a dispositivos con un peso considerable y unos valores de consumo incompatibles con UAVs propulsados a través de sistemas eléctricos, los avances en este campo han desembocado en dispositivos LiDAR compatibles con su uso en UAVs, como por ejemplo el LiDAR de estado sólido Benewake CE30, capaz de suministrar con una precisión de centímetros a una distancia máxima de 30 *metros*, con unas dimensiones (83 x 57 x 53 *mm*), peso (334 *g*) y consumo (36 *W*) acordes a las plataformas aéreas empleadas.

La idea principal de este método es poder detectar obstáculos presentes en la trayectoria del UAV y establecer maniobras de evasión sobre el mismo, a partir de la generación de rutas alternativas seguras y libres de obstáculos y de los comandos de control necesarios para corregir la trayectoria. Las maniobras de evasión mediante el recorrido de rutas alternativas segura se lleva a cabo a partir de la nube de puntos generada por el LiDAR, la cual permite generar una reconstrucción del entorno en 3D, tal y como se muestra en la Figura 4.7.

A partir de esta nube de puntos con información tridimensional del entorno, se construye un mapa de ocupación en 3D, como el mostrado en la Figura 4.8, para obtener una representación del entorno basada en celdas, las cuales pueden ser analizadas mediante el uso de librerías como Octomap. A partir del mapa de ocupación se puede conocer el estado de las celdas y, determinar si se encuentran libres u ocupadas, lo cual permite conocer si las coordenadas correspondientes al centro de dicha celda se encuentran libres de obstáculo o no. A partir de esta información, algoritmo implementado dentro de la capa puede determinar si algún obstáculo se encuentra presente en la trayectoria prefijada y seguida por UAV y, en función de ello, establecer una ruta alternativa y acometer las maniobras necesarias para sortear dicho obstáculo.

En caso de detectar un obstáculo presente en la trayectoria del UAV, se emplea el método de planificación en 3D propuesto en este trabajo para, mediante la combinación de algoritmos de PRM y de A* para explorar el entorno en busca rutas alternativas que permitan evitar los obstáculos detectados y, encontrar la

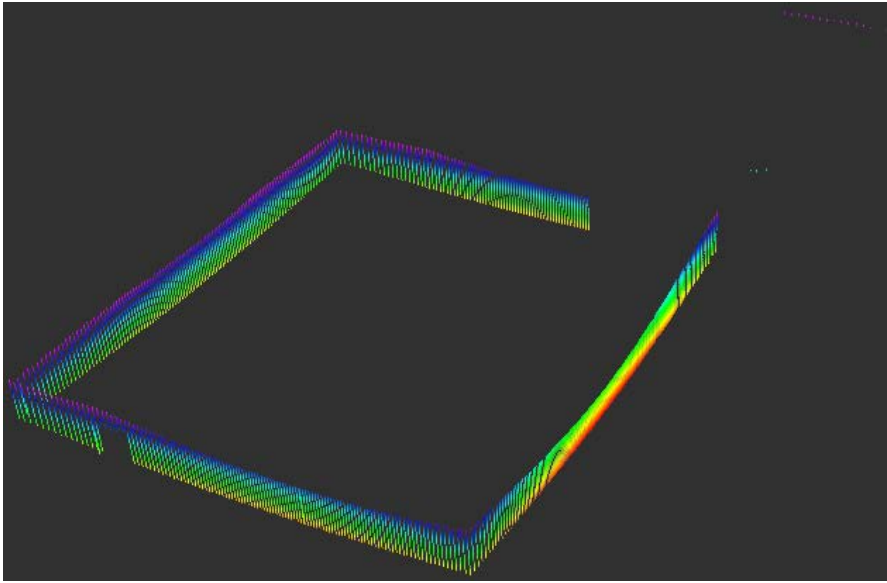


Figura 4.7: Información 3D basada en la Nube de puntos de un LiDAR (6) ©2020 IEEE.

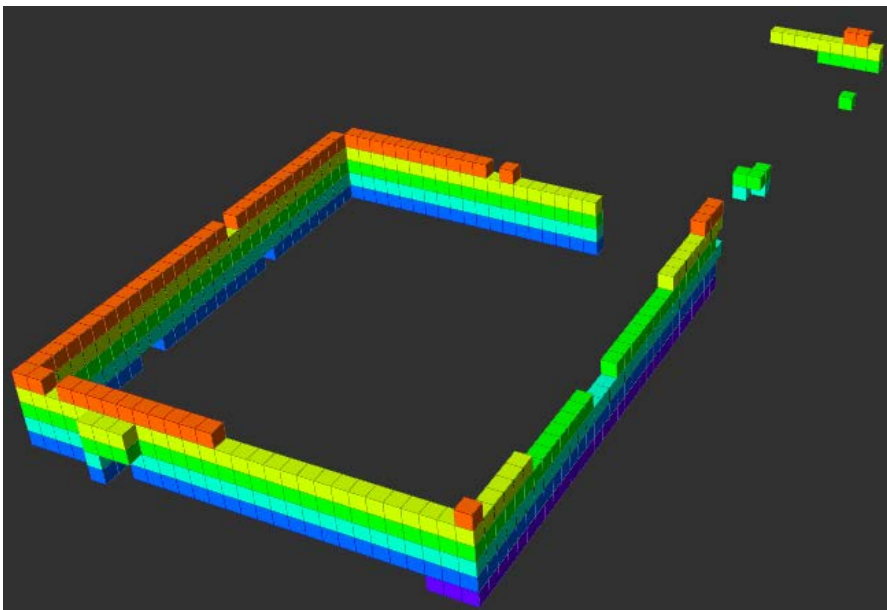


Figura 4.8: Mapa de ocupación generado a partir de la información 3D capturada por el LiDAR (6) ©2020 IEEE.

trayectoria óptima para evitarlo. De esta forma, a medida que el UAV se desplaza por el entorno hacia una ubicación de destino es capaz de detectar obstáculos

y evitarlos mediante el seguimiento de trayectorias alternativas que permitan la navegación segura sin supervisión.

Para detallar y entender el funcionamiento del método implementado se incluye la Figura 4.9, la cual recoge un diagrama de flujo sobre los procedimientos seguidos a lo largo del método para poder explorar el entorno y generar nuevas rutas alternativas que permitan al UAV alcanzar la ubicación de destino sin colisionar con ningún obstáculo.

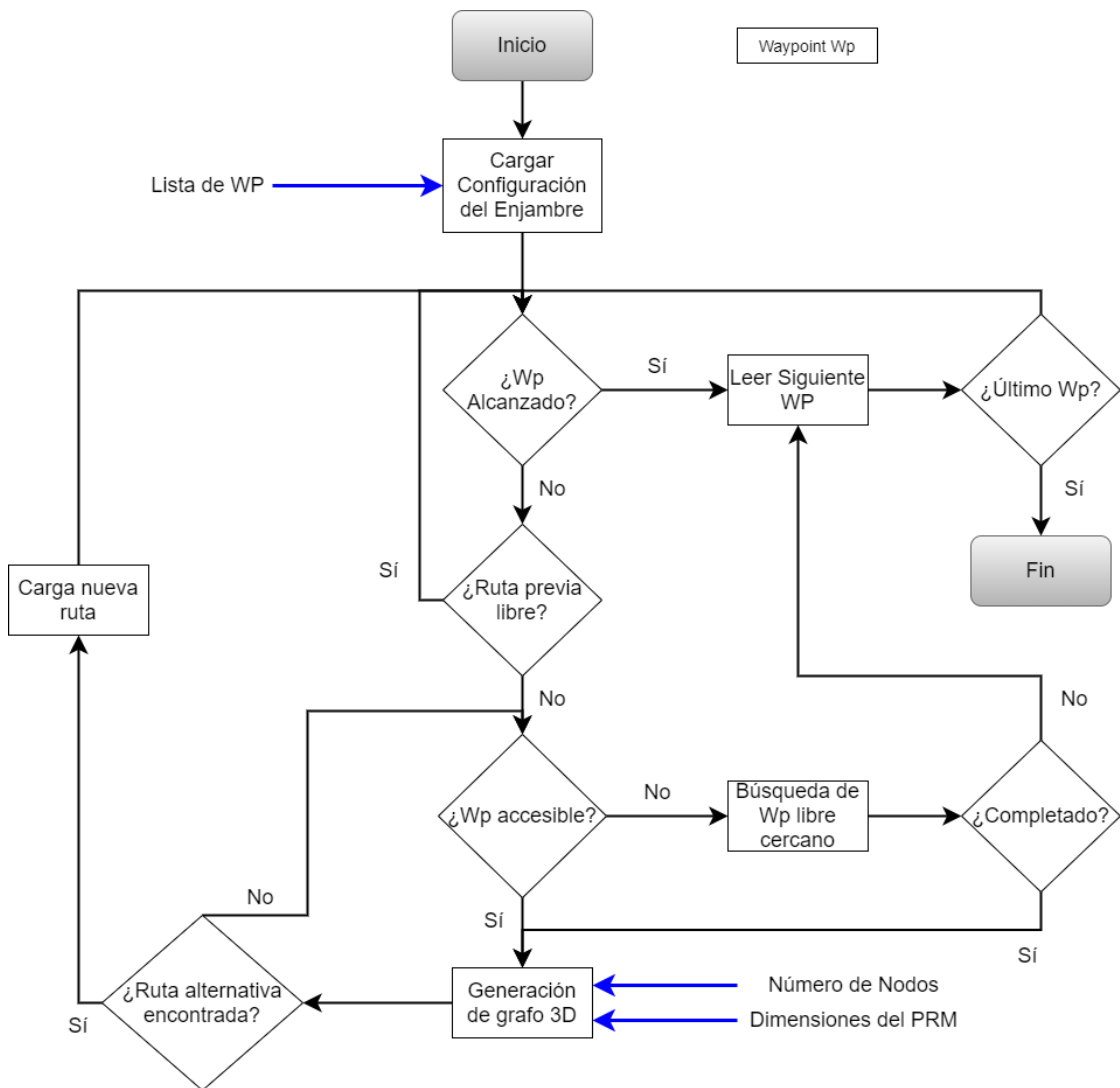


Figura 4.9: Diagrama de flujo del método de detección y evasión de obstáculos basado en información 3D del entorno (6) ©2020 IEEE.

El primer paso consiste, al igual que en la capa anterior, en cargar la configu-

ración inicial del UAV, con la lista de puntos de ruta a recorrer. A partir de este momento, se comienza un proceso iterativo con dos funcionalidades claras. La primera, establecer un seguimiento de la posición actual del UAV para comprobar si se sigue la trayectoria prefijada en caso de no detectarse obstáculos. Y, la segunda, se orienta a la detección de obstáculos dentro de la trayectoria prefijada y, en caso positivo, establecer un procedimiento que permita generar una trayectoria segura alternativa y establecer las maniobras necesarias para modificar el rumbo.

El flujo seguido en dicho proceso iterativo es el siguiente. Se comprueba si el punto de ruta cargado ha sido alcanzado. En caso afirmativo, se analiza si corresponde a la última ubicación de la trayectoria y, por tanto, se da por alcanzado el objetivo, o si, por el contrario, no es el último y es necesario que el UAV continúe navegando hacia una nueva ubicación, volviendo al inicio del proceso iterativo. En caso de que la ubicación intermedia, correspondiente a un punto de ruta, no haya sido alcanzada, se comprueba si la ruta prefijada correspondiente continúa libre de obstáculos o no, volviendo al inicio del proceso iterativo en el caso de no detectarse la presencia de obstáculos.

En caso de que sí se detecte un obstáculo en la ruta prefijada, primero se comprueba si dicho obstáculo ha cambiado la accesibilidad de la ubicación a alcanzar. En caso de que el punto de ruta sea inalcanzable, porque el obstáculo se encuentra en la ubicación del punto de ruta, se busca un nodo libre disponible en un radio cercano a dicho punto y, en caso de ser encontrado, se establece como nueva ubicación de destino. Si la búsqueda de esta ubicación cercana libre no se encuentra disponible debido a la presencia de un obstáculo de grandes dimensiones, se carga un nuevo punto de ruta de la trayectoria prefijada y, se repite el proceso iterativo.

Si al detectar un obstáculo en la trayectoria prefijada el punto de ruta sigue siendo accesible o, si se encuentra en las inmediaciones de dicho punto de ruta una ubicación segura a la que navegar, comienza el proceso por el cual se trata de establecer una ruta alternativa segura, tal y como se muestra en la Figura 4.10. Dicha figura, que muestra el resultado del algoritmo en simulación cuando un objeto aparece dentro de la trayectoria prefijada (línea amarilla), permite entender mejor la solución ofrecida por el método implementado. En ella se ve como, a partir de la información 3D obtenida por los sensores embarcados en el UAV se puede llevar a cabo la detección de conflictos en las trayectorias a seguir y, la generación de rutas alternativas que garanticen la seguridad del UAV durante la navegación.

El desarrollo detrás de la búsqueda, cálculo y generación de la ruta alternativa segura se basa en el planificador de trayectorias 3D presentado en este trabajo y que, combina algoritmos de PRM y A* para encontrar la ruta alternativa óptima que minimice la distancia recorrida por el UAV al cambiar el rumbo. Para ello, se establecen una serie de parámetros que permiten al algoritmo PRM generar un grafo en 3D con un conjunto de rutas alternativas posibles. Sobre ese grafo en 3D,

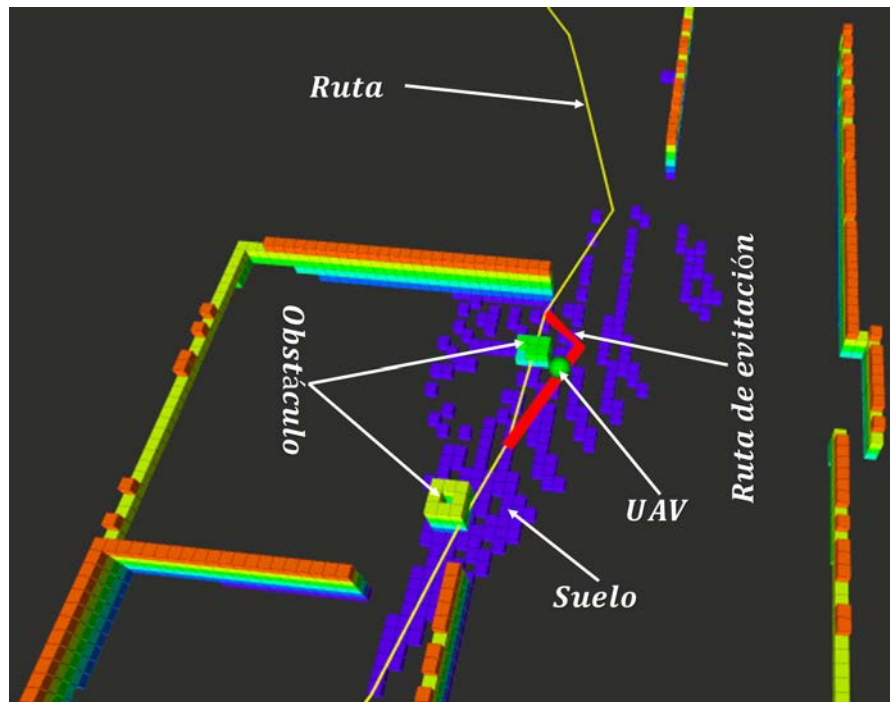


Figura 4.10: Ruta alternativa segura como solución para la navegación sin supervisión de un UAV (6) ©2020 IEEE.

se aplica el algoritmo A* para seleccionar aquella que se erige como la solución óptima en términos de distancia recorrida. Como se ha indicado, el algoritmo de PRM posee como entradas una serie de parámetros que lo definen, tales como el número de nodos a emplear y las dimensiones mínimas y máximas del mapa de ocupación sobre las que generar dicho grafo, tal y como se muestra en la Figura 4.11. De este modo, se puede adaptar el área de exploración en función de la distancia al punto de ruta que se desea alcanzar y, en caso de que dicho punto de ruta sea inalcanzable, aumentar el área de exploración a la nueva ubicación establecida. Por este motivo, el número de nodos a emplear es también un parámetro configurable dinámicamente en función del área a explorar de cara a poder mantener la densidad de nodos generada y encontrar una solución óptima como alternativa a la trayectoria previa. La posibilidad de contar con un algoritmo de PRM altamente parametrizable permite realizar exploraciones acorde a las necesidades de cada caso concreto y, permite trabajar con información 3D acotada, lo cual reduce los tiempos necesarios para generar la exploración 3D y, poder establecer una ruta alternativa óptima, los cuáles se sitúan en un orden de magnitud de milésimas de segundo, lo cual permite obtener una rápida respuesta y, poder establecer una trayectoria alternativa segura en tiempo real.

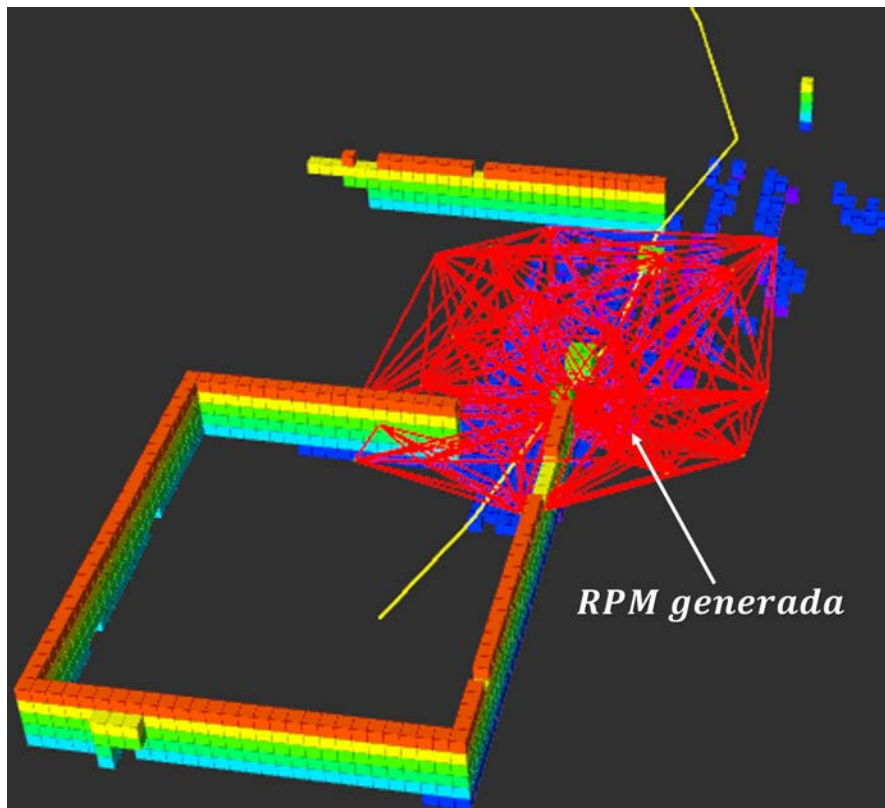


Figura 4.11: Grafo 3D generado para la obtención de rutas alternativas que eviten la colisión con los obstáculos detectados (6) ©2020 IEEE..

Una vez generada la trayectoria alternativa para continuar con la navegación, se cargan los nuevos puntos de ruta que permiten la modificación del rumbo y se inicia el proceso iterativo de seguimiento sobre la navegación autónoma del UAV.

Este método ha sido testado y validado sobre simulación en múltiples entornos de Gazebo. Es importante considerar la aplicabilidad de este método en operaciones de emergencias o incendios, dónde el área de actuación es tan cambiante. En el caso de los incendios urbanos, la posibilidad de tener acceso a información estructural de las viviendas (planos, infografía en 3D), permite el uso de los métodos de planificación de trayectorias descritos en el Capítulo 3, para establecer rutas que permitan alcanzar un destino, pero se hace necesario un método como el propuesto que permita la detección de objetos y personas dentro de los edificios y la posibilidad de alcanzar ubicaciones remotas de manera autónoma adaptándose a las condiciones cambiantes del entorno.

Un aspecto importante dentro del método propuesto es poder establecer un marco de coordenadas común al que referenciar todos los elementos del entorno.

Para ello, ROS cuenta con una herramienta llamada *tf* que permite establecer, de manera estándar, un seguimiento de los marcos de coordenadas y establecer una transformación de los datos de tal manera que se pueda establecer un marco común y cada componente del sistema pueda conocer su relación con otros marcos de coordenadas (241). De esta forma, el uso de transformadas permite, dentro del método propuesto, conocer la relación entre los datos del LiDAR y los del UAV y establecerlos sobre un frame común, tal y como se muestra en la Figura 4.12.

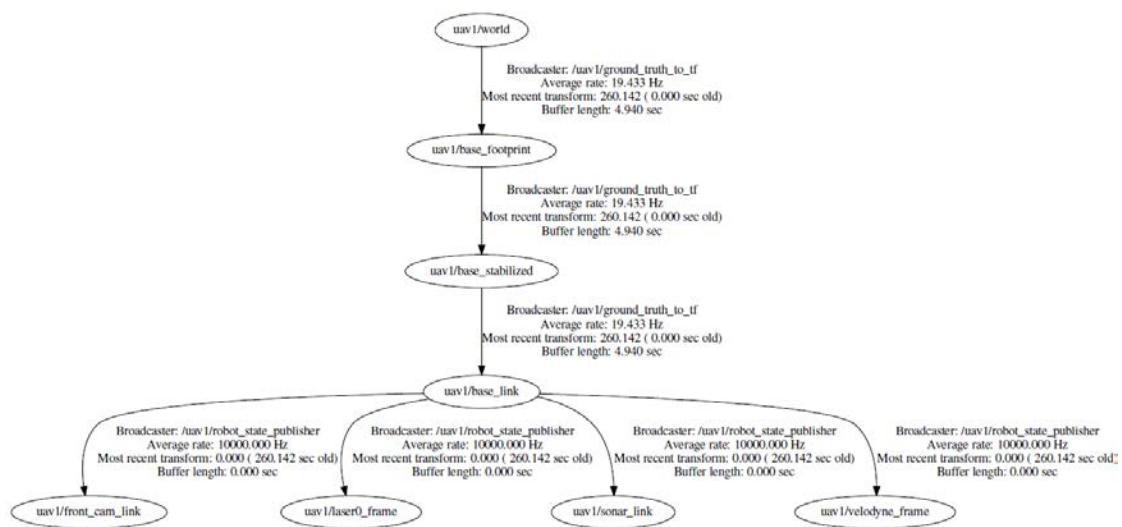


Figura 4.12: Relación de marcos de coordenadas para el método de evasión de obstáculos propuesto.

Para la validación del método se han empleado diferentes entornos, con una densidad de obstáculos variable, como son entornos urbanos y entornos forestales con la presencia de objetos dinámicos que simulen la presencia de UAVs externos al enjambre, como el de la Figura 4.13. De esta forma, se ha refrendado la aplicabilidad del método a diferentes ámbitos y, comprobando la capacidad de esta tecnología para detectar obstáculos dinámicos, incluidos otros UAVs. Los resultados cualitativos derivados de la simulación de este sistema de detección y evasión de obstáculos basado en información 3D, permiten concluir que dicho sistema es capaz de llevar a cabo detección de todo tipo de obstáculos y, replanificar una trayectoria de evasión alternativa que evite la colisión en tiempo real.



Figura 4.13: Entorno de simulación forestal con presencia de UAVs externos al enjambre.

4.2. Sistema de Navegación Basado en Perímetros Virtuales

La alta versatilidad de los enjambres de UAVs provoca que, en determinadas ocasiones, cada uno de los agentes se encuentre desarrollando una tarea diferente. De entre las distintas tareas, se encuentran aquellas relacionadas con la monitorización y vigilancia de un área determinado que requieren del mantenimiento de un UAV a una determinada altura sin variar dicha posición en el plano XY. Esta situación de estaticidad o *hover* provoca que, sistemas de evasión de colisiones entre UAVs de un mismo enjambre como el descrito en la Sección 4.1.1 no sean útiles para UAVs en maniobras de *hover*, por lo que es necesario contemplar otras soluciones que permitan a los UAVs trabajando en estas tareas, realizar maniobras de evasión sin perder el objetivo de su misión. Esta situación estática se une a la, también, posible situación de que alguno de los UAVs del enjambre no dispongan de la tecnología embarcada necesaria para implementar el método de detección de obstáculos descrito en la Sección 4.1.2.

Por este motivo, se presenta un nuevo método, basado en los sistemas de alerta de tráfico y evasión de colisión empleados en las aeronaves comerciales tripuladas y, que se encargan de prever posibles colisiones entre las diferentes aeronaves, trabajando de manera independiente a los servicios de tránsito aéreo. Su funcionamiento se basa en un sistema de transpondedor embarcado que se encarga de

intercambiar información de distancia, rumbo y altitud, con sistemas similares embarcados en otras aeronaves cercanas. Según el estándar establecido por la OACI, existe tres posibles sistemas anticolidión, clasificados en función de su actuación, yendo desde un estándar I, en el que solo se proporciona una alerta, al estándar III, aún en desarrollo, en el que se proporciona una alerta junto con una resolución de conflictos en 3D. En posición intermedia, se encuentra el estándar II, al cual se asemeja el método propuesto, y se basa en una alerta de tráfico más una resolución de conflictos en el plano vertical (242).

Por este motivo se presenta un método de evasión de colisiones para enjambres de UAVs basado en perímetros virtuales alrededor de cada uno de los agentes del enjambre. El desarrollo de aplicaciones de UAVs relacionadas con estas barreras geográficas virtuales han aumentado en los últimos años orientadas, sobre todo, a la integración futura de los UAVs junto con la aviación tradicional en un espacio aéreo compartido (243; 244; 245). Dicho método, proveería a la arquitectura propuesta de una nueva capa redundante de cara a la evasión de colisiones entre UAVs del enjambre.

En este caso, se establece, sobre la arquitectura de enjambre implementada, un método que permita a los UAVs en situación de *hover* llevar a cabo maniobras que impliquen movimientos verticales para evitar posibles colisiones con otros UAVs del enjambre que naveguen sin supervisión en un radio cercano. Para ello, se establecen dos esferas de seguridad de distintos radios alrededor de cada agente, tal y como se muestra en la Figura 4.14.

La esfera exterior, de color amarillo, constituye un primer volumen de seguridad alrededor del UAV cuyo radio, totalmente parametrizable, establece un margen de seguridad que, en caso de ser rebasado por otro UAV emitiría un aviso. Este aviso se mantiene mientras ambos UAVs se encuentren en un margen de distancia comprendido entre el primer y el segundo anillo.

En el caso de la esfera interior, de color rojo, se establece un radio, de nuevo parametrizable, que permite delimitar el volumen a partir del cual es necesario realizar una maniobra de evasión de obstáculos. De esta forma, si a pesar de la advertencia, el movimiento de alguno de los UAVs provoca que la distancia entre ambos se reduzca, estableciéndose por debajo del margen de actuación marcado, se realiza una maniobra evasiva que garantiza la seguridad del sistema. Esta maniobra ha sido definida para que se resuelva el conflicto en el plano vertical y, ambos UAVs modifiquen su altura de vuelo para evitar la posible colisión. Una vez terminada la situación de alerta, ambos UAVs recuperan su altitud marcada antes de generarse el conflicto.

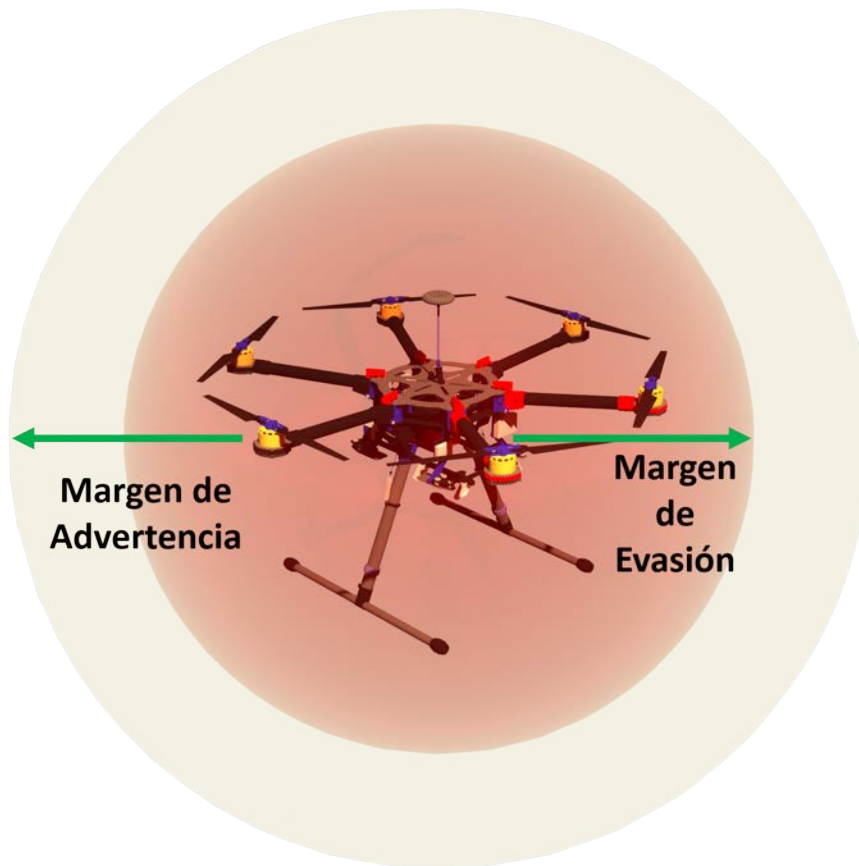


Figura 4.14: Sistema de evasión de colisiones basado en perímetros virtuales.

4.3. Conclusiones

En este capítulo se presentan un conjunto de sistemas que permiten navegar, sin supervisión, a un enjambre de UAVs de forma segura, gracias a la capacidad de estos sistemas para detectar y evitar posibles conflictos y colisiones, tanto con obstáculos estáticos como dinámicos.

En primer lugar, se implementa un método de gestión de colisiones compuesto por dos soluciones tecnológicas, a modo de capas o subsistemas, que dotan al enjambre de una tecnología robusta y redundante para navegar a través de trayectorias prefijadas hasta diferentes ubicaciones u objetivos. La primera de las capas, a través de una arquitectura centralizada, se basa en localizar y establecer la posición relativa de cada uno de los UAVs del enjambre, con respecto al resto, en cada instante de tiempo, detectando posibles conflictos y colisiones entre los agentes del enjambre y, modificando los perfiles de velocidad de los UAVs en conflicto para evitar posibles colisiones. De esta forma, a partir de una planificación previa

de trayectorias, se puede desplegar una flota de UAVs para navegar, de manera autónoma y coordinada, en un área común de manera segura.

La segunda de las capas, trabajando de manera descentralizada, se basa en un sistema de cálculo de trayectorias alternativas para evasión de obstáculos a partir de la información 3D, recopilada por cada uno de los UAVs, procedente de sensores LiDAR embarcados en los propios UAVs. Este método, que complementa a la capa anterior, se basa en la información 3D para detectar obstáculos presentes en las rutas prefijadas y, poder llevar a cabo una nueva planificación que genere una ruta segura alternativa para evadir dichos obstáculos.

Por último, basado en los sistemas de alerta de tráfico y evasión de colisiones de la aviación comercial, se presenta un método basado en perímetros virtuales para evitar colisiones entre UAVs de un mismo enjambre. El desarrollo acometido permite establecer dos esferas de distintos tamaño alrededor de cada UAV, de tal manera que, si dos UAVs se aproximan a una distancia comprendida entre ambas esferas se genera una alarma para que, en caso de que un operador supervise la misión, pueda tomar acciones de control sobre algún UAV y, si no, en caso de que el movimiento autónomo de los UAVs reduzca la distancia por debajo del segundo umbral de seguridad, el sistema genere un maniobra evasiva en cada UAV, de tal manera que se resuelva el conflicto en el plano vertical, al variar la altitud de vuelo de los UAVs del enjambre implicados en el conflicto.

Con el conjunto de métodos desarrollados y validados en el presente capítulo se pretende aumentar la robustez y versatilidad de la arquitectura propuesta para permitir a un enjambre de UAVs completar, de manera autónoma y coordinada, una misión.

Capítulo 5

Sistema de Toma de Decisiones Inteligente para Navegación Autónoma de UAVs

En los capítulos anteriores se han presentado, analizado y validado un conjunto de métodos que constituyen una arquitectura global enfocada hacia la navegación autónoma y coordinada de un enjambre de UAVs sobre entornos dinámicos y, en ocasiones adversos, como en el caso de los incendios tanto forestales como urbanos. En el presente capítulo, se presenta y analiza una nueva capa, basada en AI, que pretende dotar a la arquitectura desarrollada de un módulo encargado de la toma de decisiones inteligentes para conseguir una navegación autónoma por parte de los UAVs que forman parte de un enjambre.

Los avances en el campo de la AI han provocado la expansión de estos métodos a una infinidad de líneas de investigación entre los que se encuentra su aplicación al campo de los UAVs. El desarrollo de algoritmos basados en AI y, en concreto, el avance en el campo del RL han provocado que la implementación de un método basado en esta tecnología pueda complementar a la arquitectura presentada hasta el momento y generar un modelo que permita, a cada UAV, navegar de manera autónoma por el entorno, gracias a la capacidad para tomar decisiones por sí mismo considerando la información capturada del entorno.

El desarrollo de esta nueva capa dentro de la arquitectura busca generar robustez y mejorar la eficiencia dentro de esta, estableciendo un algoritmo de toma de decisiones, basado en RL, que permita la navegación sin supervisión desde una ubicación a otra mediante la toma de decisiones inteligente, de tal forma que sea el propio UAV el que establezca que acción realizar en cada instante de tiempo, de cara a maximizar la recompensa obtenida a lo largo del recorrido. De esta manera, se introduce un tercer lazo de control dentro de la arquitectura que garantice la

culminación de una misión en entornos adversos.

El RL está considerado un área del aprendizaje automático caracterizado por la toma de datos de un entorno dinámico, con el objetivo de generar una secuencia de acciones que generen un resultado óptimo. En concreto, el RL consiste en aprender qué hacer o que situaciones asignar a un conjunto de acciones, para maximizar una señal numérica denominada recompensa, sin decir al agente que acción debe tomar, sino que es él mismo el que debe descubrir que acciones producen una mayor recompensa a través de la exploración del entorno (246). Esta característica, de aprender por sí mismo, mediante ensayo-error para determinar el mejor comportamiento, es lo que diferencia a este tipo de algoritmo del resto dentro del ML.

El conjunto de conceptos y fundamentos de RL necesarios para el análisis de los métodos propuestos y analizados en este capítulo son recogidos en el Apéndice A. El análisis de estos fundamentos permite establecer un punto de partida para detallar y explicar los algoritmos implementados que tratan de dotar, dentro de la arquitectura completa propuesta, a los UAVs de una AI que permita a cada agente tomar decisiones por sí mismo sobre qué acciones realizar sobre el entorno para alcanzar una ubicación determinada mediante el uso de algoritmos de RL. Aunque es cierto, que los algoritmos clásicos de RL se presentan como una solución eficaz a problemas de dimensiones bajas, muestran limitaciones a la hora de convertirse en métodos escalables capaces de generar soluciones a problemas de altas dimensiones (247). Para hacer frente al problema de la dimensionalidad han surgido, en los últimos años, una corriente investigadora encargada de la fusión de métodos basados en DL junto con métodos basados en RL, que han permitido el progreso de esta técnica en campos complejos en los que es necesario el tratamiento de grandes cantidades de información, mediante el uso de algoritmos de aprendizaje por refuerzo profundo (Deep Reinforcement Learning (DRL)).

5.1. Implementación de Métodos Basados en AI para Navegación Autónoma de UAVs

5.1.1. Aprendizaje por refuerzo profundo

El uso de algoritmos de DRL permite dar un paso hacia la creación de sistemas autónomos con un nivel de comprensión mayor del entorno en el que se encuentran, provocando que dichos algoritmos se puedan aplicar a problemas de tomas de decisiones en los que los espacios de acción y de observación presentan una alta dimensión.

El DRL presenta como idea principal que la capacidad de aprendizaje del agen-

te venga determinada por un modelo de DL o por una DNN. Por lo general, los algoritmos de DRL mantienen los mismos enfoques que los empleados en los algoritmos de RL con la novedad de que introducen modelos de DL para que los agentes aprendan las políticas. Así pues, algoritmos como el DQN, la optimización de políticas de la región de confianza (Trust Region Policy Optimization (TRPO)) y el método actor crítico de ventajas asíncronas (Asynchronous Advantage Actor Critic (A3C)) se establecen como el conjunto central de métodos dentro del DRL. El uso de CNNs como componente dentro de un agente permite aprender políticas a partir de entradas en crudo y de alta dimensión como pueden ser las imágenes.

En general, los algoritmos de DRL se basan en el entrenamiento de ANNs para aproximarse a políticas óptimas (π^*) o a funciones de valor óptimas (Q^* o V^*). La capacidad de las ANNs de aproximar funciones ha provocado que la combinación de estos dos métodos se oriente hacia la regresión de funciones para su uso en agentes de RL, favoreciendo el uso explícito de funciones de valor, capaces de capturar la estructura que subyace al entorno.

5.1.2. Algoritmo Deep Q-Network

El algoritmo DQN (248) constituye uno de los métodos más empleado y estudiado dentro del DRL y, se caracteriza por la combinación de una red neuronal convolucional profunda con el método de RL Q-Learning. Surgió como una extensión del Neural Fitted Q-Learning (NFQ) (249), tratando de solucionar problemas de estabilidad y divergencia ocasionados al combinar algoritmos de RL sin modelos con ANN. El motivo de estos problemas nace, en primer lugar, de la correlación entre los estados posteriores en las tareas de RL y, en segundo lugar, de los cambios frecuentes en la política como consecuencia de ligeras modificaciones en los valores de Q (250).

Para hacer frente a dicha problemática, el algoritmo DQN incluye un enfoque denominado repetición de la experiencia, en el cual, se almacena experiencia del agente (s_t, a_t, r_t, r_{t+1}) la cual se emplea a la hora de realizar actualizaciones de los valores de Q , introduciendo un muestreo aleatorio uniforme de dicha memoria de repetición de la experiencia durante el entrenamiento, eliminando, en cierto modo, las correlaciones entre muestras consecutivas. Para el segundo de los problemas, la red se entrena con el objetivo de obtener objetivos de Q-Learning consistentes que fijen los pesos empleados y se actualicen periódicamente.

El uso de este método tanto en planificación de trayectorias como en la navegación autónoma de vehículos como los UAVs es cada vez más extendido debido, principalmente, a la posibilidad de generar un modelo que permita, a un UAV, alcanzar un objetivo concreto navegando sin supervisión por entornos desconocidos, de los cuales no se posee información previa y, en los que mediante información

recopilada del entorno y del estado de la aeronave, se pueda llevar a cabo una toma de decisión acertada. De esta forma, gracias al conocimiento adquirido por el UAV a través de métodos de DRL como DQN, se podrán acometer tareas complejas en entornos dinámicos como son los incendios forestales y urbanos. Este aprendizaje complejo se hace posible al poder introducir a las ANNs un conjunto de observaciones de altas dimensiones que permitan al agente aprender aquella política que escoja, a cada momento, aquella acción que maximice el valor de retorno, es decir, el conjunto de recompensas al realizar una secuencia de acciones sobre el entorno.

5.1.3. Análisis e implementación de algoritmo de DQN para navegación autónoma de UAVs

Por tanto, la idea principal del desarrollo propuesto en el presente capítulo se centra en poder incluir, dentro de la arquitectura de navegación autónoma, una nueva capa que dote a los sistemas de una AI suficientemente avanzada como para que cada UAV elija una secuencia de acciones que les permita navegar de un punto de ruta a otro evitando colisionar con objetos presentes en el entorno mediante la toma de decisiones autónoma.

A lo largo de la sección se muestran los desarrollos realizados en esta área y los resultados alcanzados, comparando los mismos en función de los parámetros de entrenamiento empleados, el conjunto de estados, el conjunto de observaciones y el entorno empleado.

Tanto para el entrenamiento del algoritmo, como para la validación de los resultados, se ha empleado una arquitectura de software basada en TensorFlow (251) y Keras (252) e integrando ROS y *Gazebo*, de tal manera que permita realizar el entrenamiento sobre un entorno simulado que facilite, en el futuro, el paso de la implementación y validación a plataformas aéreas reales.

De cara a poder establecer un modelo de DRL lo más versátil y completo posible que permita a un UAV navegar, de un punto de ruta a otro por un entorno desconocido, se han llevado a cabo diferentes entrenamientos en los que se han modificado aspectos como la estructura de la ANN, el espacio de acciones o parámetros de entrenamiento y aprendizaje.

Lo que se ha mantenido constante durante todo el desarrollo son las entradas del sistema, o lo que en DRL se conoce como espacio de observación y que, en este caso, se encuentra ligado a la información procedente del UAV y, que permite conocer el estado del mismo dentro del entorno. El número de observaciones son 7 y se corresponden con la posición x , y , z del UAV, su orientación en los ángulos de *alabeo*, *cabeceo* y *guiñada* y, por último, el valor de distancia a obstáculos obtenido mediante el uso de un sensor tipo sonar. Este conjunto de parámetros constituye la entrada a la ANN y permite conocer el estado del UAV en cada paso, pudiendo

saber si se está aproximando o alejando del objetivo, si la estabilidad del sistema es la correcta, si la acción tomada ha provocado que el UAV adquiera un estado físico comprometido o, por último, conocer si se aproxima o se aleja de un obstáculo presente en el entorno.

Cómo se ha indicado, el espacio de observaciones constituye la entrada al modelo de red neuronal empleado durante el entrenamiento y que, con modificaciones en el número de capas y dimensiones del espacio de salida de cada una de ellas, presenta una estructura general similar en todos los casos, tal y como se muestra en la Figura 5.1, en el sentido de que la entrada a la red siempre será el espacio de observaciones y la salida de la misma estará caracterizada por el espacio de acción empleado en dicho momento.

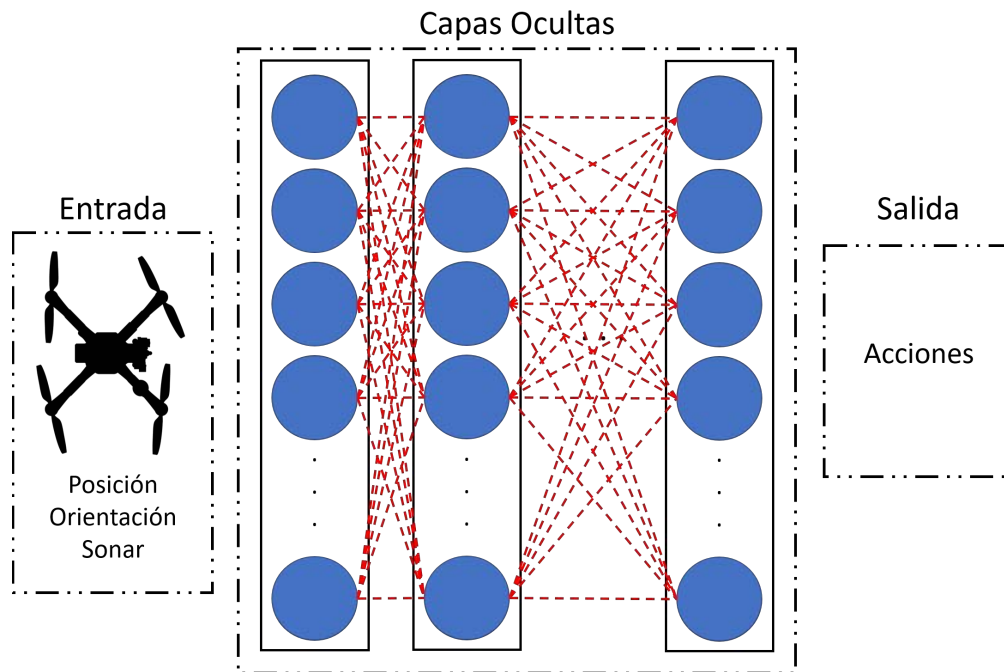


Figura 5.1: Estructura general de la ANN empleada en el algoritmo DQN.

A continuación, se analizan aquellos casos en los cuales se ha obtenido un modelo capaz de navegar por el entorno, sin supervisión, desde una ubicación determinada a otra, alcanzando el objetivo previamente fijado.

5.1.4. Modelo 1 de toma de decisiones inteligente

Este primer modelo genera una solución para la navegación autónoma de un UAV hacia una ubicación determinada en un escenario con presencia de obstáculos estáticos.

A la hora de elegir la estructura de la red neuronal, para el aprendizaje de la política a seguir para maximizar la recompensa por parte del agente, es importante establecer una red lo suficientemente compleja como para poder aproximarse a la función buscada, pero no tan compleja que imposibilite el entrenamiento. Por este motivo, y debido a que durante este desarrollo no se va a analizar información visual procedente del entorno, lo cual haría de las CNNs una estructura ideal para la red del agente se ha decidido comenzar la investigación haciendo uso de las redes multicapa más generales como son las redes completamente conectadas, en las cuales, cada neurona de una capa está conectada a todas las neuronas de la siguiente. En este modelo concreto se ha empleado una estructura como la que se recoge en la Tabla 5.1, caracterizada por una estructura simétrica de 5 capas ocultas completamente conectadas.

Tabla 5.1: Estructura de la ANN para el modelo número 1 de DRL.

Capa (Tipo)	Forma de Salida	Parámetros #
dense_1 (Dense)	(None, 24)	192
dense_2 (Dense)	(None, 48)	1200
dense_3 (Dense)	(None, 64)	3136
dense_4 (Dense)	(None, 48)	3120
dense_5 (Dense)	(None, 24)	1176
dense_6 (Dense)	(None, 6)	150

La última capa se corresponde a la salida de la red y se parametriza en función del número de acciones que caracterizan al agente y que, en este caso, se limita a 6. En concreto, el agente puede elegir entre el siguiente conjunto de acciones: movimiento positivo en el eje X , movimiento negativo en el eje X , movimiento positivo en el eje Y , movimiento negativo en el eje Y , movimiento positivo en el eje Z y, por último, movimiento negativo en el eje Z .

Dentro de la estructura de la red neuronal se hace necesario el uso de optimizadores que ayuden a la convergencia de las mismas hacia un mínimo local. En este trabajo se emplea el optimizador de Estimación del Momento Adaptativo (ADaptative Moment estimation (ADAM)) (253), el cual combina las fortalezas del algoritmo del Gradiente Adaptativo (Adaptative Gradient Algorithm (Ada-Grad)) y del algoritmo de Propagación del Valor cuadrático medio (Root Mean Square Propagation (RMSProp)) y, propone como novedad el momento, o lo que es lo mismo, la aceleración del descenso por gradiente. Es un optimizador altamente empleado hoy en día debido a su bajo uso de memoria y a su efectividad computacional y que, cuenta con la tasa de aprendizaje como parámetro esencial en el funcionamiento del mismo. La tasa de aprendizaje consiste en un hiperparámetro que modifica el desempeño de la red y que debe ser ajustado para evitar que el

aprendizaje se demore en exceso, en casos donde la tasa de aprendizaje sea baja, o sea errático, cuando la tasa de aprendizaje sea muy alta.

Un elemento fundamental dentro de los algoritmos de RL y DRL es la recompensa, es decir, el valor numérico que recibe el agente en función de cómo de buena o mala es la acción ejecutada sobre el entorno. En este caso, se establece el siguiente esquema de recompensas:

- **Para cada iteración con el entorno:** Después de ejecutarse cada acción, el algoritmo comprueba si la distancia al objetivo se ha reducido con dicha acción o si ha aumentado. En caso de que la acción reduzca la distancia al objetivo se genera una recompensa positiva de 10, mientras que si se aleja del destino la recompensa es negativa de valor similar al anterior. La idea de emplear este sistema de recompensas es evitar las oscilaciones del agente en el entorno, ya que sin recompensa negativa o, con una recompensa negativa de menor magnitud que la positiva, el agente tendía a detectar un obstáculo y comenzar a realizar movimientos repetitivos hacia delante y hacia atrás obteniendo una recompensa acumulada mayor que para casos en los que el UAV es capaz de navegar de manera directa hacia el objetivo.
- **Para el final de cada época:** Al finalizar cada época, bien porque el agente haya alcanzado el objetivo, porque se haya dado un número máximo de iteraciones o porque se origine alguna situación contemplada para que esto suceda, se establece una recompensa que varía en función de si el objetivo se ha alcanzado o no y, en función de la distancia al mismo. De tal forma que, si el destino es alcanzado, se genera una recompensa que sigue la Ecuación 5.1, mientras que si el objetivo no es alcanzado la recompensa generada sigue la Ecuación 5.2. De esta manera se premia, tanto si el objetivo ha sido alcanzado como si no, la proximidad a la ubicación de destino. Esta solución permite al agente tener un conocimiento de cómo de buena ha sido esta secuencia de acciones con respecto a las anteriores, incluso aunque el objetivo no sea alcanzado, ya que, al introducir el parámetro de la distancia al objetivo, aunque el entrenamiento no sea satisfactorio en esa época, el agente puede conocer si es mejor que otras secuencias donde tampoco ha sido alcanzado el punto de ruta en cuestión y, de esta manera mejorar su conocimiento y aprendizaje.

$$\gamma = 200 - 0,1 \cdot d \quad (5.1)$$

$$\gamma = -200 - 0,1 \cdot d \quad (5.2)$$

Dónde γ corresponde al valor de recompensa y d a la distancia al objetivo.

Con el objetivo de mejorar el aprendizaje y acelerar la convergencia del entrenamiento, se han establecido una serie de casos para los cuáles se da por finalizada la época, se reinicia el entorno y se comienza de nuevo. Estos casos vienen determinados por el espacio de observación y la interpretación de los mismos por parte del algoritmo.

Así pues, se da por finalizado la época en el caso de que le UAV haya colisionado con un obstáculo o se encuentre en una situación incompatible con la estabilidad de vuelo del sistema. Para ello, se comprueba, en primer lugar, el valor del sensor sonar embarcado, de tal forma que si dicho valor se encuentra por debajo de 0.5 *metros* se da por hecho que la aeronave va a colisionar con un obstáculo del entorno. En segundo lugar, se comprueba los ángulos de *alabeo* y *cabeceo* y, si se encuentran por encima de los 90° se considera que el UAV ha alcanzado una posición incompatible con la estabilidad y vuelo del sistema. Por último, de cara a acelerar el proceso de aprendizaje, durante el entrenamiento se establece un espacio de trabajo que delimita un área por el cual se puede mover el agente, finalizando la época y comenzando una nueva en el caso de que una acción conduzca al UAV a una ubicación fuera del área delimitada.

Según se observa en la Figura 5.2, se puede ver como a medida que avanza el entrenamiento el aprendizaje de la red mejora y, el modelo de toma de decisiones generado se va acercando a una política que consigue maximizar el valor retornado, siendo muestra de que la estructura de DQN empleada en este modelo es satisfactoria.

La Figura 5.2 muestra la recompensa total obtenida al final de cada época del entrenamiento y, en ella se puede observar la esencia de los algoritmos de DRL basados en el aprendizaje de ensayo-error. En las primeras épocas, cuando la tasa de exploración es mayor, el agente establece secuencias de acciones que no son satisfactorias acumulando, al final de cada época, una recompensa total negativa. A medida que el entrenamiento avanza y, el agente va, en primer lugar, aprendiendo cuales son las mejores acciones a tomar y, en segundo lugar, se va reduciendo la tasa de exploración para dar prioridad a la explotación de las mejores acciones, se puede observar como la recompensa acumulada al final de cada época va siendo más positiva, aumentando con el entrenamiento y alcanzando una cierta estabilidad entorno al valor máximo de recompensa al final del entrenamiento.

Este análisis se completa con la Figura 5.3, la cual muestra la recompensa total acumulada a medida que avanza el entrenamiento. En ella, se puede observar cómo al comienzo del entrenamiento, la falta de conocimiento y aprendizaje y, la configuración del entrenamiento para realizar una alta exploración al comienzo en busca de la secuencia de acciones que maximice el valor retornado, provoca que la recompensa acumulada vaya decayendo hacia valores negativos en la primera

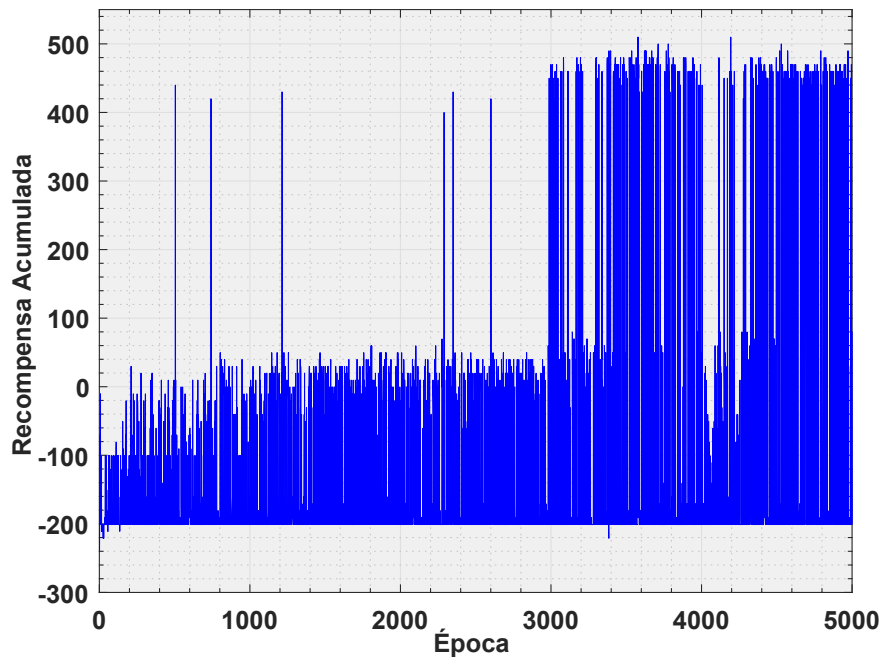


Figura 5.2: Recompensa acumulada por época de entrenamiento para el modelo 1 de toma de decisiones.

mitad del entrenamiento. A medida que el agente va ganando aprendizaje, el valor retornado va mejorando y aumentando, y el abandono de la exploración en detrimento de la explotación provoca que la pendiente de la curva suavice se establezca la recompensa acumulada. El resultado final de la curva permite concluir que, aunque el resultado final obtenido es satisfactorio en cuanto a la navegación autónoma del UAV, serían necesarias un conjunto de épocas mayor de cara a invertir la pendiente de la curva, establecer un crecimiento de la recompensa total acumulada y terminar estabilizando dicha curva en valores de recompensa positivos.

Es importante destacar que, tal y como se ha detallado previamente en este capítulo, el algoritmo DQN se caracteriza por emplear un enfoque de repetición de la experiencia para eliminar, de esta manera, problemas relacionados con las correlaciones entre muestras consecutivas. Por este motivo, el presente modelo se encuentra configurado para que se puedan almacenar en memoria un máximo de 64 lotes, que permitan la actualización de los valores de Q a partir de un muestreo aleatorio de dicha memoria de repetición.

Junto a este, se establece otro conjunto de parámetros que buscan hacer frente al problema de la exploración-explotación. Por tanto, se incluye una variable que permite controlar la exploración, de tal forma que al comienzo del entrenamiento favorezca la misma, pero que dicha constante vaya decayendo a medida que au-

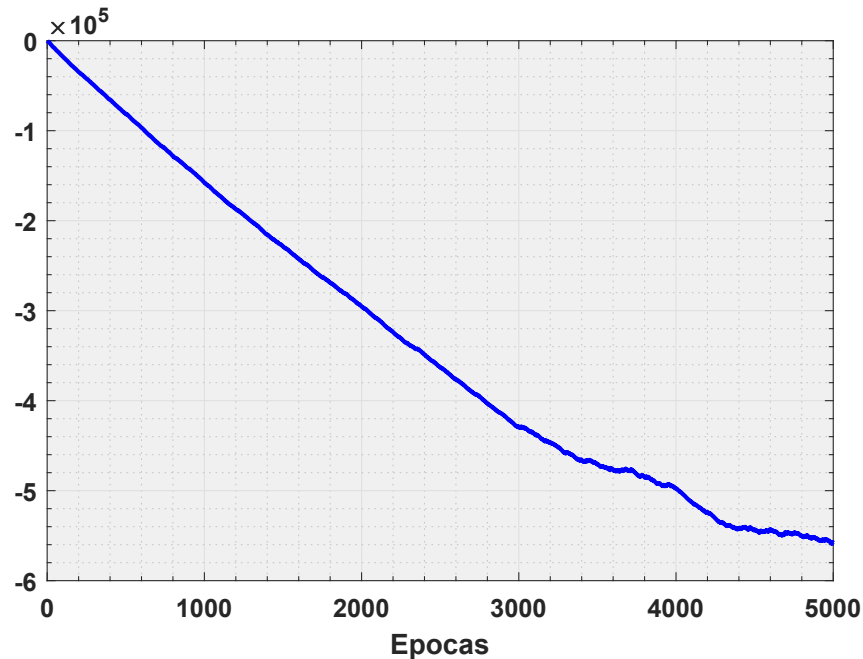


Figura 5.3: Recompensa total acumulada por el modelo 1 de toma de decisiones inteligente a lo largo del entrenamiento.

mentan las épocas, favoreciendo de esta forma la explotación a medida que avanza el entrenamiento.

A partir de este entrenamiento, se obtiene un modelo que permite a un UAV navegar, sin supervisión, hacia una ubicación de destino sobre un entorno similar al explorado.

5.1.5. Modelo 2 de toma de decisiones inteligente

Este segundo modelo genera una solución para la navegación autónoma de un UAV hacia una ubicación determinada en un escenario con presencia de obstáculos estáticos idéntico al escogido para el primer modelo de aprendizaje. El objetivo de este segundo modelo es continuar profundizando en los algoritmos de DRL y tratar de dotar al UAV de un comportamiento más similar al vuelo real.

Dado que, en el caso anterior, la estructura de la red neuronal consiguió resultados satisfactorios y puesto que, el espacio de observaciones se mantiene inamovible, se ha optado en este segundo modelo por emplear una estructura basada en redes multicapa como en el caso anterior. Para este modelo en concreto, se ha optado de nuevo por una red simétrica, tal y como se muestra en la Tabla 5.2, completamente conectada, pero modificando la estructura de cada una de las capas ocultas,

variando el número de neuronas debido a que, en este caso, se emplea un espacio de acción diferente, con el cual la estructura de red neuronal anterior no era capaz de obtener un aprendizaje correcto.

Tabla 5.2: Estructura de la ANN para el modelo de toma de decisiones número 2 de DRL.

Capa (Tipo)	Forma de Salida	Parámetros #
dense_1 (Dense)	(None, 48)	384
dense_2 (Dense)	(None, 64)	3136
dense_3 (Dense)	(None, 128)	8320
dense_4 (Dense)	(None, 64)	8256
dense_5 (Dense)	(None, 48)	3120
dense_6 (Dense)	(None, 6)	294

De nuevo, la última capa se corresponde a la salida de la red y se parametriza en función del número de acciones que caracterizan al agente y que, en este caso, se limita a 6. En concreto, para este segundo modelo y, de cara a obtener un comportamiento más real, el agente puede elegir entre el siguiente conjunto de acciones: movimiento positivo en el eje X , movimiento negativo en el eje X , giro positivo en el eje Z , giro negativo en el eje Z , movimiento positivo en el eje Z y, por último, movimiento negativo en el eje Z . De esta forma, con este nuevo espacio de acción, el UAV es capaz de realizar movimientos de ascenso y descenso, ir hacia delante o hacia atrás o, por último, realizar giros a lo largo del eje Z o guiñada, lo que permite establecer un control, no solo de posición, sino de orientación del vehículo.

En cuanto a las recompensas, es importante introducir cambios con respecto al anterior modelo para poder considerar aspectos nuevos introducidos como es el caso de la orientación. Por este motivo, en este modelo se ha establecido el siguiente esquema de recompensas:

- Para cada iteración con el entorno:** Después de cada acción el algoritmo implementado se encarga de comprobar tanto la nueva distancia al objetivo como la nueva orientación alcanzada por el UAV. Con dicha orientación se establece una comparación con la orientación previa y la orientación de destino, de tal manera que, si el agente ha establecido una acción de giro sobre el eje Z pueda tener una retroalimentación sobre como de buena o mala ha sido dicha acción. La orientación de destino se considera al ángulo α que debe girar el UAV a partir de su orientación actual para alinear su eje X con el destino, tal y como se muestra en la Figura 5.4.

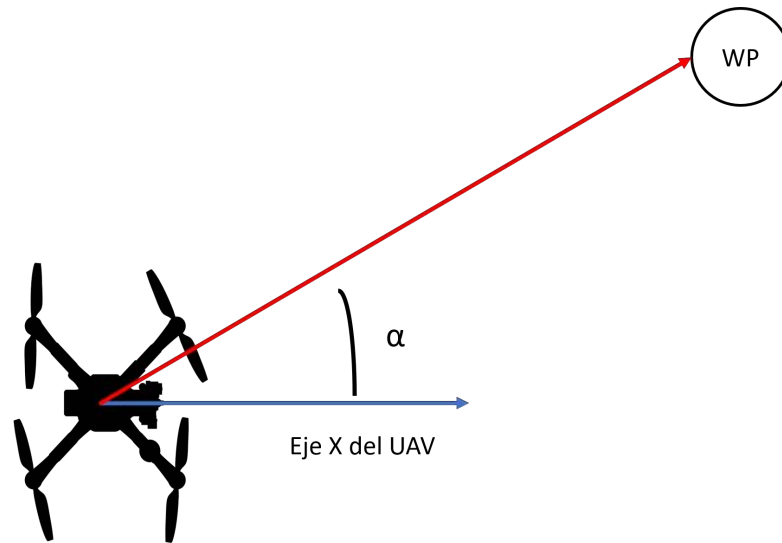


Figura 5.4: Esquema de cálculo de orientación para permitir la navegación autónoma con el UAV enfocado hacia el objetivo a alcanzar.

Con estas consideraciones, la estructura de recompensas para cada acción es la siguiente:

- Si la distancia al objetivo decrece y el ángulo de orientación se acerca al deseado se obtiene una recompensa de valor 20.
 - Si la distancia decrece, pero el giro del UAV le ha alejado de la orientación deseada, el valor de recompensa se sitúa en 10.
 - Si el valor de distancia al objetivo aumenta, la recompensa obtenida se establece en un valor de -30 . Dicho valor se establece buscando oscilaciones del agente entorno a una ubicación acumulando recompensas sin estar realizando una secuencia de acciones óptima.
- **Para el final de cada época:** Al finalizar cada época, se sigue la estructura de recompensa empleada en el modelo anterior, la cual varía en función de la distancia al objetivo, siendo positiva si se ha alcanzado el mismo o negativa en caso contrario.

En este caso, la Figura 5.5 recoge la recompensa acumulada al finalizar cada época a lo largo del entrenamiento. Se puede observar cómo en este caso, se produce una convergencia más rápida, reduciéndose el número de épocas empleadas durante el entrenamiento. Esto se debe a dos motivos, el primero, el entrenamiento se encuentra configurado para que, si la media de la recompensa acumulada se

encuentra por encima de un umbral el entrenamiento se dé por terminado, considerando que el modelo ha adquirido el aprendizaje necesario para alcanzar el objetivo. El segundo, porque en este caso se ha usado un modelo previamente entrenado cuyos resultados no eran completamente válidos, pero poseía un aprendizaje que se aproximaba al comportamiento deseado.

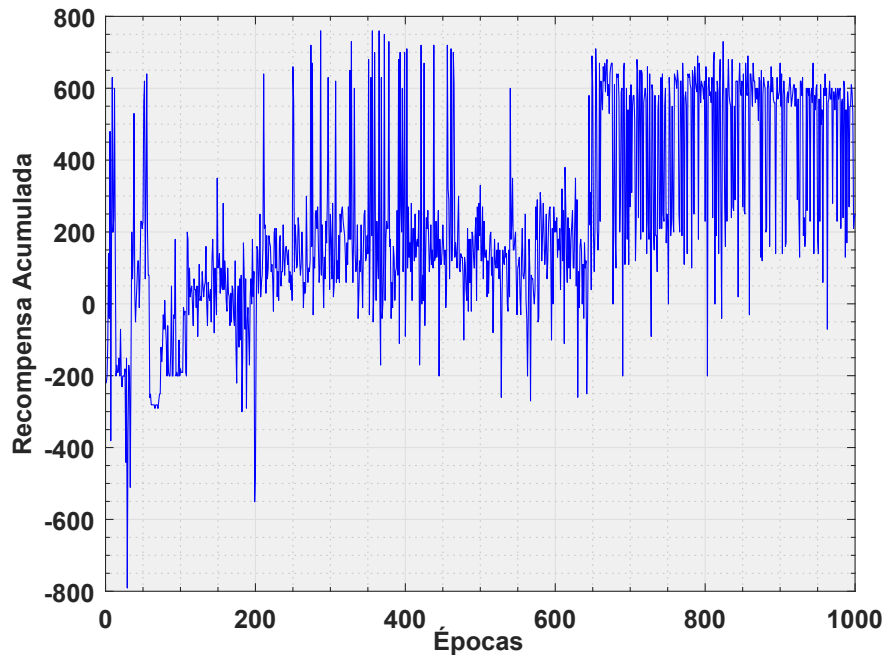


Figura 5.5: Recompensa acumulada por época de entrenamiento para el modelo 2 de toma de decisiones.

De nuevo, si se analiza la Figura 5.5 se observa como al comienzo del entrenamiento las oscilaciones son mayores debido a que la tasa de exploración es mayor y, aunque en este caso el modelo parte de un aprendizaje previo, en ocasiones el exceso de exploración conduce a una secuencia de acciones en las cuales el resultado no es satisfactorio, obteniéndose un valor de retorno negativo. A medida que aumenta el entrenamiento, la disminución de la exploración en favor de una mayor explotación provoca que el agente encadene épocas con resultado satisfactorio, obteniendo de manera más constante un valor de retorno máximo. Aun así, la exploración no se reduce a 0, por lo que incluso al final del entrenamiento aparecen épocas en las que la secuencia de acciones establecida no es correcta.

Al igual que en el modelo anterior, el análisis del desarrollo e implementación realizada se completa con la Figura 5.6. En ella, se muestra la recompensa acumulada a lo largo del entrenamiento y, en este caso sí, se puede observar como el perfil de la gráfica se adapta al resultado óptimo esperado de los algoritmos de

DRL, es decir, comienza con una recompensa acumulada negativa fruto de la falta de aprendizaje y de la fase de exploración y, a medida que el entrenamiento va avanzando, el modelo va adquiriendo mayor conocimiento, va abandonando la exploración en detrimento de la explotación y la recompensa acumulada va creciendo progresivamente hasta situarse en valores de retorno máximo.

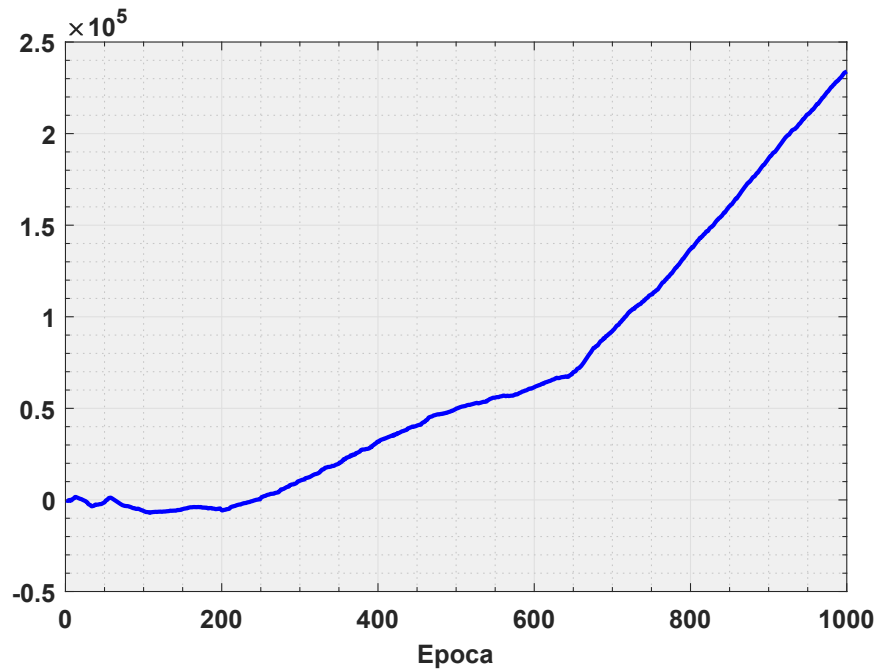


Figura 5.6: Recompensa total acumulada por el modelo 2 de toma de decisiones inteligente a lo largo del entrenamiento.

La idea de emplear estos modelos que previamente han sido entrenados es reducir, tal y como se refleja en la Figura 5.5, el tiempo que tarda la red en converger hacia una política óptima en la que se maximice la recompensa acumulada. Basándose en la estructura de red establecida en el modelo anterior, el algoritmo carga los parámetros de entrenamiento y comienza el proceso iterativo tratando de alcanzar una política óptima que permita al agente completar de manera satisfactoria el objetivo marcado.

Si se analiza los resultados del modelo entrenado previamente, se puede observar como la modificación de la estructura de la ANN aumentando el número de nodos completamente conectados en cada una de las capas permite una convergencia más rápida en la fase de aprendizaje. En este caso, el modelo consigue encadenar épocas con una recompensa acumulada por época alta, generando un modelo que es capaz de navegar hacia la zona de interés prefijada, pero con una precisión mejorable, de ahí que se empleara este modelo para realizar un nuevo

entrenamiento y, tratar de mejorar la precisión del mismo. Si se suman ambos procesos de entrenamiento, se establece que son necesarias 1.749 épocas para obtener un modelo capaz de navegar entre dos ubicaciones concretas de manera completamente autónoma.

El modelo previamente entrenado era capaz de realizar una navegación sin supervisión, pero la menor distancia conseguida al destino se situaba en los 1,98 *metros*, mientras que para este modelo en cuestión se consigue una aproximación a la ubicación de interés inferior a los 0,8 *metros*.

Si se compara este modelo de aprendizaje, en el que el estado de acción ha sido configurado para generar un comportamiento más realista desde el punto de vista de la estabilidad y dinámica de vuelo, con el modelo de aprendizaje analizado en la Sección 5.1.4, se puede observar que para este nuevo modelo, el agente es capaz de alcanzar el objetivo con una mayor precisión para una distancia total recorrida similar. Así pues, el modelo 1 de toma de decisiones inteligente recorre 11,71 *metros* navegando por el entorno hasta alcanzar el objetivo con una precisión de 0,99 *metros*, mientras que el modelo 2 es capaz de alcanzar el destino con una precisión de 0,70 *metros* tras haber recorrido 11,85 *metros*.

5.1.6. Modelo 3 de toma de decisiones inteligente

Este tercer modelo genera una nueva solución para la navegación autónoma de un UAV entre dos ubicaciones concretas en un escenario con presencia de obstáculos estáticos, similar al empleado en los modelos de aprendizaje anteriores. El objetivo de este tercer modelo es continuar profundizando en los algoritmos de DRL y tratar de mejorar la precisión a la hora de alcanzar una ubicación determinada gracias a la introducción de restricciones a la hora de realizar acciones durante el entrenamiento.

Para lograr mejorar la precisión con la que el UAV navega hacia una ubicación prefijada se ha establecido un espacio de trabajo delimitado en forma de triángulo, con vértice en dicha ubicación, de tal manera que, cuanto más se aproxime al destino menos capacidad de movimiento en el plano XY tiene el agente. Para detallar y comprender mejor esta restricción de la movilidad del UAV al aproximarse hacia el objetivo, se ha incluido la Figura 5.7, la cual muestra un esquema de cómo se lleva a cabo la acotación en el plano XY del área de trabajo disponible.

De esta forma, en función de un radio r , se establece un triángulo sobre el plano XY sobre el cual el UAV se puede mover en busca de una trayectoria óptima hacia al objetivo establecido (WP_1). Este espacio (área sombreada en amarillo) se ha acotado a lo largo del plano XY y, no se ha extrapolado a la tercera dimensión, como un volumen en forma de cono, para permitir al agente esquivar obstáculos mediante movimientos de subida y bajada, en el caso de que no hubiera espacio

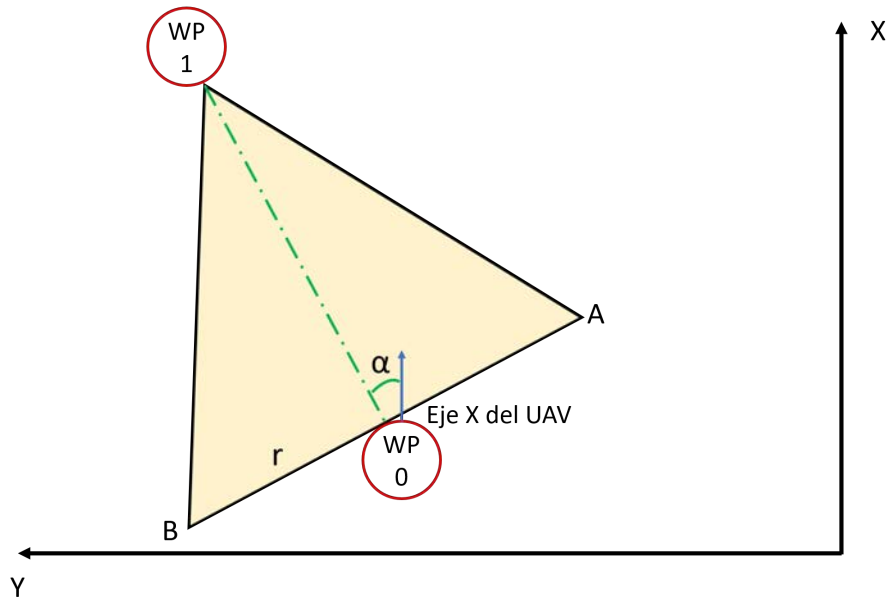


Figura 5.7: Limitación del área de exploración para navegación precisa hacia una ubicación prefijada.

suficiente en el plano XY para llevar a cabo maniobras de evasión.

Para su implementación dentro del algoritmo de DRL se establece, tras la realización de cada acción, un proceso de comprobación para establecer si el agente continúa moviéndose dentro del espacio escogido para ello o si, por el contrario, la acción tomada le ha llevado a salir fuera del mismo, motivo por el cual se reiniciará el entorno, dando por finalizada la época, acumulando la recompensa correspondiente y comenzando una nueva época.

Desde el punto de vista de desarrollo, se implementa un método encargado de generar un polígono regular con vértice en el punto de destino, en el punto A y en el punto B y, en cada iteración se toma, del espacio de estados, la posición actual del agente y se comprueba si dicha ubicación se encuentra dentro o fuera del triángulo. Para el cálculo de los puntos A y B se hace uso de conceptos de trigonometría y, a partir del ángulo α , del punto de partida del UAV (WP_0) y del radio preestablecido (r) se calculan las coordenadas X e Y haciendo uso de las Ecuaciones 5.3 y 5.4.

$$X_A = -X_B = X_{WP_0} + r \cdot \sin(\alpha) \quad (5.3)$$

$$Y_A = -Y_B = -(Y_{WP_0} + r \cdot \cos(\alpha)) \quad (5.4)$$

En este tercer modelo se mantiene la estructura de ANN que permite obtener resultados satisfactorios en los modelos anteriores, empleando redes multicapa

completamente conectadas. Para este modelo en concreto, se ha optado, de nuevo, por una red simétrica, tal y como se muestra en la Tabla 5.3, pero modificando la estructura al eliminar una de las capas ocultas, tratando de comprobar si para este caso, una red con un menor número de nodos conectados es capaz de aprender un comportamiento como el que se pretende para el agente.

Tabla 5.3: Estructura de la ANN para el modelo 3 de DRL.

Capa (Tipo)	Forma de Salida	Parámetros #
dense_1 (Dense)	(None, 48)	384
dense_2 (Dense)	(None, 64)	3136
dense_3 (Dense)	(None, 64)	4160
dense_4 (Dense)	(None, 48)	3120
dense_5 (Dense)	(None, 6)	294

En cuanto a las recompensas, dado que la única variación con respecto al modelo de aprendizaje anterior es la modificación del área de trabajo para conseguir una mayor precisión a la hora de acercarse a una ubicación determinada, se ha optado por mantener el mismo sistema de recompensas que para el modelo 2 de toma de decisiones inteligente.

En la Figura 5.8 se muestra la evolución de la recompensa acumulada al finalizar cada época por el agente a lo largo del entrenamiento. En este caso, de nuevo se ha empleado un modelo previamente entrenado, el cual no había conseguido un resultado completamente satisfactorio al finalizar el entrenamiento, pero cuyo modelo de aprendizaje generado se aproximaba al comportamiento deseado. Se puede observar cómo, partiendo de un modelo previamente entrenado, son tan solo necesarias 1925 épocas para lograr la convergencia de la ANN hacia una política óptima que maximice la recompensa total acumulada por el agente.

En este caso, la Figura 5.8 incluye los resultados del modelo previamente entrenado de ahí que, al comienzo de toda la gráfica los resultados obtenidos sean negativos, el entrenamiento mejore y, en mitad del entrenamiento vuelvan a aparecer valores negativos. Estos espacios negativos que aparecen a lo largo de las épocas 1000 y 1500 son debidos a que el entrenamiento del modelo comienza a partir del aprendizaje anterior, por lo que la fase de exploración vuelve a ser alta y, a pesar de poseer un aprendizaje previo, se intercalan épocas con una secuencia de acciones satisfactorias, con épocas de exploración dónde el resultado final se aleja del objetivo a alcanzar.

Junto a este análisis, se incluye el avance de la recompensa total adquirida a lo largo del entrenamiento, tal y como se refleja en la Figura 5.9. De nuevo, se puede observar como la respuesta obtenida se aproxima al resultado esperado y, es que tras un conjunto de épocas en las que el agente explora y tratar de encontrar

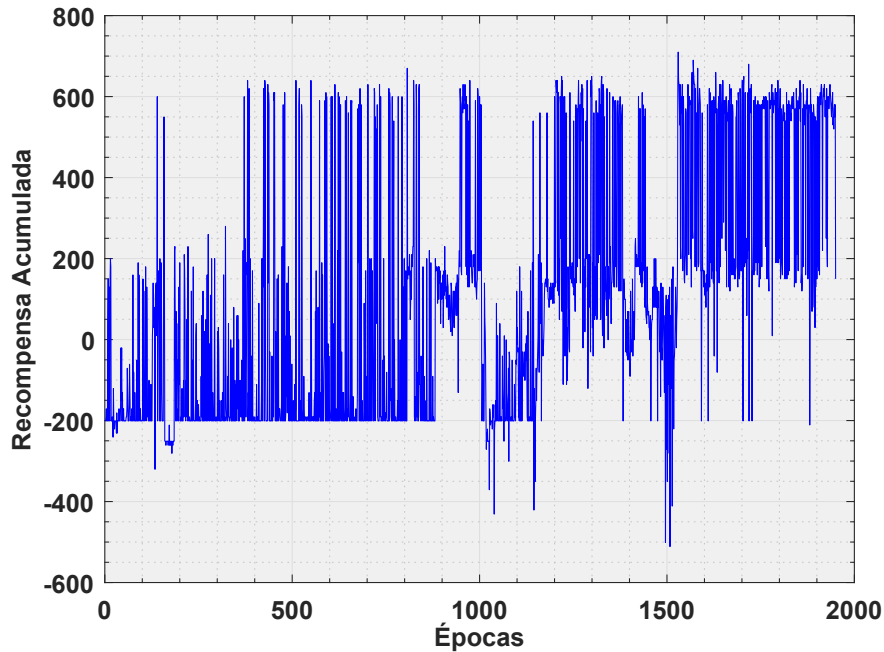


Figura 5.8: Recompensa acumulada por época de entrenamiento para el modelo 3 de toma de decisiones.

nuevas alternativas óptimas, finalmente se alcanza la fase de explotación en la que el agente va escogiendo aquellas acciones que retorna un valor de recompensa positivo y, se va acumulando el valor de retorno obteniendo máximos hacia el final del entrenamiento.

De nuevo, es importante establecer una comparación sobre como las modificaciones introducidas en cuanto a la restricción de movimiento del UAV dentro del entorno, provocan o no, una mejora considerable en cuanto a la precisión con la que se alcanza el punto de destino. La Tabla 5.4 muestra una comparativa del uso de los modelos generados sobre un mismo entorno para alcanzar un mismo destino. Dicha tabla muestra, por un lado, la distancia recorrida en cada acción y como varía la distancia al objetivo que se pretende alcanzar. Si se observa la última fila de cada modelo, se puede observar como para todos los casos la distancia total recorrida es muy similar, oscilando entre los 11,71 *metros* y los 11,85 *metros*, mientras que, en el caso de la precisión, es el último modelo, gracias al espacio de trabajo diseñado, el que consigue una reducción de un 34% en la reducción de la precisión, fijando dicho valor en 0,65 *metros*.

Por tanto, este modelo de aprendizaje permite obtener una solución para la navegación autónoma de un UAV mejorando la precisión con la que se alcanzan las ubicaciones de destino gracias a una limitación de movimientos en el plano

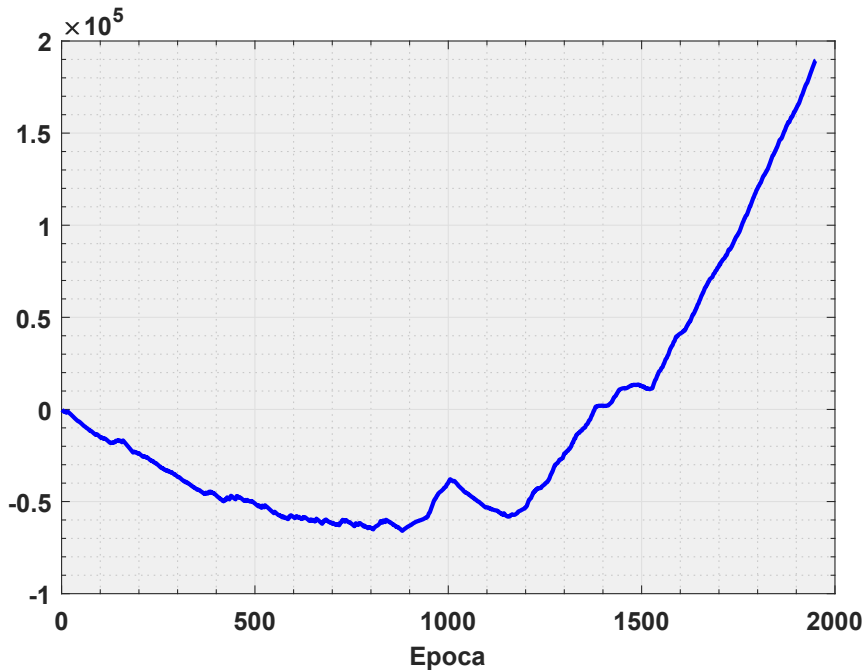


Figura 5.9: Recompensa total acumulada por el modelo 3 de toma de decisiones inteligente a lo largo del entrenamiento.

XY , que restringe el área libre a medida que el agente se aproxima al objetivo.

5.1.7. Modelo 4 de toma de decisiones inteligente

Hasta este momento, todos los modelos de aprendizaje analizados han trabajado sobre un entorno estático, con una baja densidad de obstáculos, pero han permitido generar modelos dinámicos de UAV basados en DRL capaces de introducir una capa de toma de decisión autónoma dentro de la arquitectura presentada que aumente la robustez del sistema de cara a navegar hacia una ubicación concreta, sin supervisión.

De cara a poder conocer como los desarrollos basados en DRL permiten introducir en la arquitectura una nueva capa capaz de complementar a los métodos de planificación vistos en el Capítulo 3, se ha establecido un modelo capaz de navegar de manera autónoma por un entorno estructurado con presencia de obstáculos dinámicos que se corresponde con uno de los entornos empleados para la validación de los métodos de planificación de trayectorias y navegación autónoma, visto en los capítulos previos, al que se ha añadido la presencia de obstáculos dinámicos. La idea de incluir este tipo de obstáculos dentro del entorno es para poder establecer un modelo que posea la capacidad de realizar tareas en entornos cambiantes

Tabla 5.4: Comparativa entre modelos de aprendizaje.

Modelo de Aprendizaje 1		Modelo de Aprendizaje 2		Modelo de Aprendizaje 3	
Distancia Recorrida	Distancia al Objetivo	Distancia Recorrida	Distancia al Objetivo	Distancia Recorrida	Distancia al Objetivo
0.66	9.91	1.00	9.92	1.00	9.92
1.07	9.58	1.37	9.76	1.36	9.76
1.55	9.16	1.82	9.37	1.81	9.37
2.04	8.74	2.32	8.94	2.31	8.94
2.54	8.33	2.82	8.52	2.81	8.53
3.04	7.92	3.31	8.12	3.31	8.12
3.46	7.68	3.82	7.70	3.81	7.71
3.78	7.47	4.32	7.29	4.31	7.30
4.25	7.09	4.82	6.90	4.81	6.91
4.75	6.71	5.32	6.52	5.32	6.52
5.25	6.33	5.82	6.15	5.60	6.31
5.75	5.98	6.11	5.96	5.86	6.10
6.26	5.67	6.18	5.93	6.33	5.70
6.62	5.32	6.39	5.73	6.61	5.47
7.08	4.92	6.86	5.31	6.65	5.43
7.44	4.56	7.15	5.05	6.86	5.22
7.91	4.30	7.40	4.80	7.32	4.78
8.27	3.98	7.87	4.34	7.82	4.28
8.73	3.56	8.37	3.85	8.32	3.79
9.08	3.21	8.86	3.36	8.82	3.29
9.54	2.99	9.37	2.87	9.32	2.80
9.90	2.71	9.87	2.39	9.82	2.31
10.37	2.24	10.37	1.92	10.32	1.83
10.87	1.78	10.87	1.47	10.83	1.37
11.36	1.33	11.37	1.06	11.33	0.94
11.71	0.99	11.85	0.78	11.83	0.65

como es el caso de los incendios, siendo capaz de generar respuestas y tomas de decisiones correctas en caso de detectar la presencia de algún obstáculo. De esta forma, el agente es capaz de tomar decisiones ante la presencia de objetos externos, como pueden ser el resto de UAVs del entorno u obstáculos presentes en el área de trabajo.

Para este cuarto modelo se ha mantenido una estructura de capas ocultas completamente conectadas, que permita generar una alta densidad de conocimiento basado en una red altamente conectada. En concreto, la estructura empleada es similar a la establecida en el modelo de aprendizaje 2, tal y como se recoge en la Tabla 5.5, con un aumento de las neuronas de cada una de las capas, dado que, aunque el espacio de observaciones se mantienen constante, la información procedente del entorno implica mayor complejidad ante la presencia de obstáculos

dinámicos.

Tabla 5.5: Estructura de la ANN para el modelo número 4 de DRL.

Capa (Tipo)	Forma de Salida	Parámetros #
dense_1 (Dense)	(None, 48)	384
dense_2 (Dense)	(None, 64)	3136
dense_3 (Dense)	(None, 128)	8320
dense_4 (Dense)	(None, 64)	8256
dense_5 (Dense)	(None, 48)	3120
dense_6 (Dense)	(None, 6)	294

En cuanto al resto de elementos fundamentales de los algoritmos de DRL, todo se mantiene constante con respecto al último modelo implementado, tanto en lo correspondiente al estado de acciones como a la estructura de las recompensas.

La idea de este método es que actúe como una nueva capa dentro de la arquitectura de navegación autónoma y coordinada del enjambre permitiendo a cada UAV alcanzar, sin supervisión, una ubicación de destino tomando acciones por sí mismo. Esta capa aporta inteligencia y autonomía a los sistemas y permite dar robustez al sistema formado por el resto de métodos presentados a lo largo de la tesis doctoral. Junto a ello, busca mejorar y optimizar la navegación autónoma desde una ubicación inicial a una ubicación objetivo, complementando a los métodos de planificación de trayectorias propuestos en el Capítulo 3.

Tal y como se ha detallado a lo largo del presente trabajo, en trabajos donde el tiempo de respuesta es un aspecto clave en el éxito de la misión, la distancia recorrida por los UAVs es un parámetro esencial a la hora de evaluar los métodos propuestos, ya que, ante el uso de sistemas similares, en los que aspectos dinámicos como la velocidad son idénticos, el tiempo empleado en alcanzar un objetivo viene determinado por la distancia recorrida hasta dicha ubicación. Por este motivo, se ha testado y validado el modelo de toma de decisiones propuesto sobre un escenario empleado para la validación de los métodos de planificación en 3D, comprobando si, el algoritmo DQN empleado permite dotar a cada UAV de la inteligencia suficiente para al menos establecer una solución igual de eficiente que los métodos anteriores propuestos.

Si se analiza el comportamiento del modelo de aprendizaje para este caso concreto, en el que se pretende alcanzar una ubicación final, se puede observar, mediante los datos recibidos del posicionamiento del UAV, como a medida que el agente navega sin supervisión por el entorno es capaz de ir reduciendo la distancia hacia el destino, siendo capaz de alcanzarlo, dentro de una tolerancia inferior a los 0,5 *metros*, tras haber recorrido 22,74 *metros*. En cuanto al algoritmo de planificación de trayectorias en 3D propuesto, para un mismo entorno, partiendo

desde la misma posición inicial y tratando de alcanzar la misma ubicación final, se establece una trayectoria segura, libre de colisiones, cuya distancia total a recorrer por el UAV es de 24,05 *metros*, quedando dicha comparativa resumida en la Tabla 5.6.

Por tanto, a raíz de este caso de estudio, se puede observar cómo esta capa de la arquitectura permite dotar al enjambre de cierta inteligencia para acometer una secuencia de acciones que les permita alcanzar una ubicación destino minimizando la distancia recorrida por el UAV. De esta forma, esta capa de desarrollo podría actuar sobre el control de cada uno de los agentes para establecer las maniobras a realizar para navegar sin supervisión a través de rutas generadas por el método de planificación propuesto. Junto a ello, este método de toma de decisiones inteligente se posiciona como una herramienta complementaria a los sistemas de detección y evasión de obstáculos presentados en el Capítulo 4, pudiendo ejercer como un tercer lazo de control y ayudar en la optimización de las maniobras de evasión a realizar en caso de detectar un obstáculo en la ruta previamente fijada.

Tabla 5.6: Optimización de trayectoria propuesta por el planificador en 3D mediante el uso de algoritmos de DRL.

	Trayectoria basada en DQN	Trayectoria Inicial
Distancia Re- corrida (m)	22,74 <i>metros</i>	24,05 <i>metros</i>

5.2. Conclusiones

A lo largo de este capítulo se presenta un conjunto de modelos de toma de decisiones inteligentes basados en DRL y, en concreto, en el algoritmo DQN para lograr la navegación sin supervisión de UAVs, desde una ubicación concreta a otro, mediante una toma de decisión autónoma a partir de la información obtenida del entorno.

Los avances en el campo de la AI y, en concreto, la combinación de algoritmos clásicos de RL con métodos de DL ha desembocado en la aparición de algoritmos basados en DRL capaces de trabajar en entornos de alta dimensionalidad. A partir de estos avances en el campo de DRL se ha implementado un conjunto de modelos de toma de decisión inteligentes capaces de establecer una secuencia de acciones sobre un UAV que den como resultado una navegación autónoma y segura del agente hasta alcanzar un objetivo concreto. Esta metodología se incluye como una capa adicional dentro de toda la arquitectura de navegación autónoma para enjambres de UAVs propuesta a lo largo de la tesis doctoral.

A lo largo del capítulo se recogen un conjunto de pruebas e implementaciones que han permitido desarrollar un conjunto de modelos de toma de decisión inteligente basados en el algoritmo DQN que cumplen con el objetivo marcado de alcanzar un destino de manera completamente autónoma. Por tanto, y a raíz de los resultados obtenidos, se puede concluir que los modelos de DRL basados en el algoritmo DQN constituyen una capa redundante que actúe como complemento al uso de otros desarrollos propuestos como la planificación de trayectorias combinada con métodos de detección y evasión de obstáculos basados en la percepción del entorno y replanificación de rutas.

Los resultados obtenidos abren una línea de investigación de alto potencial para conseguir agentes completamente autónomos, dotados de AI y capaces de tomar decisiones por sí mismo en función del resto de elementos participantes en el entorno.

Junto a estos modelos de toma de decisión, se incluye un método de aproximación de trayectorias basado en un triángulo de limitación del movimiento en el plano XY , el cual permite la mejora de la precisión con la que un UAV es capaz de alcanzar una ubicación determinada. Este aspecto presenta su importancia por dos aspectos principales:

1. Durante la navegación autónoma, autopilotos como Pixhawk poseen parámetros internos que modifican el comportamiento dinámico del UAV cuando se aproxima a un punto de ruta, disminuyen su velocidad y estableciendo un parámetro de tolerancia a partir del cual se considera que dicho objetivo ha sido alcanzado, pasando a la siguiente ubicación de destino.
2. El movimiento autónomo mediante el seguimiento de puntos de ruta requiere que el sistema sea capaz de alcanzar dichas ubicaciones dentro de una cierta precisión y tolerancia, debido a que, una navegación inexacta puede provocar la acumulación de errores, provocando que la trayectoria seguida no se aproxime a la deseada.

Por ello, no solo es importante que nuestro modelo de navegación inteligente sea capaz de alcanzar las ubicaciones con cierta precisión, sino que es importante conocer la precisión del mismo de cara a poder ajustar parámetros del autopiloto que permitan un comportamiento dinámico correcto de cada uno de los UAV.

Capítulo 6

Conclusiones y Trabajos Futuros

En la actualidad, los enjambres de UAVs se han posicionado como una herramienta tecnológica robusta, versátil y eficaz para el desempeño de un gran conjunto de aplicaciones y servicios, pero, sobre todo, en aquellas tareas en las que el tiempo de actuación y de respuesta es un factor clave en el desempeño de la misión, como es el caso de las emergencias y, en concreto, de los incendios forestales.

La posibilidad de desplegar, en un entorno adverso, un conjunto de UAVs, capaces de trabajar de manera autónoma y coordinada, permite a los equipos de emergencia contar con una herramienta de soporte y apoyo de alto valor añadido. La posibilidad de acometer tareas de manera autónoma va ligada a la necesidad de dotar al enjambre de UAVs con una arquitectura de múltiples capas formada por un conjunto de métodos y algoritmos que permitan la navegación, sin supervisión, por entornos de grandes dimensiones y complejidad.

6.1. Conclusiones

En la presente tesis doctoral se establece una arquitectura de software en la cual se combinan un conjunto de métodos destinados a permitir la navegación autónoma y coordinada de un enjambre de UAVs en entornos de diversa índole, como pueden ser incendios declarados en entornos forestales o urbanos. En el Capítulo 2, se han estudiado los principales progresos en el campo de los UAVs y su alta aplicabilidad en números servicios y sectores, enfatizando en su aplicación en emergencias, como es el caso de los incendios, dónde la empresa Drone Hopper aplica su propia tecnología en conjunto a los desarrollos realizados a lo largo del presente trabajo. Junto a este análisis, se incluye una revisión del estado del arte de esta tecnología, comenzando por aspectos generales relacionados con los MRS, profundizando en la especialización de los mismos hacia enjambres de UAVs y, por último, realizando una revisión pormenorizada de los métodos de planificación

de trayectorias que permiten la generación de planes de vuelo y navegación para enjambres de UAVs.

La principal contribución de este trabajo es desarrollar una arquitectura de software formada por múltiples capas en las cuales, mediante la implementación de un conjunto de métodos y algoritmos se busca generar soluciones que permitan la navegación autónoma y coordinada de un enjambre de UAV de cara a poder desempeñar una misión conjunta. Las soluciones propuestas buscan establecer un sistema de navegación para grupos de UAVs robusto, escalable y seguro que ponga solución a problemas como la planificación de rutas seguras y eficaces, la evasión de obstáculos, la navegación coordinada bajo una formación determinada o la toma de decisiones autónoma en función de la formación capturada del entorno.

La arquitectura de software propuesta presenta un conjunto de métodos desarrollados e implementado a lo largo de este trabajo y que, son resumidos a continuación:

- **Planificación de trayectorias en 2D:** Se ha propuesto y validado un algoritmo de planificación de trayectorias en 2D para enjambres escalables de UAVs basado en la combinación de métodos de exploración de muestreo probabilístico como es el algoritmo de PRM y de métodos de optimización de trayectorias como es el algoritmo A*.

El uso de algoritmos de exploración del entorno basado en PRM permite trabajar con entornos de grandes dimensiones y poder, en un corto intervalo de tiempo, generar un grafo en 2D con un conjunto de posibles trayectorias para alcanzar ubicaciones concretas del entorno. Este método de exploración, unido al algoritmo A* permite a un enjambre de UAVs alcanzar un conjunto de objetivos concretos o ser capaces de posicionarse, sobre una ubicación determinada, bajo una formación concreta.

La principal ventaja y aportación de este método es su escalabilidad basada en realizar una única exploración del entorno y aprovechar el resultado de la misma para encontrar el conjunto de trayectorias óptimas que permita, a cada agente del enjambre, alcanzar un destino concreto previamente fijado o, encontrar el conjunto de pares UAV-tareas que minimice la distancia total recorrida por el enjambre. Este aspecto también constituye una aportación de este método de cara a mejorar la eficacia del enjambre en tareas donde el tiempo de respuesta es crítico como es el caso de los incendios. Así pues, el método de planificación de trayectorias propuesto presenta una alta capacidad de configuración, siendo posible emplearlo en diferentes casos de uso, entre los que se encuentra la posibilidad de establecer el conjunto de rutas que permita minimizar la distancia total recorrida por el enjambre, de cara a mejorar los tiempos de respuesta y actuación del mismo.

La escalabilidad del método se logra a partir de la amortización de la generación del grafo, proceso realizado una única vez en el método independientemente del número de UAVs empleados y, sobre el que el algoritmo A* se encarga de encontrar la trayectoria óptima y segura que permite a cada UAV alcanzar una ubicación determinada en el entorno.

Este método de planificación en 2D tiene una alta aplicabilidad en entornos abiertos, en los cuales, a partir de cierta altura, el vuelo se puede considerar libre de obstáculos dinámicos y, por tanto, permite, de manera rápida, generar una solución óptima considerando obstáculos estáticos, que permita desplegar una flota de UAVs que trabaje de manera coordinada sobre una misma área de trabajo.

Por último, y de cara a eliminar la generación de trayectorias bajo un patrón en Zig-Zag, principal problema de los métodos de planificación de trayectorias basados en algoritmos de PRM, el método propuesto incluye un desarrollo que permite el suavizado de las trayectorias mediante la eliminación de puntos de ruta intermedios haciendo uso de técnicas de línea de vista. Esta técnica permite comprobar si existen puntos de ruta intermedios innecesarios debido a que desde un punto de ruta previo exista un camino libre de obstáculos disponible hacia un punto de ruta posterior, de tal manera que se eliminen los puntos de ruta intermedios.

- **Planificación de trayectorias en 3D:** En la presente tesis doctoral se incluye y valida un algoritmo de planificación de trayectorias en 3D para enjambres de UAVs basado, al igual que en caso de la planificación de trayectorias en 2D, en la combinación de métodos de exploración de muestreo probabilístico como es el algoritmo de PRM y de métodos de optimización de trayectorias como es el algoritmo A*, generando una solución escalable, en cuanto al número de agentes intervinientes en la misión, al problema de la planificación de trayectorias para enjambres de UAVs.

Los algoritmos de exploración del entorno basado en PRM poseen la propiedad esencial de poder ser empleados en problemas de altas dimensionalidad con un gasto computacional bajo, generando un grafo en 3D con un conjunto de posibles rutas sobre las que navegar hacia un destino de manera segura. De cara a poder aportar un método óptimo que minimice el tiempo que emplea la flota de UAVs en desplegarse en un entorno de emergencias, se hace necesario el uso de algoritmos que optimicen las trayectorias en aspectos como la distancia recorrida. Para acometer este aspecto, el método de planificación de trayectorias en 3D incluye el algoritmo A*, el cual genera un conjunto óptimo de trayectorias en función de la distancia recorrida.

El método propuesto presenta, en primer lugar, la ventaja de ser un método

altamente escalable, permitiendo mantener el gasto computacional empleado para la exploración del entorno y generación de trayectorias al ir aumentando el número de agentes participantes en el enjambre. En segundo lugar, la aplicabilidad del sistema a trabajos en entornos adversos como es el caso de los incendios hace que, en ocasiones, sean necesario acometer tareas que requieran de una formación específica del enjambre para navegar, por ese motivo, el método presentado, altamente configurable, permite generar trayectorias que permitan al conjunto de UAVs alcanzar una ubicación del área de trabajo bajo una formación concreta, acorde a las características del incendio. En tercer lugar, la parametrización de ciertos aspectos permite hacer uso del método propuesto en casos de uso donde un algoritmo previo ha asignado los pares UAV-tarea en función de aspectos como la alcanzabilidad, la mejor adecuación a la tarea o la posibilidad de transportar una cierta carga de pago, pero también, el método implementado en este trabajo permite, en el caso de enjambres homogéneos, realizar una asignación de tareas para cada vehículo que minimice la distancia total recorrida por el enjambre, mejorando así los tiempos de respuesta y, con ello, mejorando la eficacia en labores como la extinción o la monitorización de incendios.

Este método de planificación en 3D tiene una alta aplicabilidad tanto en entornos abiertos como en entornos urbanos, en los cuales, la presencia de obstáculos estáticos a diferentes alturas hace necesario la generación de trayectorias sobre un modelo 3D del entorno. Tal y como muestran los resultados de dicho método presenta una alta aplicabilidad a emergencias urbanas, en edificios de viviendas con un alto número de plantas, dónde se hace necesaria la monitorización y actuación coordinada en diferentes áreas del edificio.

Por último, y al igual que sucede con el método de planificación en 2D, los algoritmos basados en PRM presentan la desventaja de una generación de trayectorias bajo un patrón en Zig-Zag, lo cual puede provocar comportamientos dinámicos de las aeronaves poco deseados e incluso, problemáticos con el paso del tiempo. Por ello, el método propuesto para la planificación de trayectos incluye un desarrollo que permite el suavizado de las trayectorias mediante la eliminación de puntos de ruta intermedios haciendo uso de técnicas de línea de vista. Esta técnica permite comprobar si existen puntos de ruta intermedios innecesarios debido a que desde un punto de ruta previo existe un camino libre de obstáculos disponible hacia un punto de ruta posterior, de tal manera que se eliminen los puntos de ruta intermedios.

- **Navegación autónoma coordinada en enjambre:** El uso de enjambres de UAVs implica una actuación coordinada y un conocimiento del conjunto por cada uno de los agentes que forman el sistema. Por este motivo, en la

presente tesis doctoral se propone un método de navegación sin supervisión basado en una arquitectura centralizada que permite establecer un conocimiento común sobre cuál es la localización y posicionamiento de cada una de las aeronaves y, poder llevar a cabo acciones de control sobre cada uno de los elementos del enjambre para evitar colisiones en caso de dos o más UAVs se aproximen a una misma ubicación.

En primer lugar, este método permite generar una conciencia de grupo a los diferentes UAVs del enjambre, al poder, a través de un nodo central conocer la posición de cada una de las aeronaves. En segundo lugar, se propone una solución al problema de la navegación coordinada sobre un mismo entorno de trabajo, introduciendo una arquitectura de control que actúe sobre la velocidad de crucero de cada una de las aeronaves en caso de que dos o más se aproximen a un mismo punto de ruta y exista una posibilidad real de colisión. La validación del método sobre simulación permite concluir que dicho método permite a un enjambre de UAVs desplegarse por un entorno sin supervisión y navegar hacia diferentes ubicaciones siguiendo una ruta, previamente fijada, sin que ocurran colisiones entre miembros de un mismo enjambre.

- **Evasión de obstáculos:** La necesidad de compartir espacio aéreo con la aviación civil y comercial tradicional hace necesaria, cada día más, la incorporación de sistemas de detección, evasión y toma de decisiones de estas aeronaves no tripuladas, de cara a poder evitar colisiones con objetos dinámicos presentes en el entorno. Por este motivo y, debido también a la aplicabilidad de enjambres en áreas hostiles y altamente cambiantes como son las de los incendios, en este trabajo se presenta un método de detección y evasión de obstáculos basado en información 3D capturada del entorno.

La solución propuesta y, validada en simulación, permite a cada uno de los UAV capturar información 3D del entorno para, a través de un mapa de ocupación, detectar la presencia de obstáculos en las inmediaciones del mismo y, mediante un algoritmo de planificación de trayectorias, establecer una alternativa de navegación segura que evite la colisión con el objeto detectado. De esta forma, y mediante el uso de una arquitectura descentralizada, cada uno de los agentes es capaz de tomar un conjunto de decisiones que le permitan evadir el obstáculo y continuar hacia el destino fijado. Además, este método se presenta dentro de una arquitectura global que incluye entre sus capas el método de navegación coordinada anterior, generando entre ambos un sistema de gestión de colisiones redundante y eficaz.

- **Sistema de toma de decisiones inteligente para navegación autónoma de UAVs:** Para completar la arquitectura de software propuesta,

se ha establecido un desarrollo que pretende alcanzar el objetivo de dotar al enjambre de un comportamiento inteligente que permita, a cada uno de los miembros del mismo, navegar mediante la toma de decisiones autónoma, para conseguir alcanzar una ubicación concreta sin supervisión, se ha propuesto un método basado en DRL que permita la generación de modelos de aprendizaje basados en ensayo y error capaces de establecer una secuencia de acciones por sí mismos, que complemente el conjunto de métodos que forman la arquitectura de software y, permitiendo la navegación a través de una trayectoria óptima hacia un destino concreto.

Mediante el uso de ANNs se consigue la convergencia del modelo hacia una política óptima en la que se maximice la recompensa total acumulada fruto de la iteración del agente con el entorno. De esta forma, mediante un equilibrio entre exploración y explotación se consigue que el agente aprenda una secuencia de acciones que le haga obtener la mayor recompensa posible, obteniendo un modelo capaz de alcanzar una ubicación concreta mediante la toma de decisión autónoma de que acción realizar en cada instante. De esta forma, se presenta un método complementario que permite, no sólo dotar a cada UAV del enjambre de un comportamiento inteligente, sino dar robustez a la arquitectura de software presentada y, complementa la planificación global de trayectorias permitiendo una navegación autónoma en la que se alcancen ubicaciones concretas dentro del entorno mediante una secuencia de acciones que maximice el valor de retorno.

Esta tesis doctoral presenta un conjunto de soluciones tecnológicas para la actuación coordinada y autónoma de un enjambre de UAVs en entornos adversos y dinámicos como es el caso de los incendios, tanto urbanos como en áreas abiertas. Los métodos implementados y validados a lo largo del presente trabajo dotan al enjambre de un sistema de planificación de trayectorias, navegación sin supervisión coordinada, detección y evasión de obstáculos y toma de decisión autónoma e inteligente robusto, versátil y eficaz, que permita desplegar una flota de UAVs sobre un área de trabajo de cara a desempeñar tareas de soporte y ayuda a los equipos de extinción de incendios, tanto urbanos como forestales, ubicados en tierra.

6.2. Trabajos Futuros

A lo largo de la tesis doctoral se ha demostrado la potencial aplicación de los enjambres de UAVs para actuar de manera autónoma y coordinada diferentes entornos, mediante la implementación de algoritmos de planificación de trayectorias y navegación autónoma.

Los algoritmos y enfoques propuestos en este trabajo han mostrado diferentes soluciones, con altos niveles de robustez y precisión, en aplicaciones de navegación autónoma y coordinada de UAVs.

Sin embargo, para aumentar el rendimiento de estos algoritmos, se investigarán diferentes áreas en los trabajos futuros. Dichas investigaciones futuras se centran en las siguientes áreas:

- Tal y como se recoge en el Apéndice B, al término de la presente tesis doctoral, se está trabajando en el desarrollo y construcción de un enjambre de UAVs real, con estructura similar a la presentada a lo largo de la presente tesis doctoral, de cara a poder ser empleada como un demostrador de los algoritmos y métodos implementados a lo largo del presente trabajo. Debido a la situación excepcional acontecida en este último año, no se ha podido alcanzar un nivel de desarrollo mayor en esta área y, actualmente se trabaja para poder embarcar la arquitectura de software desarrollada sobre este dicho enjambre y comprobar la navegación autónoma y coordinada de plataformas aéreas reales.
- Uno de los aspectos cruciales que limita el uso de los UAVs es la capacidad de carga de pago y la autonomía de los sistemas. Por este motivo, se hace imprescindible poder contar herramientas de software que permitan obtener la mayor cantidad de información del entorno a partir del uso de sistemas de bajo peso y consumo. En la actualidad, se está trabajando en el desarrollo de métodos basados en DL que permitan la generación de mapas de ocupación tridimensionales a partir de la información visual obtenida por una cámara monocular (9). La salida de dicho método podrá ser empleada como entrada a los métodos de replanificación de rutas para evasión de obstáculos propuestos en este trabajo.
- Uno de los problemas detectados en la literatura en los métodos de DRL es la falta de estandarización de los modelos y, la necesidad de adaptar los entrenamientos a los entornos en los que se desea trabajar. Aunque los avances e investigación en el campo del DL están ofreciendo soluciones a esta cuestión. Por ese motivo, se estudia la posibilidad de incluir ANNs de mayor complejidad para incluir en el estado de observación información visual procedente de sensores embarcados en la aeronave. De esta forma, aumentar y mejorar el aprendizaje del agente de cara a conseguir un modelo robusto, heterogéneo y estándar capaz de trabajar de manera autónoma sobre diversos tipos de entornos. Además, el aumento de la información de entrada al agente no solo traerá cambios en la complejidad estructural de las redes empleadas, sino que implicará el uso de algoritmos de DRL más complejos.

- Otro campo de exploración y orientación de los futuros desarrollos es el del uso de algoritmos de DRL hacia sistemas multi-agente, de cara a poder aplicar los modelos de aprendizaje a un enjambre de vehículos autónomos, tanto aéreos como terrestres. Estos desarrollos están ligados a la generación de un conocimiento común en el que todos los agentes involucrados formen parte del proceso de entrenamiento, generando un aprendizaje común y colaborativo.
- Otra línea de investigación, la cual se está explorando en la actualidad, es la extrapolación de los métodos propuestos a enjambres de vehículos autónomos heterogéneos, en los que se produzca una colaboración entre vehículos autónomos terrestres y vehículos autónomos aéreos. Así pues, la posibilidad de planificar trayectorias en 2D y en 3D abre la puerta a dicha colaboración y, en la actualidad, se encuentra abierta una línea de investigación orientada a la navegación y procesamiento de información coordinada de estas dos clases de vehículos, que permita desarrollar aplicaciones y servicios orientados a tareas en entornos de rescate y desastre naturales (10) o a monitorización y soporte al tráfico rodado por carretera, generando una red robusta de sistemas inteligentes de transporte (254).
- Por último, otro área de futuros trabajos consiste en la adaptación de los métodos propuestos a autopilotos profesionales como los instalados recientemente en las plataformas aéreas de la empresa Drone Hopper. La necesidad de contar con autopilotos certificados de cara a la certificación y comercialización de los vehículos aéreos diseñados y fabricados por Drone Hopper, hace necesario la generación de protocolos de comunicación que permitan extrapolar los desarrollos y métodos presentados y validados en este trabajo, sobre el firmware de PX4 y el autopiloto Pixhawk, a los autopilotos profesionales de la empresa MicroPilot (255) embarcados en las plataformas aéreas de la empresa Drone Hopper.

En general, la tesis doctoral se centra en el desarrollo de una arquitectura de software compuesta por un conjunto de capas complementarias cuya implementación permita el trabajo cooperativo y autónomo de enjambres de UAVs en entornos dispares como puede ser la monitorización de áreas forestales o alcanzar ubicaciones remotas dentro de un edificio. Los métodos que conforman la arquitectura de software presentada se basan en tres áreas principales y esenciales para el desempeño autónomo de un sistema aéreo como son: la planificación de trayectorias, la detección y evasión de obstáculos y la toma de decisiones inteligente. Cada una de estas áreas es objeto de investigaciones, mejoras y avances futuros.

Apéndice A

Fundamentos del Aprendizaje por Refuerzo

Los métodos basados en RL se basan en un proceso de realimentación con nueva información, el cual se repite de manera cíclica hasta lograr el aprendizaje. El proceso se realiza de tal forma que el agente valora si cambiar o repetir la acción tomada en función de la recompensa obtenida. La Figura A.1 muestra un esquema de los métodos basados en RL junto con los principales elementos que conforman este área de conocimiento, los cuales son descritos a continuación.

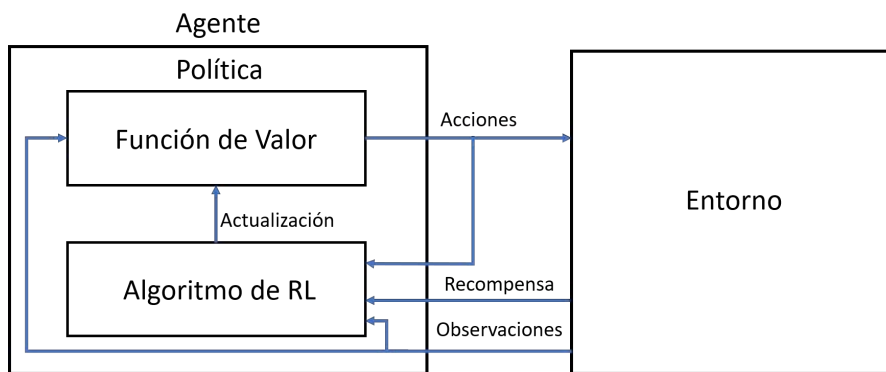


Figura A.1: Esquema general de métodos basados en RL.

1. **Entorno:** Su definición general engloba a todo aquel elemento que se encuentre fuera del agente y, la función del mismo consiste en recibir las acciones del agente y enviar las observaciones y recompensas en función de la misma.

Los métodos basados en RL poseen un modelo del entorno que busca imitar el comportamiento del mismo. Los modelos se emplean para llevar a cabo

una planificación del método, es decir, permiten conocer el curso de la acción considerando posibles situaciones futuras antes de que se experimenten realmente. Este elemento permite dividir los sistemas de RL en dos tipos:

- **Algoritmos de RL basados en modelo libre o sin modelo:** Se caracterizan por ser más sencillos y se basan explícitamente en el aprendizaje por ensayo-error. Este caso es útil para problemas generales o cuando es difícil representar o conocer el modelo del mismo. Para este tipo de algoritmos, el agente es capaz de aprender la política óptima para cada entorno, gracias al acceso a las observaciones, recompensas, acciones y estados internos.
- **Algoritmos de RL basados en el modelo del entorno:** En contraposición al anterior, se emplea un modelo del entorno que permite llevar a cabo una planificación del método, sin necesidad de que el agente explore todas las áreas para tener un entendimiento completo del mismo. En este caso, se dan datos conocidos del espacio de estados y el agente puede llevar a cabo una exploración en la que se evite áreas en las cuales la realización de determinadas acciones está ligada a la obtención de malas recompensas.

2. **Recompensas:** Este elemento se basa en una función que permitan al agente entender cuando la política mejora y converge hacia el resultado esperado. Dicha función, la cual no presenta restricciones a la hora de ser creada, produce un número escalar que representa cuanto de bueno es el agente en un estado y acción en concreto. La función depende de un conjunto de variables, tal y como se muestra en la Ecuación A.1, como son el estado actual y próximo del entorno y la acción escogida por la política.

$$r_t = R(S_t, S_{t+1}, a_t) \quad (\text{A.1})$$

La señal de recompensa es la base principal para modificar una política, si dicha política escoge una acción y la recompensa que acompaña a la misma es baja, la política puede establecer modificaciones para seleccionar otra acción en el futuro. Los algoritmos de RL establecen recompensas de diferentes tipos y adaptables al agente.

- **Sparse Reward:** Se establece una recompensa después de una larga secuencia de acciones, como por ejemplo el generar dicho valor después de que un UAV navegue durante un intervalo de tiempo o recorra una distancia determinada. Presenta el principal inconveniente de que la falta de recompensas durante un largo periodo de tiempo puede llevar

al agente a realizar acciones o visitar estados sin obtener un aprendizaje de los mismos.

- **Shaping Reward:** En este caso, se establecen pequeñas recompensas intermedias que guían al agente a lo largo del proceso de aprendizaje. Presenta el inconveniente de que puede provocar que se acumulen recompensas en situaciones no óptimas y, el agente lo entienda como un buen comportamiento.

Un aspecto clave en los algoritmos de RL relacionado con las recompensas es la resolución del conflicto Exploración-Explotación. El aprendizaje en línea del agente provoca que la toma de decisiones y acciones del agente determinen los datos y estados a conocer del entorno, de tal forma que si se toma una acción en cuestión (a_1), el agente puede alcanzar un estado concreto (s_2), pero no alcanzar otro estado diferente (s_3), tal y como muestra el esquema de la Figura A.2.

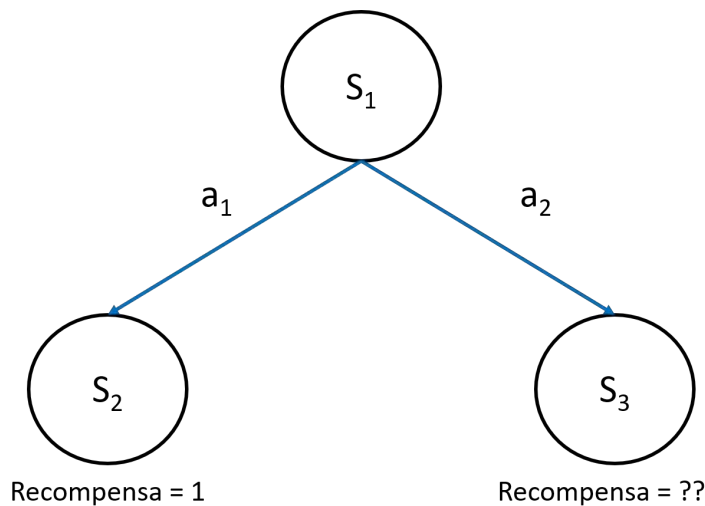


Figura A.2: Conflicto Exploración-Explotación.

Por tanto, se debe establecer una serie de instrucciones dentro de la política que determinen la manera de actuar del agente en cuanto a la elección de acciones, de tal forma que el agente se decante por escoger una recompensa conocida (explotación) o, por el contrario, si el agente debe tratar de elegir acciones que exploren el entorno desconocido en busca de nuevas recompensas. Lo recomendable es llevar a cabo una política que establezca un equilibrio entre ambos conceptos, ya que una explotación pura puede aumentar el tiempo empleado en encontrar una política óptima o llevar al agente a converger en una solución no óptima, mientras que en el caso de la

exploración pura es un método ineficiente, ya que, aunque se puede alcanzar una solución óptima global, la tasa de aprendizaje se ralentiza no encontrando una solución en un tiempo de aprendizaje razonable. Para alcanzar este equilibrio, los algoritmos de RL optan por una forma sencilla en la que se dota al agente de un aprendizaje basado en exploración al comienzo del entrenamiento y, gradualmente, se va dirigiendo las acciones del agente hacia la explotación.

En definitiva, la meta del agente es maximizar la recompensa acumulada o retorno. Estos métodos establecen dos tipos de retorno. El primero de ellos, conocido como retorno con horizonte finito, el valor final consiste en la suma de las recompensas obtenidas a lo largo de un conjunto de pasos. El segundo caso se conoce como retorno con descuento en un horizonte infinito, en el que el valor final corresponde a la suma de todas las recompensas obtenidas por el agente, pero considerando un factor de descuento, cuyo valor se establece entre 0 y 1, que permite considerar las recompensas en función de su importancia en el futuro. Cuanto más se aproxime a 0 este factor menos importancia les da el agente a las recompensas futuras.

3. **Agente:** El agente es elemento encargado de aprender un tipo de comportamiento para alcanzar un objetivo concreto. Está formado por la política y un algoritmo de aprendizaje.
 - **Política:** Constituye un elemento clave dentro de los métodos basados en RL, tal y como se describe a continuación. En líneas generales, se establece como una función encargada de mapear las observaciones hacia las acciones.
 - **Algoritmo de aprendizaje:** Es el método de optimización usado dentro del RL para encontrar la política óptima.
4. **Política:** Establece como va a comportarse el agente, mediante el mapeo de estados-acciones. Existen dos tipos de políticas, las deterministas y las estocásticas.
 - **Determinista:** A partir de un estado s se obtiene siempre la misma acción a , es decir, se da un mapeo previo de estados-acciones.
 - **Estocástica:** Se basa en una distribución de probabilidad sobre las diferentes acciones a alcanzar.

Por tanto, la política es el núcleo de un agente ya que por sí sola es suficiente para determinar el comportamiento del mismo.

5. **Observaciones:** Este elemento constituye la segunda fuente de información para el agente, junto con las recompensas, siendo las observaciones una pieza de información del entorno que permite conocer que está pasando alrededor del agente. Pueden ser relevantes para la próxima recompensa en casos en los que dicha observación incluya información sobre la misma.
6. **Acciones:** Tal y como puede deducirse a través de lo analizado hasta este momento, las acciones se corresponden con la forma con la que el agente puede relacionarse con el entorno, como por ejemplo pueden ser los movimientos permitidos. En los métodos de RL se distinguen dos tipos de acciones (193):
 - **Discretas:** Conjunto finito de cosas mutuamente excluyentes que un agente puede realizar, como por ejemplo moverse arriba o abajo.
 - **Continuas:** Poseen algún valor, como puede ser el giro de un volante, lo cual va ligado a un ángulo y dirección. Así, diferentes ángulos de giro podrían dirigir al agente a escenarios diferentes.
7. **Función Valor:** Se establece como la recompensa máxima que el agente puede ser capaz de acumular a partir del estado en el que se encuentra. No debe confundirse con la recompensa, dado que dicho término corresponde es una adquisición inmediata, mientras que la función valor determina el conjunto de recompensas obtenidas tras visitar un conjunto de estados. Debido a que el objetivo del RL es establecer una planificación de acciones que alcance la mayor recompensa posible, la función valor presenta un papel importante en los algoritmos de RL, pero, como se detalla a lo largo de esta sección, la obtención o definición de esta función es compleja y son necesarios métodos que permitan estimar esta función.

El RL presenta como estrategia más común considerar los problemas como un proceso de decisión de Markov (Markov Decision Process (MDP)), es decir, que el cambio a un estado futuro a través de la realización de una acción, solo depende del estado de partida y de la acción realizada, pero no a la sucesión de estados o acciones anteriores (256). Los MDP se caracterizan por una función de transición que posee la distribución de probabilidad de alcanzar un estado determinado a partir de uno previo tras ejecutar una acción. Junto a esta función de transición, los MDP, incorporan una función de recompensa que devuelve los valores obtenidos al visitar el siguiente estado.

Junto con los elementos y formalismos claves de los algoritmos de RL, se establece, como aspecto fundamental a tratar, los enfoques principales de los algoritmos de RL para resolver los problemas. Estos enfoques se clasifican en métodos basados en funciones de valor, métodos basados en búsqueda de políticas y una hibridación de ambos conocida como actor-crítico (257).

1. **Métodos basados en la función valor:** Se basan en estimar el valor o retorno esperado de encontrarse en un estado dado. La función *estado – valor*, recogida en la Ecuación A.2, se define como la recompensa esperada al seguir una política π comenzando en un estado determinado s . Por tanto, dichas funciones estiman, siguiendo dicha política, cuanto de bueno es para un agente encontrarse en un estado determinado. Además, si dicha función sigue la política óptima (π^*), se establece como la función *estado – valor* óptima (V^*).

$$V^\pi(s) = \mathbb{E}[R|s, \pi] \quad (\text{A.2})$$

La imposibilidad de disponer de una dinámica de transiciones hace necesaria la construcción de una nueva función, similar a V^π , con la diferencia de que en este caso se proporciona la acción inicial a , conocida como función de calidad ($Q^\pi(s, a)$) y formulada a través de la Ecuación A.3. Por tanto, esta función permite conocer el refuerzo que se obtendría si, empleando dicha política, se ejecuta la acción determinada a desde el estado concreto s .

$$Q^\pi(s, a) = \mathbb{E}[R|s, \pi] \quad (\text{A.3})$$

En este caso, también se da la función de calidad óptima ($Q^*(s, a)$), encargada de maximizar el valor de los pares estado-acción para cualquier política.

La programación dinámica se establece como una de las maneras de poder explotar la propiedad de los MDPs y aprender realmente Q^π a través del uso de la Ecuación de Bellman, lo cual quiere decir que Q^π se puede mejorar mediante el uso de los valores actuales de la estimación. Esto quiere decir que, a medida que se acumulan recompensas se actualizan los estados previos y la acción provocando que el agente mejore su aprendizaje y pueda alcanzar la mayor recompensa. Este principio, conocido como *bootstrapping* cuya formulación se recoge en la Ecuación A.4, de actuación es la base de algoritmos como el *Q – Learning* y el algoritmo de *estado – acción – recompensa – estado – acción* (state-action-reward-state-action (SARSA)).

$$Q^\pi(s_t, a_t) \leftarrow Q^\pi(s_t, a_t) + \alpha \delta \quad (\text{A.4})$$

dónde α corresponde a la tasa de aprendizaje y δ es el error de la diferencia temporal (Temporal Difference (TD)).

Por tanto, para encontrar Q^* a partir de un Q^π se emplean iteraciones de políticas generalizadas, que se basan en la evaluación y mejora de las mismas.

La evaluación de políticas mejora la estimación de la función valor a través de la minimización de los errores TD de la consecución de estados y acciones visitados al seguir la política.

Al margen de la programación dinámica se encuentran los métodos de Monte Carlo, los cuáles emplean el promedio de los retornos de múltiples implementaciones de la política para poder estimar el retorno esperado.

2. **Métodos basados en búsqueda de políticas:** Se caracterizan por realizar la búsqueda directa de una política óptima (π^*), sin necesidad de un modelo de función de valor. A partir de la actualización de los parámetros de una política parametrizada, utilizando una optimización basada en gradientes o sin gradientes, se maximiza el retorno o rendimiento esperado.

En este tipo de métodos, en los cuales se construye la política directamente basándose en gradientes, es habitual producir parámetros para una distribución de probabilidad, que en el caso de las acciones continuas se base en desviaciones medias y estándar de distribuciones gaussianas, mientras que en el caso de acciones discretas siga una distribución multinomial. Como resultado se genera una política estocástica encargada de muestrear directamente las acciones.

En el caso de los métodos sin gradientes, la búsqueda de políticas se basa en una búsqueda heurística mediante una clase predefinida de modelos, presentando la ventaja de poder optimizar políticas no diferenciables.

3. **Métodos Actor-Crítico:** Estos métodos surgen de la combinación de funciones valor con una representación explícita de la política. El actor constituye la política y trata de tomar la que cree que es mejor acción mediante el feedback procedente del crítico, basado en funciones de valor, el cual trata de estimar el valor del estado y la acción que toma el actor.

Permite el trabajo en espacios continuos debido al que el crítico solo mira la acción única que toma el actor, sin tener que buscar la mejor acción evaluando todas ellas. En este caso, cada uno mantiene las tareas propias de sus métodos. El actor elige una acción, la cual se aplica sobre el entorno, mientras que el crítico hace una predicción del valor de esa acción para el conjunto estado actual-acción.

A la salida del crítico se introduce una fase de comparación que permite, a partir de la recompensa, determinar la exactitud de la predicción del valor. De esta manera, se emplea un error basado en la diferencia entre el nuevo valor estimado y el anterior para que el crítico genere un aprendizaje de como de buena fue la acción tomada. El esquema mostrado en la Figura A.3 muestra el proceso seguido en este método y, como se puede observar, el

crítico emplea este error para realizar actualizaciones sobre sí mismo. En el caso del actor, se actualiza con esta respuesta del crítico, pudiendo ajustar las probabilidades de tomar la acción de nuevo en el futuro.

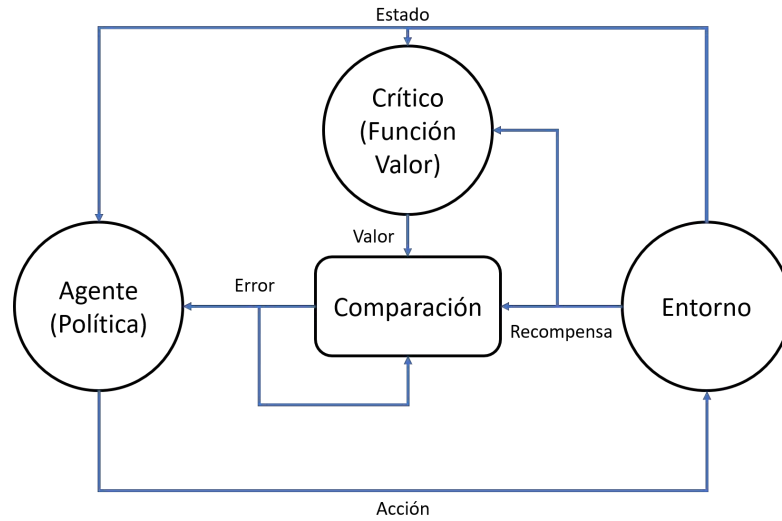


Figura A.3: Diagrama método Actor-Crítico.

Apéndice B

Arquitectura de Hardware

Hasta este punto, todos los trabajos e implementaciones realizados en la presente tesis doctoral se encuentran vinculados al desarrollo software que permita establecer una arquitectura de navegación autónoma multicapa, aunque entre los avances y trabajos realizados dentro de la tesis doctoral se encuentra un importante desarrollo de hardware, cuya consecución ha permitido disponer de un enjambre de UAVs, como el que se muestra en la Figura B.1 destinado a banco de pruebas para la validación de los métodos que conforman la arquitectura de software para navegación autónoma propuesta, en los capítulos previos de este trabajo, sobre plataformas aéreas reales.



Figura B.1: Enjambre de UAVs homogéneo resultado del desarrollo de hardware realizado.

La idea principal es construir una arquitectura de hardware que presente el mayor grado de similitud a la arquitectura de software empleada en simulación y, presentada en la Sección 3.3.1, de cara a facilitar la implementación y validación,

sobre plataformas reales, de los métodos desarrollados y testados en simulación.

El objetivo de la arquitectura de hardware construida es el desarrollo y constitución de un enjambre de UAVs homogéneo y estandarizado sobre el que llevar a cabo pruebas de navegación autónoma y coordinada. Para ello, se han desarrollado un conjunto de multirrotores, de similares características, en el que cada UAV está formado por: chasis, módulo de alimentación, módulo de potencia, autopiloto, módulo de posicionamiento, módulo de comunicaciones y módulo de procesamiento.

En las secciones del presente capítulo se detallan cada uno de los elementos que conforma las plataformas aéreas construidas, analizando sus principales características y funciones dentro de la arquitectura de hardware.

B.1. Chasis

El primer elemento clave a describir es la estructura sobre la que se realiza el montaje del resto de elementos que conforman el UAV completo. La idea es disponer de un demostrador de vuelo en enjambre sencillo, pero que pueda alojar el conjunto de sistemas necesarios para permitir una navegación autónoma y coordinada del multirrotor dentro del enjambre.

Por ello, y debido también a las restricciones legislativas dentro del campo de los UAVs, se ha optado por emplear multirrotores de dimensiones reducidas que posean la capacidad de albergar el conjunto de módulos y sistemas imprescindibles para testar y validar la arquitectura de software para navegación autónoma propuesta. Por tanto, debido a estos dos motivos principales, se ha establecido como elemento estructural principal el chasis S500 Quad, el cual permite disponer de un UAV de cuatro motores como el que se muestra en la Figura B.2.

Este sencillo frame posee unas dimensiones de 480 *mm* de motor a motor y una altura de 170 *mm* y un peso de 425 *g*, lo que lo hace ligero de cara a poder destinar el empuje de los motores a levantar sistemas embarcados o carga útil necesaria para la navegación autónoma del sistema. A pesar de la ligereza del chasis, posee una gran resistencia gracias a la fibra de vidrio y nylon en la que está construido.

Aunque sobre dicho chasis se pueden embarcar la mayoría de módulos necesarios para el funcionamiento correcto del UAV tanto en vuelo manual como en vuelo autónomo, de cara a mantener la estabilidad del mismo y facilitar la instalación de los sistemas, ha sido necesaria la modificación del chasis y la adición de elementos estructurales construidos en Ácido poliláctico (PLA), de diseño propio y generados mediante impresora 3D, para poder embarcar elementos como el módulo de procesamiento.

Así pues, se ha modificado la estructura original, sustituyendo la pieza situada en la parte inferior del frame destinada al soporte del estabilizador de la cámara o la



Figura B.2: Chasis S500.

batería, por la pieza diseñada en PLA que, permite embarcar en el UAV elementos correspondientes al módulo de comunicaciones, al módulo de alimentación y al módulo de procesamiento. Este nuevo elemento estructural, cuyo diseño se observa en la Figura B.3, no solo permite albergar en el UAV un conjunto de elementos indispensables para el vuelo autónomo en enjambre del agente, sino que permite el reparto de pesos a lo largo del chasis, de cara a conservar la estabilidad del UAV manteniendo centrado tanto el centro de masas como el centro de gravedad del mismo. Con tan sólo 67 g , una ligereza que no condiciona su resistencia permite el anclaje al chasis de elementos esenciales como el ordenador de a bordo, las antenas WiFi para las comunicaciones con la GCS o el sistema de alimentación, introduciendo un peso extra de 50 g dado que este nuevo diseño viene a sustituir a una pieza original del chasis.

Por tanto, con la combinación de los elementos descritos en esta sección, se posee un chasis ligero (475 g), de dimensiones reducidas que pueda albergar el conjunto de sistemas necesarios para abordar una misión de manera autónoma con un enjambre de UAVs, los cuales son descritos a continuación.

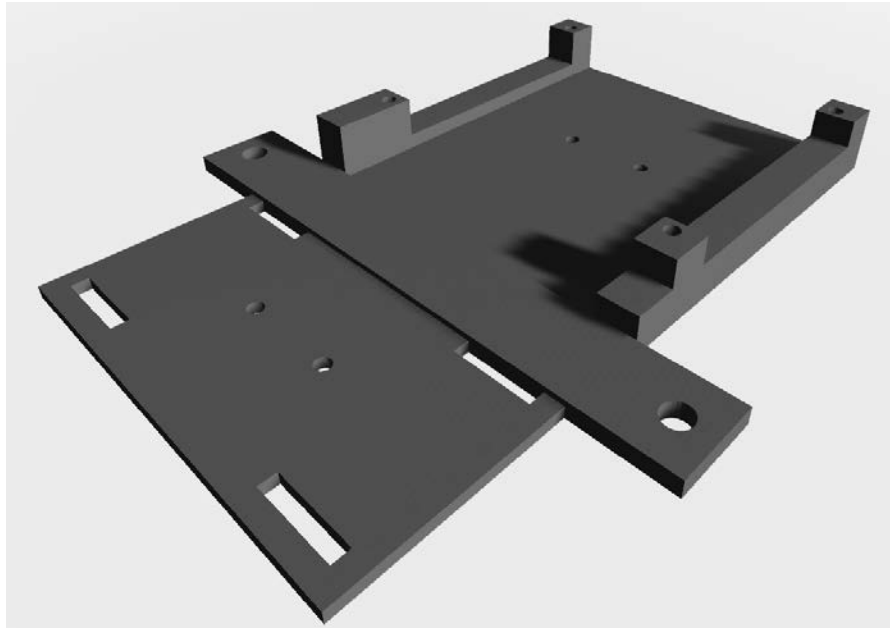


Figura B.3: Elemento estructural de PLA para embarcar el sistema de procesamiento, alimentación y comunicaciones.

B.2. Autopiloto

Este elemento constituye el cerebro del UAV, siendo el encargado del control automático de la plataforma aérea. En este caso, y tal y como se introdujo previamente en la Sección 3.3.1, se ha optado por emplear el piloto automático Black Cube de la empresa Pixhawk, el cual se basa en software libre y está orientado al uso de UAVs en modo autónomo.

Aunque no se puede considerar un autopiloto profesional desde el punto de vista de certificaciones por parte de las agencias de seguridad aéreas, es una controladora bastante comercial y extendida en el mercado de los drones de ocio, entretenimiento y proyectos industriales de bajo costo. Esto se debe, principalmente, a la relación calidad-precio de este piloto automático ya que, posee unas buenas especificaciones a un precio competitivo dentro del mercado de los autopilotos para UAVs. Entre las especificaciones más destacadas de la presente controladora se encuentra la redundancia en sensores claves como la Unidad de Medición Inercial (Inertial Measurement Unit (IMU)), los barómetros, un peso ligero de 300 g o, la posibilidad de instalar múltiples sistemas de posicionamiento. Junto a esto, el sistema cuenta con un segundo procesador a prueba de fallos de cara a mejorar la robustez de la unidad de control del UAV.

Este piloto automático destaca, además, por poseer toda la unidad de gestión

de vuelo o Flight Management Unit (FMU) y la IMU alojadas en el interior de un cubo, de dimensiones reducidas tal y como se observa en la Figura B.4, aisladas, para reducir la interferencia entre sensores, y amortiguadas mecánicamente, para disminuir el efecto de las vibraciones de alta frecuencia.

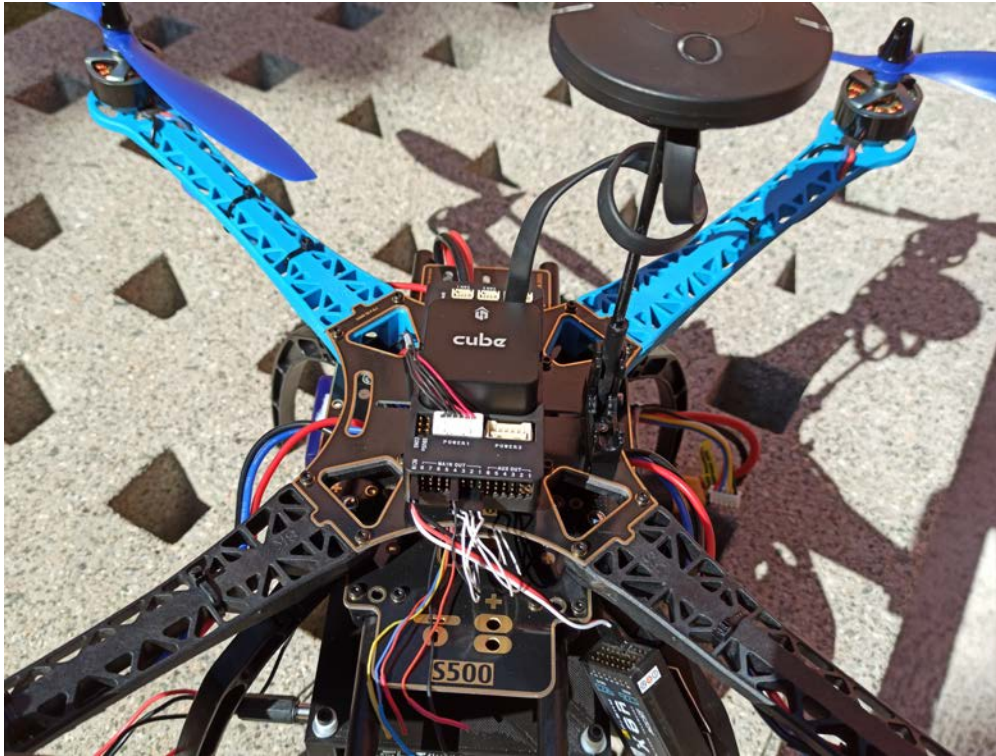


Figura B.4: Controladora Pixhawk Black Cube y sistema de posicionamiento *Here+* embarcados sobre UAV.

Dentro del vuelo autónomo, el autopiloto, configurado con el firmware *PX4*, es el encargado de llevar a cabo las instrucciones de control al módulo de potencia que permiten mantener la estabilidad del UAV y navegar hacia el objetivo a partir de las órdenes de navegación generadas por los métodos de software propuestos dentro de la arquitectura presentada a lo largo de la tesis doctoral. Junto a esta función, la controladora es la encargada de, mediante el uso de sensores conectados a la misma, suministrar información sobre el estado del UAV a los algoritmos de navegación autónoma y, de esta forma, poder conocer el modo de vuelo, la localización y posicionamiento del UAV o el nivel de batería. Dicha información se extrae del autopiloto a través del puerto serie de telemetría y, se produce el intercambio de información con la arquitectura de software propuesta a través del protocolo de comunicaciones MavLink.

Como se ha indicado al comienzo de la presente sección, el autopiloto consti-

tuye el cerebro del UAV. Por este motivo, en primer lugar, al constituir el centro de control del UAV posee un alto número de parámetros cuya configuración es necesaria para el buen hacer del sistema aéreo. Dentro de estos parámetros destacan aquellos relacionados con el comportamiento del sistema durante el modo autónomo u *Offboard*, como son la velocidad máxima de crucero, la velocidad vertical máxima al despegue, el umbral a partir del cual el autopiloto considera que ha alcanzado una ubicación concreta o, la posibilidad de suministrar corriente a los motores estando el autopiloto conectado por puerto serie a algún dispositivo de procesamiento.

En segundo lugar, es necesario que se encuentre conectado al resto de sistemas que forman parte de la plataforma aérea, tal y como se detalla en las secciones posteriores, para poder establecer un correcto funcionamiento durante el vuelo, tanto manual como autónomo. Así pues, la controladora se encuentra conectada al sistema de alimentación para nutrir a los sensores embarcados en el mismo de corriente para su correcto funcionamiento, al módulo de potencia para poder enviar la señal de control correspondiente a cada uno de los motores encargados del empuje y sustentación del UAV y, por último, al módulo de procesamiento mediante el puerto de telemetría para el intercambio de información con la arquitectura de software propuesta encargada de la navegación autónoma del enjambre.

B.2.1. Módulo de posicionamiento

Aunque puede considerarse un módulo independiente, se incluye dentro del autopiloto pues corresponde a un elemento indispensable del mismo ya que, sin dicho módulo es imposible el vuelo del UAV. Si bien, el autopiloto Pixhawk cuenta con modos de vuelo que permiten realizar maniobras, siempre en vuelo manual, sin una señal de GPS correcta, si es necesario, tanto para la calibración del sistema como para la activación de motores y vuelos manuales, disponer de un sistema de posicionamiento conectado al autopiloto, cuyo posicionamiento dentro del UAV se recomienda que sea en una posición ligeramente elevada, de cara a evitar interferencias magnéticas con el resto de elementos.

En este caso, se ha optado por un sistema GPS *Here+* capaz de funcionar con sistemas de posicionamiento Real Time Kinematic (RTK), es decir, de proporcionar, a partir de una única estación de referencia, correcciones en tiempo real para obtener una precisión en el posicionamiento GPS con un orden de magnitud de centímetros. Dicho sistema debe ser instalado sobre el frame del UAV respetando la orientación del autopiloto y, conectado al mismo mediante el puerto de GPS establecido para ello. Su diseño compacto y ligero, tal y como puede verse en la Figura B.4, alberga tanto el módulo del sistema global de navegación por satélite o Global Navigation Satellite System (GNSS), como una brújula digital para la

referencia del rumbo y, indicadores LED para conocer, de manera visual, el estado del UAV en cada momento.

Este módulo no es imprescindible, únicamente, desde el punto de vista del funcionamiento del autopiloto, sino que es necesario para poder llevar a cabo vuelos autónomos y, en concreto, constituye una entrada clave a la arquitectura de navegación autónoma en enjambre propuesta.

B.3. Módulo de Procesamiento

El módulo de procesamiento está formado principalmente por el ordenador embarcado en el UAV, el cual se encarga, principalmente, de ejecutar una serie de algoritmos y configuraciones necesarios para el correcto funcionamiento de la arquitectura de software para navegación autónoma en enjambre propuesta.

De cara a poder contar con un conjunto de UAVs versátiles, cuyo uso permita la ejecución de algoritmos complejos basados en visión por computador, análisis del entorno basado en información tridimensional o modelos de toma de decisiones inteligentes se ha optado por embarcar en cada UAV un ordenador embebido de altas capacidades como es la Jetson AGX Xavier, como se muestra en la Figura B.5, la cual permite procesamiento en la Unidad de Procesamiento Central o Central Processing Unit (CPU), como en la Unidad de Procesamiento Gráfica o Graphics Processing Unit (GPU).



Figura B.5: Jetson AGX Xavier embarcada sobre UAV.

A pesar de poseer unas grandes especificaciones el ordenador embebido escogido cuenta con unas dimensiones ($105\text{ mm} \times 105\text{ mm} \times 65\text{ mm}$) y peso (600 g)

reducidos, lo cual permite su utilización en UAVs de pequeño tamaño como los presentados en la Sección B.1.

De cara a poder establecer un enjambre de UAVs físico que sirva como demostrador de la arquitectura de software propuesta, es necesario establecer una arquitectura de hardware que guarde la mayor de las similitudes con respecto a la arquitectura empleada en simulación. Así pues, se hace necesario que dicha arquitectura cuente con una GCS u ordenador central en tierra dónde se encuentre el nodo maestro (ROS_MASTER) de la arquitectura de software propuesta y, sobre él, se ejecuten y ubiquen el conjunto de nodos de ROS individuales correspondientes, en primer lugar, a cada uno de los UAV del enjambre y, en segundo lugar, a los nodos creados para cada uno de los métodos empleados dentro de la arquitectura. La Figura B.6 recoge un esquema general del proceso de comunicación e intercambio de información de la arquitectura de hardware propuesta y, que presenta un alto grado de similitud con la arquitectura propuesta en las simulaciones.

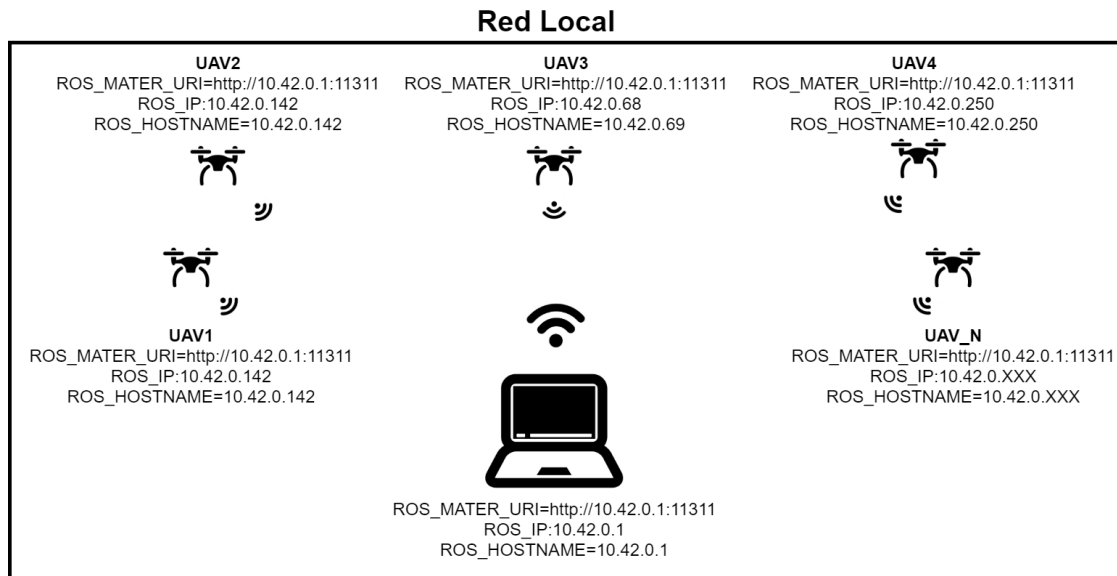


Figura B.6: Esquema general de la arquitectura de hardware propuesta.

A partir de la Figura B.6, se analiza la arquitectura propuesta y, se detalla el procedimiento seguido para desplegar un enjambre de UAVs sobre un área de trabajo de cara a realizar una navegación autónoma y coordinada real sobre el mismo.

La base de la arquitectura diseñada versa en el hecho de emplear un solo nodo central sobre el que establecer la comunicación con el conjunto de los UAVs que participen en el enjambre. De esta forma, desde el ordenador central, ubicado en tierra, se habilita un acceso WiFi a modo de red local para el conjunto de

agentes del enjambre y, sobre dicha GCS, se ejecuta el nodo maestro de ROS sobre el que se ejecuten tanto los métodos propuestos dentro de la arquitectura de navegación autónoma, como los nodos de ROS propios de cada UAV, encargados de habilitar, mediante MavLink, el intercambio de información entre el autopiloto y los algoritmos de navegación autónoma.

Como se detalla en la Sección B.3.1, el sistema de procesamiento incluye un módulo de comunicaciones WiFi, que permite a cada uno de los UAV disponer de conectividad a redes locales mediante el uso de esta tecnología. Por tanto, dentro de esta red local habilitada se asigna, a cada UAV del enjambre, una dirección IP fija a partir de la cual establecer una conexión por protocolo Secure SHell (SSH) entre la GCS y cada agente del enjambre. Una vez establecida dicha conexión, es necesario ejecutar, sobre cada uno de los ordenadores de a bordo, un archivo de configuración que establezca como nodo maestro de la arquitectura aquel que está siendo ejecutado en la GCS. Junto a este archivo de configuración, se ejecuta MAVROS, es decir, el paquete de ROS encargado de establecer un puente de comunicaciones vía MavLink entre el autopiloto y la arquitectura de navegación autónoma propuesta. En este punto, es importante aclarar que, de cara a la escalabilidad del sistema, es necesario que cada UAV cuente con un identificador de grupo que se añada al archivo de lanzamiento de MAVROS. De esta forma, se puede establecer una diferenciación entre los nodos correspondientes a un agente u otro y, se puede mantener la arquitectura centralizada en la que únicamente se cuente con un nodo maestro en toda la arquitectura.

Una vez establecida esta configuración dentro de la arquitectura de hardware el enjambre físico desarrollado se encuentra en una situación similar al enjambre simulado. Por tanto, a partir de este punto, el procedimiento a seguir para la ejecución de los métodos y algoritmos propuestos a lo largo de la tesis doctoral desde la GCS es similar al realizado en simulación, de tal forma que, no es necesaria la ejecución de los algoritmos en cada uno de los UAVs por separado y, se puede mantener la centralización de la arquitectura de cara al intercambio de información entre los diferentes agentes del enjambre.

B.3.1. Módulo de comunicaciones

El ordenador embebido embarcado en cada UAV cuenta con un módulo de comunicaciones que permite la conectividad WiFi de los agentes a redes locales para poder ser operados de manera autónoma y remota. Para ello, cada una de las Jetson AGX Xavier cuenta con un adaptador inalámbrico de banda dual, como es el Intel Dual Band Wireless A-8265, provisto de una tasa de transferencia máxima de 867 Mbps/s y un estándar WiFi IEEE 802.11ac de doble banda, lo cual permite establecer conexiones desde cada agente a redes WiFi que emiten en banda de 2,4

GHz y en banda de 5GHz.

Las dimensiones reducidas del adaptador (22 mm x 30 mm x 2,4 mm), su compatibilidad con Jetson AGX Xavier, su fácil instalación y su posibilidad de operar en distintas bandas de frecuencias la hacen ideal para las plataformas aéreas propuestas. De cara a aumentar el rango de cobertura dentro de una red local, dicho adaptador cuenta con dos antenas de comunicación, las cuales establecen el rango de la operativa en un radio de 100 metros y, que son embarcadas en el frame del UAV a partir del elemento estructural diseñado e impreso en 3D, descrito en la Sección B.1.

B.4. Módulo de Alimentación y Potencia

Está formado por la batería y un circuito de eliminación de la batería o Battery Eliminating Circuit (BEC) y, es el encargado de generar, transportar y convertir a los valores correctos la energía eléctrica del conjunto de elementos del UAV.

El elemento clave dentro de este módulo es la batería, la cual constituye la única fuente de energía dentro de cada UAV y, por tanto, se encarga de alimentar a todos los sistemas embarcados en el mismo. Junto a la batería se incluye un BEC, es decir, un circuito electrónico encargado de convertir el voltaje suministrado por la batería, el cual por lo general es superior a lo soportado por elementos como los ordenadores embebidos o ciertos sensores, a diferentes valores de voltajes, a los cuales son alimentados los sistemas embarcados del UAV. Junto a esta conversión, el BEC permite eliminar el ruido eléctrico, o lo que es lo mismo picos de tensión, con el que son alimentados los diferentes componentes. Así pues, con estos dos elementos como base, dentro de cada UAV se establece una esquema de conexiones a nivel eléctrico como el mostrado en la Figura B.7.

Así pues, a partir de una batería LiPo de 4 celdas, cuyo voltaje máximo de salida es 16,8 *Voltios*, se alimenta el BEC, el autopiloto y el control de velocidad electrónico o Electronic Speed Control (ESC) encargado de controlar las revoluciones de giro de los motores.

A la hora de elegir la batería, es importante establecer una relación potencia-peso correcta, ya que el aumento de la capacidad y potencia suministrada por la batería va ligada a un aumento del tamaño y peso de la misma, lo cual hace que dicho aumento de potencia no vaya acompañado de un aumento significativo de la autonomía de vuelo. Otro aspecto que condiciona la elección de la batería es la planta de potencia de cada UAV, es decir, el conjunto ESC-motor-hélice encargado del empuje y sustentación del UAV. En este caso, el tamaño reducido del chasis limita la longitud de la hélice empleada, y con ello, el tipo de motor a embarcar, cuyo voltaje máximo de funcionamiento marca el número de celdas máximo por

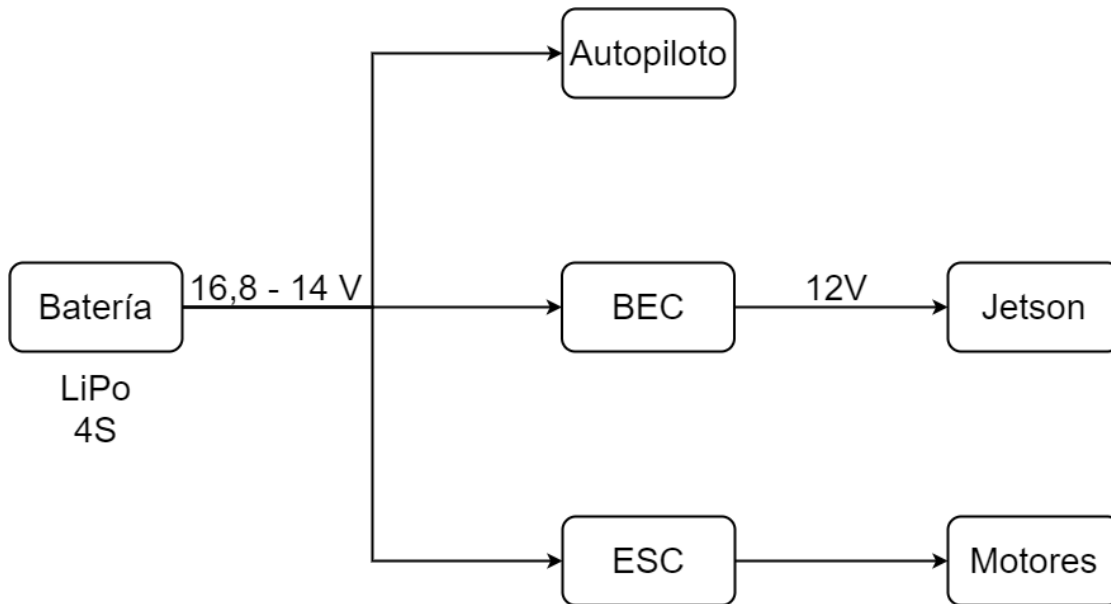


Figura B.7: Esquema general de conexiones eléctricas dentro de cada UAV.

batería, que en los UAVs construidos se establece en 4S, y un peso aproximado por batería de 550 *g*.

Tal y como muestra el esquema eléctrico de la Figura B.7, la batería es conectada al autopiloto a través del módulo de potencia suministrado por Pixhawk para, en primer lugar, alimentar el autopiloto y todos los sensores albergados en él y, en segundo lugar, para poder monitorizar el voltaje de la batería y, con ello, tener un conocimiento sobre la autonomía de vuelo de cada UAV de cara a evitar posibles incidentes debidos a pérdidas de potencia. Junto a la controladora, el BEC también es alimentado por la fuente de energía principal, para convertir el voltaje de salida de la batería a un voltaje estable de 12 *Voltios* que alimente de manera continua el ordenador de a bordo. Por último, el ESC, encargado del control de velocidad de giro de los motores, también se conecta a la batería para transmitir, mediante pulsos, la corriente de alimentación a los motores.

Estrechamente ligado con el módulo de alimentación se encuentra el módulo o la planta de potencia del UAV. Aunque dentro de la planta de potencia se incluye un elemento ya analizado como es la batería, los elementos principales encargados de la generación de empuje en el sistema es el conjunto ESC-motor-hélice, que junto a las señales de control enviadas por la controladora se encargan de establecer el régimen de revoluciones apto para mantener la estabilidad y la altitud del UAV durante el vuelo.

La elección de este conjunto está totalmente relacionada, ya que en función de la longitud y paso de la pala el motor suministrará un empuje concreto a costa de

demandar una corriente de la batería, la cual es controlada por el ESC. Por tanto, a la hora de dimensionar este conjunto se debe, en primer lugar calcular el empuje necesario por motor para el vuelo del UAV, este empuje se calcula a partir de la Ecuación B.1. A partir de dicho cálculo y, en función del tamaño máximo de hélice que se pueda montar sobre el chasis, se establece que motor es capaz de suministrar el empuje necesario y, a través de las tablas de especificaciones de los motores poder conocer cuál es la demanda de corriente máxima para dicho motor y dicha hélice. Una vez conocida esta corriente máxima al voltaje de trabajo establecido por la batería, se puede establecer el tipo de ESCa embarcar en la plataforma de cara a que todos los elementos sean capaces de resistir la demanda de corriente máxima del motor.

$$Empuje_{Motor} = \frac{2 \cdot Peso_{UAV}}{Num_{Motores}} \quad (B.1)$$

Por tanto, considerando todos los elementos descritos hasta ahora el peso total del UAV se sitúa en torno a los 2 *Kg*, a lo que hay q añadir el peso significativo de los motores y de los ESCs. Considerando todos los elementos se puede estimar un peso aproximado del UAV de 2,5 *Kg*. Por lo que, estableciendo como regla que el empuje total suministrado por el conjunto de motores debe ser de al menos el doble, se establece la necesidad de un empuje por motor de 1,25 *Kg*. En función de este cálculo, y con una hélice máxima de 10 *pulgadas* y un paso de 4,5 *pulgadas*, se establecen cuatro motores de 750KV, cuyo empuje supera los 1,25*Kg* necesarios y cuya corriente máxima demandada es de 27 *A*. Una vez establecida la corriente de funcionamiento, se establecen los controladores ESC cuya especificación a considerar es la corriente máxima soportada y el rango de voltaje al cual puede trabajar. Si ambos valores se encuentran dentro de los establecidos para cada UAV el ESC es válido. En este caso, se ha optado por un ESC de la empresa T-Motor que alberga, en un sólo dispositivo de 17,5*g* como el mostrado en la Figura B.8, todos los circuitos de control correspondientes a los 4 motores del multirroto.

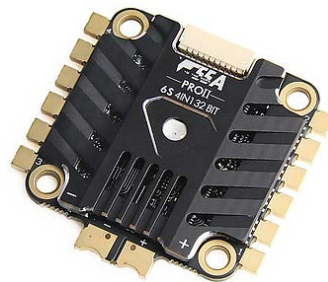


Figura B.8: ESC modelo Tmotor F 55A montado sobre cada uno de los UAVs.

B.5. Conclusiones

Al término de la presente tesis doctoral se dispone de 4 UAVs completamente equipados con el conjunto de elementos descritos a lo largo del presente capítulo, tal y como se muestra con detalle en la Figura B.9, sobre los que se han completado un conjunto de vuelos manuales para comprobación y ajuste de parámetros de vuelo.



Figura B.9: Prototipo demostrador empleado para vuelo autónomo y coordinada en enjambre.

En este capítulo se ha tratado de sintetizar todos los desarrollos realizados para poder construir una arquitectura de hardware que posea, en relación a la arquitectura de enjambre empleada en simulación, el mayor grado de similitud posible, de cara a garantizar y facilitar la implementación de la arquitectura de software presentada a lo largo de la tesis sobre un enjambre de UAVs real.

Junto a todos los elementos de hardware necesarios para el correcto funcionamiento del enjambre durante una navegación autónoma y coordinada, a lo largo del capítulo se ha presentado y detallado el procedimiento desarrollado para poder establecer una red local de comunicación e intercambio de información escalable entre todos los UAVs del enjambre, de cara a poder establecer una arquitectura de hardware completa capaz de ser empleada como demostrador del vuelo en enjambre sin supervisión haciendo uso de la arquitectura de software presentada a lo largo de la tesis doctoral.

Bibliografía

- [1] Abdulla Al-Kaff, Ángel Madridano, Sergio Campos, Fernando García, David Martín, and Arturo de la Escalera. Emergency support unmanned aerial vehicle for forest fire surveillance. *Electronics*, 9(2):260, 2020.
- [2] Ángel Madridano, Abdulla Al-Kaff, David Martín, et al. 3d trajectory planning method for uavs swarm in building emergencies. *Sensors*, 20(3):642, 2020.
- [3] Raúl Sosa San Frutos, Abdulla Al Kaff, Ahmed Hussein, Ángel Madridano, David Martín, and Arturo de la Escalera. Ros-based architecture for multiple unmanned vehicles (uxvs) formation. In *International Conference on Computer Aided Systems Theory*, pages 11–19. Springer, 2019.
- [4] Ángel Madridano, Abdulla Al-Kaff, David Martín, Arturo de la Escalera, and José María Armingol. Método de planificación de trayectorias múltiples para enjambre de uavs. In *XL Jornadas de Automática*, pages 741–748. Universidade da Coruña, Servizo de Publicacións, 2019.
- [5] Ángel Madridano, Abdulla Al-Kaff, David Martín Gómez, and Arturo de la Escalera. Multi-path planning method for uavs swarm purposes. In *2019 IEEE International Conference of Vehicular Electronics and Safety (ICVES)*, pages 1–6. IEEE, 2019.
- [6] Ángel Madridano, Abdulla Al-Kaff, Pablo Flores, David Martín, and Arturo de la Escalera. Obstacle avoidance manager for uavs swarm. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 815–821. IEEE, 2020.
- [7] Ángel Madridano Carrasco, Sergio Campos Novoa, Abdulla Al-Kaff, Fernando García Fernández, David Martín Gómez, and Arturo de la Escalera Hueso. Vehículo aéreo no tripulado para vigilancia y monitorización de incendios. *Revista Iberoamericana de Automática e Informática industrial*, 2020.

-
- [8] Abdulla Al-Kaff, Ángel Madridano, A. Radwan, Francisco Miguel Moreno, Ahmed Hussein, Arturo de la, and Escalera. Heterogeneous multiple vehicles cooperation approach for smart roads. In *2019 International Micro Air Vehicles, Conferences and Competitions (IMAV)*, pages 222–227, 2019.
- [9] Armando Astudillo, Abdulla Al-Kaff, Ángel Madridano, Fernando García, David Martín, and Arturo de la Escalera. Mono-Isde: Lightweight semantic-cnn for depth estimation from monocular aerial images*. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 807–814, 2020.
- [10] Marcos Rodríguez, Abdulla Al-Kaff, Ángel Madridano, David Martín, and Arturo de la Escalera. Wilderness search and rescue with heterogeneous multi-robot systems*. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 110–116, 2020.
- [11] Agencia Estatal de Seguridad Aérea. Registro de comunicaciones previas de operadores de rpa’s. Technical report, 2020.
- [12] Drones rc: Drones Profesionales y Radio Control. ¿qué es un dron? tipos, nombres y componentes. urlhttps://juguetecnic.com/blog/107_que-es-un-drone--tipos--nombres--y--componentes.html, 2020.
- [13] Drone Hopper. Wild hopper - heavy-duty uav for day and night firefighting operations. <https://cordis.europa.eu/project/id/880642/es>, 2019. 21-09-2020.
- [14] CJ Taylor. Robotics: Computational motion planning. <https://www.coursera.org/learn/robotics-motion-planning>, 2018.
- [15] Xia Chen, Guang-yao Li, and Xiang-min Chen. Path planning and cooperative control for multiple uavs based on consistency theory and voronoi diagram. In *2017 29th Chinese Control And Decision Conference (CCDC)*, pages 881–886. IEEE, 2017.
- [16] Wilbert G Aguilar and Stephanie G Morales. 3d environment mapping using the kinect v2 and path planning based on rrt algorithms. *Electronics*, 5(4):70, 2016.
- [17] Kiril Solovey, Oren Salzman, and Dan Halperin. Finding a needle in an exponential haystack: Discrete rrt for exploration of implicit roadmaps in multi-robot motion planning. In *Algorithmic Foundations of Robotics XI*, pages 591–607. Springer, 2015.

- [18] Farid Bounini, Denis Gingras, Herve Pollart, and Dominique Gruyer. Modified artificial potential field method for online path planning applications. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 180–185. IEEE, 2017.
- [19] Multi-Robot Systems Lab. Coordinated agile control of aerial vehicles.
- [20] Jiayi Sun, Jun Tang, and Songyang Lao. Collision avoidance for cooperative uavs with optimized artificial potential field algorithm. *IEEE Access*, 5:18382–18390, 2017.
- [21] Zicong Wu, Weizhou Su, and Junhui Li. Multi-robot path planning based on improved artificial potential field and b-spline curve optimization. In *2019 Chinese Control Conference (CCC)*, pages 4691–4696. IEEE, 2019.
- [22] Christoph Burger and Martin Lauer. Cooperative multiple vehicle trajectory planning using miqp. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 602–607. IEEE, 2018.
- [23] XU Zhen, Zhang Enze, and CHEN Qingwei. Rotary unmanned aerial vehicles path planning in rough terrain based on multi-objective particle swarm optimization. *Journal of Systems Engineering and Electronics*, 31(1):130–141, 2020.
- [24] Hyansu Bae, Gidong Kim, Jonguk Kim, Dianwei Qian, and Sukgyu Lee. Multi-robot path planning method using reinforcement learning. *Applied Sciences*, 9(15):3057, 2019.
- [25] J San-Miguel-Ayanz, T Durrant, R Boca, G Libertà, A Branco, D de Rigo, D Ferrari, P Maianti, TA Vivancos, E Schulte, et al. Forest fires in europe, middle east and north africa. *Publications Office*, 09 2017.
- [26] Abdulla Hussein Al-Kaff. *Vision-Based navigation system for unmanned aerial vehicles*. PhD thesis, Universidad Carlos III de Madrid.
- [27] Stuart I Granshaw. Rpv, uav, uas, rpas or just drone? *The Photogrammetric Record*, 33(162):160–170, 2018.
- [28] Abdulla Al-Kaff, José María Armingol, and Arturo de La Escalera. A vision-based navigation system for unmanned aerial vehicles (uavs). *Integrated Computer-Aided Engineering*, 26(3):297–310, 2019.
- [29] R Bravo and Adriana Leiras. Literature review of the application of uavs in humanitarian relief. *Proceedings of the XXXV Encontro Nacional de Engenharia de Producao, Fortaleza, Brazil*, pages 13–16, 2015.

- [30] OACI Circular. 328, sistemas de aeronaves no tripuladas (uas). *International Civil Aviation Organization*, 2011.
- [31] Boletín Oficial del Estado. Ley 18/2014, de 15 de octubre, de aprobación de medidas urgentes para el crecimiento, la competitividad y la eficiencia. *BOE*, 252:83921–84082, 2014.
- [32] Boletín Oficial del Estado. Real decreto 1036/2017, de 15 de diciembre, por el que se regula la utilización civil de las aeronaves pilotadas por control remoto, y se modifican el real decreto 552/2014, de 27 de junio, por el que se desarrolla el reglamento del aire y disposiciones operativas comunes para los servicios y procedimientos de navegación aérea y el real decreto 57/2002, de 18 de enero, por el que se aprueba el reglamento de circulación aérea. *BOE*, 316:129609–129641, 2017.
- [33] M Herrmann and Esq JD. A comparison of unmanned aerial vehicle regulations in the united states and europe. In *Proceedings of 53rd ASC Annual International Conference, Seattle, Washington*, pages 299–306, 2017.
- [34] Diario Oficial del Unión Europea. Reglamento delegado (ue) 2019/945 de la comisión de 12 de marzo de 2019 sobre los sistemas de aeronaves no tripuladas y los operadores de terceros países de sistemas de aeronaves no tripuladas. *Diario Oficial de la Unión Europea*, 152:1–39, 2019.
- [35] Diario Oficial del Unión Europea. Reglamento de ejecución (ue) 2019/947 de la comisión de 24 de mayo de 2019 relativo a las normas y los procedimientos aplicables a la utilización de aeronaves no tripuladas. *Diario Oficial de la Unión Europea*, 152:45–71, 2019.
- [36] Agencia Estatal de la Seguridad Aérea. Nuevo marco normativo sobre uas/drones, ¿qué debes saber sobre los reglamentos europeos? Documentación pública, Ministerio de transporte, movilidad y agenda urbana, 8 2020.
- [37] JU SESAR. European drones outlook study unlocking the value for europe. *Siebert, JU, Nov*, 2016.
- [38] Ministerio de Fomento. Plan estratégico para el desarrollo del sector civil de los drones en españa. [urlhttps://www.fomento.gob.es/NR/rdonlyres/7B974E30-2BD2-46E5-BEE5-26E00851A455/148411/PlanEstrategicoDrones.pdf](https://www.fomento.gob.es/NR/rdonlyres/7B974E30-2BD2-46E5-BEE5-26E00851A455/148411/PlanEstrategicoDrones.pdf), 2018.
- [39] GV Hoy and B Wang. Forecast: Personal and commercial drones worldwide 2016. *Gartner*, 2016.

- [40] Esticast Research. Commercial drone market, industry trends, estimation studio forecast 2016-2024. *Telecom IT*, 2016.
- [41] Pieter van Lierop, Erik Lindquist, Shiroma Sathyapala, and Gianluca Franceschini. Global forest area disturbance from fire, insect pests, diseases and severe weather events. *Forest Ecology and Management*, 352:78–88, 2015.
- [42] Bruno Chatenoux and Pascal Peduzzi. Biomass fires: preliminary estimation of ecosystems global economic losses. *UNEP/GRID-Geneva*, 2013.
- [43] Fay H Johnston, Sarah B Henderson, Yang Chen, James T Randerson, Miriam Marlier, Ruth S DeFries, Patrick Kinney, David MJS Bowman, and Michael Brauer. Estimated global mortality attributable to smoke from landscape fires. *Environmental health perspectives*, 120(5):695–701, 2012.
- [44] MJ Karter. Fire loss in the united states during 2012. 2013. *National Fire Protection Association: Quincy, MA*, page 56, 2009.
- [45] Pesca y Alimentación. Gboierno de España Ministerio de Agricultura. Los incendios forestales en españa. *Decenio 2006-2015*, 2019.
- [46] Pesca y Alimentación. Gboierno de España Ministerio de Agricultura. Los incendios forestales en españa. *Avance informativo 1 enero - 31 diciembre 2016*, 2019.
- [47] Pesca y Alimentación. Gboierno de España Ministerio de Agricultura. Los incendios forestales en españa. *Avance informativo 1 enero - 31 diciembre 2017*, 2019.
- [48] Pesca y Alimentación. Gboierno de España Ministerio de Agricultura. Los incendios forestales en españa. *Avance informativo 1 enero - 31 diciembre 2018*, 2019.
- [49] Pesca y Alimentación. Gboierno de España Ministerio de Agricultura. Los incendios forestales en españa. *Avance informativo 1 enero - 31 diciembre 2019*, 2019.
- [50] José Luis Liz Graña and Pablo Ferrer Gisbert. Sistema balístico para el control y extinción de incendios forestales en zonas inaccesibles. *Dyna*, 83(5):257–264, 2008.
- [51] Gastón A Addati and Gabriel Pérez Lance. Introducción a los uav's, drones o vants de uso civil. Technical report, Serie Documentos de Trabajo, Universidad del CEMA, 2014.

- [52] Comunidad de Madrid. Los drones y sus aplicaciones a la ingeniería civil. *Madrid: Comunidad de Madrid*, 2015.
- [53] Ashutosh Natraj, Dieu Sang Ly, Damien Eynard, Cédric Demonceaux, and Pascal Vasseur. Omnidirectional vision for uav: Applications to attitude, motion and altitude estimation for day and night conditions. *Journal of Intelligent & Robotic Systems*, 69(1-4):459–473, 2013.
- [54] Thomas Müller. Robust drone detection for day/night counter-uav with static vis and swir cameras. In *Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VIII*, volume 10190, page 1019018. International Society for Optics and Photonics, 2017.
- [55] Wenyuan Kong, Jingyi Cheng, Xin Liu, Fan Zhang, and Teng Fei. Incorporating nocturnal uav side-view images with viirs data for accurate population estimation: a test at the urban administrative district scale. *International Journal of Remote Sensing*, 40(22):8528–8546, 2019.
- [56] Anwasha Borthakur and Pardeep Singh. Drones: new tools for natural risk mitigation and disaster response. *Current Science*, 110(6):958, 2016.
- [57] Borja Sorribas. Aprobado por las autoridades el primer dron autónomo en europa. [urlhttps://www.todrone.com/aprobado-autoridades-primer-dron-autonomo-europa/](https://www.todrone.com/aprobado-autoridades-primer-dron-autonomo-europa/), 2019. Accedido 12-09-2020.
- [58] Randal W Beard and Timothy W McLain. *Small unmanned aircraft: Theory and practice*. Princeton university press, 2012.
- [59] Chun Fui Liew, Danielle DeLatte, Naoya Takeishi, and Takehisa Yairi. Recent developments in aerial robotics: A survey and prototypes overview. *arXiv preprint arXiv:1711.10085*, 2017.
- [60] Peter van Blyenburgh. Uav systems: global review. In *Conference, Amsterdam, The Netherlands*, 2006.
- [61] Konstantinos Dalamagkidis, Kimon P Valavanis, and Les A Piegl. *On integrating unmanned aircraft systems into the national airspace system: issues, challenges, operational restrictions, certification, and recommendations*, volume 54. springer science & Business Media, 2011.
- [62] Dario Floreano and Robert J Wood. Science, technology and the future of small autonomous drones. *Nature*, 521(7553):460–466, 2015.
- [63] Francesco Nex and Fabio Remondino. Uav for 3d mapping applications: a review. *Applied geomatics*, 6(1):1–15, 2014.

- [64] Stuart Young and Alexander Kott. A survey of research on control of teams of small robots in military operations. *arXiv preprint arXiv:1606.01288*, 2016.
- [65] Filippo Gambella, Luisella Sistu, Davide Piccirilli, Sergio Corposanto, Maria Caria, Ettore Arcangeletti, Andrea Rosario Proto, Giovanni Chessa, and Antonio Pazzona. Forest and uav: a bibliometric review. *Contemporary Engineering Sciences*, 9(28):1359–1370, 2016.
- [66] Alena Otto, Niels Agatz, James Campbell, Bruce Golden, and Erwin Pesch. Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: A survey. *Networks*, 72(4):411–458, 2018.
- [67] Christopher Gomez and Heather Purdie. Uav-based photogrammetry and geocomputing for hazards and disaster risk monitoring—a review. *Geoenvironmental Disasters*, 3(1):23, 2016.
- [68] Abdulla Al-Kaff, David Martin, Fernando Garcia, Arturo de la Escalera, and José María Armingol. Survey of computer vision algorithms and applications for unmanned aerial vehicles. *Expert Systems with Applications*, 92:447–463, 2018.
- [69] Francesco Marinello, Andrea Pezzuolo, Alessandro Chiumenti, and Luigi Sartori. Technical analysis of unmanned aerial vehicles (drones) for agricultural applications. *Engineering for Rural Development*, 15, 2016.
- [70] Radek Baranek and František Šolc. Modelling and control of a hexa-copter. In *Proceedings of the 13th International Carpathian Control Conference (ICCC)*, pages 19–23. IEEE, 2012.
- [71] Hyon Lim, Jaemann Park, Daewon Lee, and H Jin Kim. Build your own quadrotor: Open-source projects on unmanned aerial vehicles. *IEEE Robotics & Automation Magazine*, 19(3):33–45, 2012.
- [72] Hyeonbeom Lee, Suseong Kim, Hyon Lim, H Jin Kim, and Daewon Lee. Control of an octa-copter from modeling to experiments. In *IEEE ISR 2013*, pages 1–5. IEEE, 2013.
- [73] Kanaiya Agrawal and Punit Shrivastav. Multi-rotors: A revolution in unmanned aerial vehicle. *International Journal of Science and Research*, 4(11):1800–1804, 2015.
- [74] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, Young-Han Nam, and Mérouane Debbah. A tutorial on uavs for wireless networks: Applications,

- challenges, and open problems. *IEEE Communications Surveys & Tutorials*, 21(3):2334–2360, 2019.
- [75] Hazim Shakhatreh, Ahmad H Sawalmeh, Ala Al-Fuqaha, Zuochoao Dou, Eyad Almaita, Issa Khalil, Noor Shamsiah Othman, Abdallah Khreishah, and Mohsen Guizani. Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges. *Ieee Access*, 7:48572–48634, 2019.
- [76] Gerardo Gutiérrez, Grace Erny, Alyssa Friedman, Melanie Godsey, and Machal Gradoz. Archaeological topography with small unmanned aerial vehicles. *SAA Archaeol. Rec*, 16:10–13, 2016.
- [77] Przemysław Klapa, Piotr Bożek, and Izabela Piech. Charting topographic maps based on uav data using the image classification method. *Geomatics, Landmanagement and Landscape*, 2019.
- [78] Joon Kyu Park and Kap Yong Jung. Construction of 3d geospatial information for development and safety management of open-pit mine. *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, 38(1):43–48, 2020.
- [79] Ademir Marques Junior, Dalva Maria De Castro, Taina Thomassin Guimarães, Leonardo Campos Inocencio, Maurício Roberto Veronez, Frederico Fábio Mauad, and Luis Gonzaga Jr. Statistical assessment of cartographic product from photogrammetry and fixed-wing uav acquisition. *European Journal of Remote Sensing*, 53(1):27–39, 2020.
- [80] Abdulla Al-Kaff, María José Gómez-Silva, Francisco Miguel Moreno, Arturo de la Escalera, and José María Armingol. An appearance-based tracking algorithm for aerial search and rescue purposes. *Sensors*, 19(3):652, 2019.
- [81] Konstantinos Kanistras, Goncalo Martins, Matthew J Rutherford, and Kimon P Valavanis. A survey of unmanned aerial vehicles (uavs) for traffic monitoring. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 221–234. IEEE, 2013.
- [82] Andrey V Savkin and Hailong Huang. Asymptotically optimal deployment of drones for surveillance and monitoring. *Sensors*, 19(9):2068, 2019.
- [83] Yue Li, Yongmin Zhang, and Lin Cai. Optimal location of supplementary node in uav surveillance system. *Journal of Network and Computer Applications*, 140:23–39, 2019.

- [84] Abdulla Al-Kaff, Francisco Miguel Moreno, Arturo de la Escalera, and José María Armingol. Intelligent vehicle for search, rescue and transportation purposes. In *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pages 110–115. IEEE, 2017.
- [85] Mirela Kundid Vasić and Vladan Papić. Multimodel deep learning for person detection in aerial images. *Electronics*, 9(9):1459, 2020.
- [86] Mario Silvagni, Andrea Tonoli, Enrico Zenerino, and Marcello Chiaberge. Multipurpose uav for search and rescue operations in mountain avalanche events. *Geomatics, Natural Hazards and Risk*, 8(1):18–33, 2017.
- [87] Marcelo S Menegol, Jomi F Hübner, and Leandro B Becker. Coordinated uav search and rescue application with jacamo. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 335–338. Springer, 2018.
- [88] Adiel Ismail, Emmanuel Tuyishimire, and Antoine Bagula. Generating dubins path for fixed wing uavs in search missions. In *International Symposium on Ubiquitous Networking*, pages 347–358. Springer, 2018.
- [89] FJ Perez-Grau, R Ragel, F Caballero, A Viguria, and A Ollero. Semi-autonomous teleoperation of uavs in search and rescue scenarios. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1066–1074. IEEE, 2017.
- [90] Abdulla Al-Kaff, Francisco Miguel Moreno, Luis Javier San José, Fernando García, David Martín, Arturo de la Escalera, Alberto Nieva, and José Luis Meana García. Vbii-uav: Vision-based infrastructure inspection-uav. In *World Conference on Information Systems and Technologies*, pages 221–231. Springer, 2017.
- [91] Milan Banić, Aleksandar Miltenović, Milan Pavlović, and Ivan Ćirić. Intelligent machine vision based railway infrastructure inspection and monitoring using uav. *Facta Universitatis, Series: Mechanical Engineering*, 17(3):357–364, 2019.
- [92] Sungsook Choi and Eungkon Kim. Image acquisition system for construction inspection based on small unmanned aerial vehicle. In *Advanced Multimedia and Ubiquitous Engineering*, pages 273–280. Springer, 2015.
- [93] P Ramon-Soria, M Perez-Jimenez, BC Arrue, and A Ollero. Planning system for integrated autonomous infrastructure inspection using uavs. In *2019*

- International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 313–320. IEEE, 2019.
- [94] Nan Zhao, Weidang Lu, Min Sheng, Yunfei Chen, Jie Tang, F Richard Yu, and Kai-Kit Wong. Uav-assisted emergency networks in disasters. *IEEE Wireless Communications*, 26(1):45–51, 2019.
- [95] Tao Yang, Peiqi Li, Huiming Zhang, Jing Li, and Zhi Li. Monocular vision slam-based uav autonomous landing in emergencies and unknown environments. *Electronics*, 7(5):73, 2018.
- [96] Kirtan Gopal Panda, Shrayan Das, Debarati Sen, and Wasim Arif. Design and deployment of uav-aided post-disaster emergency network. *IEEE Access*, 7:102985–102999, 2019.
- [97] Soumya Kanti Datta, Jean-Luc Dugelay, and Christian Bonnet. Iot based uav platform for emergency services. In *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 144–147. IEEE, 2018.
- [98] Ilias Gkotsis, Agavi-Christina Kousouraki, George Eftychidis, Panayiotis Koliou, and Maria Terzi. Swarm of uavs as an emergency response technology. In *Risk Analysis Based on Data and Crisis Response Beyond Knowledge: Proceedings of the 7th International Conference on Risk Analysis and Crisis Response (RACR 2019), October 15-19, 2019, Athens, Greece*, page 353. CRC Press, 2019.
- [99] Gábor Milics. Application of uavs in precision agriculture. In *International Climate Protection*, pages 93–97. Springer, 2019.
- [100] Dimosthenis C Tsouros, Stamatia Bibi, and Panagiotis G Sarigiannidis. A review on uav-based applications for precision agriculture. *Information*, 10(11):349, 2019.
- [101] UM Rao Mogili and BBVL Deepak. Review on application of drone systems in precision agriculture. *Procedia computer science*, 133:502–509, 2018.
- [102] Daeil Jo and Yongjin Kwon. Development of rescue material transport uav (unmanned aerial vehicle). *World Journal of Engineering and Technology*, 5(4):720–729, 2017.
- [103] Fadi Al-Turjman and Sinem Alturjman. 5g/iot-enabled uavs for multimedia delivery in industry-oriented applications. *Multimedia Tools and Applications*, 79(13):8627–8648, 2020.

-
- [104] Byung Duk Song, Kyungsu Park, and Jonghoe Kim. Persistent uav delivery logistics: Milp formulation and efficient heuristic. *Computers & Industrial Engineering*, 120:418–428, 2018.
- [105] Lucintel Brief. Growth opportunity in global uav market. *Las Colinas, USA*, 2011.
- [106] Drone Hopper. Drone hopper s.l - vivero de empresas uc3m. <https://drone-hopper.com/>, 2016. 21-09-2020.
- [107] Pablo Flores. Non-manned vehicle for extinguishing fires, 3 2017.
- [108] P. Singh, R. Tiwari, and M. Bhattacharya. Navigation in multi robot system using cooperative learning: A survey. In *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, pages 145–150, 2016.
- [109] Aufar Zakiev, Tatyana Tsoy, and Evgeni Magid. Swarm robotics: Remarks on terminology and classification. In *International Conference on Interactive Collaborative Robotics*, pages 291–300. Springer, 2018.
- [110] Erol Şahin. Swarm robotics: From sources of inspiration to domains of application. In *International workshop on swarm robotics*, pages 10–20. Springer, 2004.
- [111] Kelin Jose and Dilip Kumar Pratihari. Task allocation and collision-free path planning of centralized multi-robots system for industrial plant inspection using heuristic methods. *Robotics and Autonomous Systems*, 80:34–42, 2016.
- [112] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. Sampling-based multi-robot exploration. In *ISR/ROBOTIK*, pages 1–6, 2010.
- [113] Zhi Yan, Nicolas Jouandeau, and Arab Ali-Chérif. Multi-robot heuristic goods transportation. In *Intelligent Systems (IS), 2012 6th IEEE International Conference*, pages 409–414. Citeseer, 2012.
- [114] Ryan Luna and Kostas E Bekris. Efficient and complete centralized multi-robot path planning. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 3268–3275. IEEE, 2011.
- [115] Filippo Arrichiello. Coordination control of multiple mobile robots. *UNIVERSITA DEGLI STUDI DI CASSINO*, 2006.

-
- [116] Fethi Matoui, Boumedyen Boussaid, Brahim Metoui, and Mohamed Naceur Abdelkrim. Contribution to the path planning of a multi-robot system: centralized architecture. *Intelligent Service Robotics*, 13(1):147–158, 2020.
- [117] Christopher Amato, George Konidaris, Jonathan P How, and Leslie P Kaelbling. Decentralized decision-making under uncertainty for multi-robot teams. In *the Workshop on Future of Multiple Robot Research and its Multiple Identitie at IROS. IEEE*, 2014.
- [118] Christopher Amato, George Konidaris, Gabriel Cruz, Christopher A Maynor, Jonathan P How, and Leslie P Kaelbling. Planning for decentralized control of multiple robots under uncertainty. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1241–1248. IEEE, 2015.
- [119] Christopher Amato, George Konidaris, Shayegan Omidshafiei, Ali-akbar Agha-mohammadi, Jonathan P How, and Leslie P Kaelbling. Probabilistic planning for decentralized multi-robot systems. In *2015 AAAI Fall Symposium Series*, 2015.
- [120] Shayegan Omidshafiei, Ali-Akbar Agha-Mohammadi, Christopher Amato, Shih-Yuan Liu, Jonathan P How, and John Vian. Decentralized control of multi-robot partially observable markov decision processes using belief space macro-actions. *The International Journal of Robotics Research*, 36(2):231–258, 2017.
- [121] Pinxin Long, Tingxiang Fanl, Xinyi Liao, Wenxi Liu, Hao Zhang, and Jia Pan. Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6252–6259. IEEE, 2018.
- [122] Y Uny Cao, Alex S Fukunaga, and Andrew Kahng. Cooperative mobile robotics: Antecedents and directions. *Autonomous robots*, 4(1):7–27, 1997.
- [123] Golnaz Habibi, Zachary Kingston, William Xie, Mathew Jellins, and James McLurkin. Distributed centroid estimation and motion controllers for collective transport by multi-robot systems. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1282–1288. IEEE, 2015.
- [124] Golnaz Habibi, William Xie, Mathew Jellins, and James McLurkin. Distributed path planning for collective transport using homogeneous multi-robot systems. In *Distributed Autonomous Robotic Systems*, pages 151–164. Springer, 2016.

-
- [125] Jens Wawerla and Richard T Vaughan. A fast and frugal method for team-task allocation in a multi-robot transportation system. In *ICRA*, pages 1432–1437, 2010.
- [126] Jason Gregory, Jonathan Fink, Ethan Stump, Jeffrey Twigg, John Rogers, David Baran, Nicholas Fung, and Stuart Young. Application of multi-robot systems to disaster-relief scenarios with limited communication. In *Field and Service Robotics*, pages 639–653. Springer, 2016.
- [127] Neil Mathew, Stephen L Smith, and Steven L Waslander. Optimal path planning in cooperative heterogeneous multi-robot delivery systems. In *Algorithmic Foundations of Robotics XI*, pages 407–423. Springer, 2015.
- [128] Juan Jesús Roldán, Pablo Garcia-Aunon, Mario Garzón, Jorge de León, Jaime del Cerro, and Antonio Barrientos. Heterogeneous multi-robot system for mapping environmental variables of greenhouses. *Sensors*, 16(7):1018, 2016.
- [129] Yingjiang Zhou, Xinghuo Yu, Changyin Sun, and Wenwu Yu. Higher order finite-time consensus protocol for heterogeneous multi-agent systems. *International Journal of Control*, 88(2):285–294, 2015.
- [130] Carla Mouradian, Jagruti Sahoo, Roch H Glitho, Monique J Morrow, and Paul A Polakos. A coalition formation algorithm for multi-robot task allocation in large-scale natural disasters. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International*, pages 1909–1914. IEEE, 2017.
- [131] Yanjiang Huang, Ryosuke Chiba, Tamio Arai, Tsuyoshi Ueyama, and Jun Ota. Robust multi-robot coordination in pick-and-place tasks based on part-dispatching rules. *Robotics and Autonomous Systems*, 64:70–83, 2015.
- [132] Micael S Couceiro, David Portugal, João F Ferreira, and Rui P Rocha. Sem-fire: Towards a new generation of forestry maintenance multi-robot systems. In *2019 IEEE/SICE International Symposium on System Integration (SII)*, pages 270–276. IEEE, 2019.
- [133] Nicholas Fung, John Rogers, Carlos Nieto, Henrik I Christensen, Stephanie Kemna, and Gaurav Sukhatme. Coordinating multi-robot systems through environment partitioning for adaptive informative sampling. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3231–3237. IEEE, 2019.

-
- [134] Javier Tardós, Rosario Aragues, Carlos Sagüés, and Carlos Rubio. Simultaneous deployment and tracking multi-robot strategies with connectivity maintenance. *Sensors*, 18(3):927, 2018.
- [135] William Marchant and Sabri Tosunoglu. Rethinking wildfire suppression with swarm robotics. In *Proceedings of the 29th Florida Conference on Recent Advances in Robotics, FCRAR*, pages 12–13, 2016.
- [136] Mauro S Innocente and Paolo Grasso. Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems. *Journal of Computational Science*, 34:80–101, 2019.
- [137] M Bakhshipour, M Jabbari Ghadi, and F Namdari. Swarm robotics search & rescue: A novel artificial intelligence-inspired optimization approach. *Applied Soft Computing*, 57:708–726, 2017.
- [138] Maria Terzi, Antreas Anastasiou, Panayiotis Kolios, Christos Panayiotou, and Theocharis Theocharides. Swifsters: A multi-uav platform for disaster management. In *2019 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*, pages 1–7. IEEE, 2019.
- [139] Zhenhui Yuan, Jie Jin, Lingling Sun, Kwan-Wu Chin, and Gabriel-Miro Muntean. Ultra-reliable iot communications with uavs: A swarm use case. *IEEE Communications Magazine*, 56(12):90–96, 2018.
- [140] Muhammad Yeasir Arafat and Sangman Moh. Localization and clustering based on swarm intelligence in uav networks for emergency communications. *IEEE Internet of Things Journal*, 6(5):8958–8976, 2019.
- [141] Mitch Champion, Prakash Ranganathan, and Saleh Faruque. A review and future directions of uav swarm communication architectures. In *2018 IEEE International Conference on Electro/Information Technology (EIT)*, pages 0903–0908. IEEE, 2018.
- [142] Ricardo Cajo, Thoa Mac Thi, Cosmin Copot, Douglas Plaza, Robain De Keyser, and Clara Ionescu. Multiple uavs formation for emergency equipment and medicines delivery based on optimal fractional order controllers. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 318–323. IEEE, 2019.
- [143] Ana Maria Cruz-Martin et al. *Planificación de trayectorias en sistemas multirrobot*. PhD thesis, Departamento de Ingeniería de Sistemas y Automática. Universidad de Málaga, 12 2004.

-
- [144] Chad Goerzen, Zhaodan Kong, and Bernard Mettler. A survey of motion planning algorithms from the perspective of autonomous uav guidance. *Journal of Intelligent and Robotic Systems*, 57(1-4):65, 2010.
- [145] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.
- [146] H Cormen Thomas, E Leiserson Charles, L Rivest Ronald, and Stein Clifford. Section 24.3: Dijkstra’s algorithm. In *Introduction to Algorithms*, pages 595–601. MIT Press and McGraw-Hill, 2001.
- [147] Thi Thoa Mac, Cosmin Copot, Duc Trung Tran, and Robin De Keyser. A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization. *Applied Soft Computing*, 59:68–76, 2017.
- [148] Xiaoshan Bai, Weisheng Yan, Ming Cao, and Dong Xue. Distributed multi-vehicle task assignment in a time-invariant drift field with obstacles. *IET Control Theory & Applications*, 13(17):2886–2893, 2019.
- [149] Xiao Chen, Xin Zhang, Weizhao Huang, Shungui Liu, and Hao Dai. Coordinated optimal path planning of multiple substation inspection robots based on conflict detection. In *2019 Chinese Automation Congress (CAC)*, pages 5069–5074. IEEE, 2019.
- [150] D. Shiyong, Z. Xiaoping, and L. Guoqing. Cooperative planning method for swarm uavs based on hierarchical strategy. In *2012 3rd International Conference on System Science, Engineering Design and Manufacturing Informatization*, volume 2, pages 304–307, 2012.
- [151] Alexander Erokhin, Vladimir Erokhin, Sergey Sotnikov, and Anatoly Gogolevsky. Optimal multi-robot path finding algorithm based on a. In *Proceedings of the Computational Methods in Systems and Software*, pages 172–182. Springer, 2018.
- [152] Anh Vu Le, Veerajagadheswar Prabakaran, Vinu Sivanantham, and Ramesh Elara Mohan. Modified a-star algorithm for efficient coverage path planning in tetris inspired self-reconfigurable robot with integrated laser sensor. *Sensors*, 18(8):2585, 2018.
- [153] Guibin Sun, Rui Zhou, Bin Di, Zhuoning Dong, and Yingxun Wang. A novel cooperative path planning for multi-robot persistent coverage with obstacles and coverage period constraints. *Sensors*, 19(9):1994, 2019.

-
- [154] Zahra Boroujeni, Daniel Goehring, Fritz Ulbrich, Daniel Neumann, and Raul Rojas. Flexible unit a-star trajectory planning for autonomous vehicles on structured road maps. In *Vehicular Electronics and Safety (ICVES), 2017 IEEE International Conference on*, pages 7–12. IEEE, 2017.
- [155] Anthony Stentz. Optimal and efficient path planning for partially known environments. In *Intelligent Unmanned Ground Vehicles*, pages 203–220. Springer, 1997.
- [156] Jung-Hao Peng, I-Hsum Li, Yi-Hsing Chien, Chen-Chien Hsu, and Wei-Yen Wang. Multi-robot path planning based on improved d* lite algorithm. In *2015 IEEE 12th International Conference on Networking, Sensing and Control*, pages 350–353. IEEE, 2015.
- [157] Ellips Masehian and Davoud Sedighzadeh. A multi-objective pso-based algorithm for robot path planning. In *2010 IEEE International Conference on Industrial Technology*, pages 465–470. IEEE, 2010.
- [158] Jianfeng Wang, Guangliang Chang, Weihua Li, Na Yang, Boqian Wang, and Yuhui Sun. Improving lateral safety distance-based on feature detection and probabilistic roadmaps for unmanned vehicle path planning. *research square*, 2020.
- [159] Christian Zammit and Erik-Jan Van Kampen. Comparison between a* and rrt algorithms for uav path planning. In *2018 AIAA guidance, navigation, and control conference*, page 1846, 2018.
- [160] James A Preiss, Wolfgang Hönig, Nora Ayanian, and Gaurav S Sukhatme. Downwash-aware trajectory planning for large quadrotor teams. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 250–257. IEEE, 2017.
- [161] Mark Debord, Wolfgang Hönig, and Nora Ayanian. Trajectory planning for heterogeneous robot teams. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7924–7931. IEEE, 2018.
- [162] Wolfgang Hönig, James A Preiss, TK Satish Kumar, Gaurav S Sukhatme, and Nora Ayanian. Trajectory planning for quadrotor swarms. *IEEE Transactions on Robotics*, 34(4):856–869, 2018.
- [163] N. Mansard, A. DelPrete, M. Geisert, S. Tonneau, and O. Stasse. Using a memory of motion to efficiently warm-start a nonlinear predictive controller. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2986–2993, 2018.

-
- [164] W. Meng, Z. He, R. Su, P. K. Yadav, R. Teo, and L. Xie. Decentralized multi-uav flight autonomy for moving convoys search and track. *IEEE Transactions on Control Systems Technology*, 25(4):1480–1487, 2017.
- [165] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10(12):399, 2013.
- [166] Mert Turanli and Hakan Temeltas. Adaptive coverage control with guaranteed power voronoi diagrams. In *2017 4th International Conference on Systems and Informatics (ICSAI)*, pages 7–13. IEEE, 2017.
- [167] Hong-Xing Wei, Qiang Mao, Yong Guan, and Yong-Dong Li. A centroidal voronoi tessellation based intelligent control algorithm for the self-assembly path planning of swarm robots. *Expert Systems with Applications*, 85:261–269, 2017.
- [168] Carlos Sampedro, Hriday Bavle, Jose Luis Sanchez-Lopez, Ramon A Suárez Fernández, Alejandro Rodríguez-Ramos, Martin Molina, and Pascual Campoy. A flexible and dynamic mission planning architecture for uav swarm coordination. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 355–363. IEEE, 2016.
- [169] Khoulood Eledlebi, Hanno Hildmann, Dymitr Ruta, and AF Isakovic. A hybrid voronoi tessellation/genetic algorithm approach for the deployment of drone-based nodes of a self-organizing wireless sensor network (wsn) in unknown and gps denied environments. *Drones*, 4(3):33, 2020.
- [170] Liwei Huang, Hong Qu, Peng Ji, Xintong Liu, and Zhen Fan. A novel coordinated path planning method using k-degree smoothing for multi-uavs. *Applied Soft Computing*, 48:182–192, 2016.
- [171] X. Chen, G. Li, and X. Chen. Path planning and cooperative control for multiple uavs based on consistency theory and voronoi diagram. In *2017 29th Chinese Control And Decision Conference (CCDC)*, pages 881–886, 2017.
- [172] Jeongeun Kim and Hyoung Il Son. A voronoi diagram-based workspace partition for weak cooperation of multi-robot system in orchard. *IEEE Access*, 8:20676–20686, 2020.
- [173] Rongxin Cui, Yang Li, and Weisheng Yan. Mutual information-based multi-auv path planning for scalar field sampling using multidimensional rrt. *IEEE*

- Transactions on Systems, Man, and Cybernetics: Systems*, 46(7):993–1004, 2016.
- [174] Denggui Zhang, Yong Xu, and Xingting Yao. An improved path planning algorithm for unmanned aerial vehicle based on rrt-connect. In *2018 37th Chinese Control Conference (CCC)*, pages 4854–4858. IEEE, 2018.
- [175] Yoann Solana, Michele Furci, Juan Cortés, and Antonio Franchi. Multi-robot path planning with maintenance of generalized connectivity. In *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 63–70. IEEE, 2017.
- [176] Liang Yang, Juntong Qi, Dalei Song, Jizhong Xiao, Jianda Han, and Yong Xia. Survey of robot 3d path planning algorithms. *Journal of Control Science and Engineering*, 2016, 2016.
- [177] Zhang Ying and Li Xu. Leader-follower formation control and obstacle avoidance of multi-robot based on artificial potential field. In *Control and Decision Conference (CCDC), 2015 27th Chinese*, pages 4355–4360. IEEE, 2015.
- [178] Abdelrahman M Hassan, Catherine M Elias, Omar M Shehata, and Elsaayed I Morgan. A global integrated artificial potential field/virtual obstacles path planning algorithm for multi-robot system applications. *Int Res J Eng Technol*, 4:1198–1204, 2017.
- [179] Byung Duk Song, Jonghoe Kim, and James R Morrison. Rolling horizon path planning of an autonomous system of uavs for persistent cooperative service: Milp formulation and efficient heuristics. *Journal of Intelligent & Robotic Systems*, 84(1-4):241–258, 2016.
- [180] Ratan Lal, Ajay Sharda, and Pavithra Prabhakar. Optimal multi-robot path planning for pesticide spraying in agricultural fields. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 5815–5820. IEEE, 2017.
- [181] Rafael Lazimy. Mixed-integer quadratic programming. *Mathematical Programming*, 22(1):332–349, 1982.
- [182] Daniel Mellinger, Alex Kushleyev, and Vijay Kumar. Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 477–483. IEEE, 2012.

- [183] Alex Kushleyev, Daniel Mellinger, Caitlin Powers, and Vijay Kumar. Towards a swarm of agile micro quadrotors. *Autonomous Robots*, 35(4):287–300, 2013.
- [184] Vojtech Spurny, Tomas Baca, and Martin Saska. Complex manoeuvres of heterogeneous mav-ugv formations using a model predictive control. In *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 998–1003. IEEE, 2016.
- [185] Carlos E Luis, Marijan Vukosavljev, and Angela P Schoellig. Online trajectory generation with distributed model predictive control for multi-robot motion planning. *IEEE Robotics and Automation Letters*, 5(2):604–611, 2020.
- [186] Nathan Ratliff, Matt Zucker, J Andrew Bagnell, and Siddhartha Srinivasa. Chomp: Gradient optimization techniques for efficient motion planning. In *2009 IEEE International Conference on Robotics and Automation*, pages 489–494. IEEE, 2009.
- [187] Jennifer David, Rafael Valencia, and Karl Iagnemma. Task assignment and trajectory planning in dynamic environments for multiple vehicles. In *RSS 2016 Workshop on Task and Motion Planning, Ann Arbor, Michigan, USA, June 19, 2016*, 2016.
- [188] María A Guzmán, Cristian Peña, et al. Algoritmos bioinspirados en la planeación off-line de trayectorias de robots seriales. *Visión electrónica*, 7(1):27–39, 2013.
- [189] Thomas Back. *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.
- [190] Zengliang Han, Dongqing Wang, Feng Liu, and Zhiyong Zhao. Multi-agv path planning with double-path constraints by using an improved genetic algorithm. *PloS one*, 12(7), 2017.
- [191] Milad Nazarahari, Esmaeel Khanmirza, and Samira Doostie. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. *Expert Systems with Applications*, 115:106–120, 2019.
- [192] Roy Glasius, Andrzej Komoda, and Stan CAM Gielen. Neural network dynamics for path planning and obstacle avoidance. *Neural Networks*, 8(1):125–133, 1995.

-
- [193] Maxim Lapan. *Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q-networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more*. Packt Publishing Ltd, 2018.
- [194] Han Qie, Dianxi Shi, Tianlong Shen, Xinhai Xu, Yuan Li, and Liuqing Wang. Joint optimization of multi-uav target assignment and path planning based on multi-agent reinforcement learning. *IEEE Access*, 7:146264–146272, 2019.
- [195] David Luviano Cruz and Wen Yu. Path planning of multi-agent systems in unknown environment with neural kernel smoothing and reinforcement learning. *Neurocomputing*, 233:34–42, 2017.
- [196] Ahmed M Affi, Omar H Alhosainy, Catherine M Elias, Omar M Shehata, and Elsayed I Morgan. Deep policy-gradient based path planning and reinforcement cooperative q-learning behavior of multi-vehicle systems. In *2019 IEEE International Conference of Vehicular Electronics and Safety (ICVES)*, pages 1–7. IEEE, 2019.
- [197] Mehmet Korkmaz and Akif Durdu. Comparison of optimal path planning algorithms. In *2018 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*, pages 255–258. IEEE, 2018.
- [198] Robert Martin C Santiago, Anton Louise De Ocampo, Aristotle T Ubando, Argel A Bandala, and Elmer P Dadios. Path planning for mobile robots using genetic algorithm and probabilistic roadmap. In *2017 IEEE 9th international conference on humanoid, nanotechnology, information technology, communication and control, environment and management (HNICEM)*, pages 1–5. IEEE, 2017.
- [199] Jonathan Bennett. *OpenStreetMap*. Packt Publishing Ltd, 2010.
- [200] Jose F Pertusa Grau. *Técnicas de análisis de imagen: Aplicaciones en Biología*, volume 65. Universitat de València, 2003.
- [201] Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 2013. Software available at <http://octomap.github.com>.
- [202] Sining Yang, Shaowu Yang, and Xiaodong Yi. An efficient spatial representation for path planning of ground robots in 3d environments. *IEEE Access*, 6:41539–41550, 2018.

- [203] T Knerr. Osm2world create 3d models from openstreetmap. <http://osm2world.org/>, 2019. 07-11-2020.
- [204] Rudolph Triebel, Patrick Pfaff, and Wolfram Burgard. Multi-level surface maps for outdoor terrain mapping and loop closing. In *2006 IEEE/RSJ international conference on intelligent robots and systems*, pages 2276–2282. IEEE, 2006.
- [205] Jari P Saarinen, Henrik Andreasson, Todor Stoyanov, and Achim J Lilienthal. 3d normal distributions transform occupancy maps: An efficient representation for mapping in dynamic environments. *The International Journal of Robotics Research*, 32(14):1627–1644, 2013.
- [206] Todor Stoyanov, Martin Magnusson, Henrik Andreasson, and Achim J Lilienthal. Path planning in 3d environments using the normal distributions transform. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3263–3268. IEEE, 2010.
- [207] Friedrich Fraundorfer, Lionel Heng, Dominik Honegger, Gim Hee Lee, Lorenz Meier, Petri Tanskanen, and Marc Pollefeys. Vision-based autonomous mapping and exploration using a quadrotor mav. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4557–4564. IEEE, 2012.
- [208] Nathaniel Fairfield, George Kantor, and David Wettergreen. Real-time slam with octree evidence grids for exploration in underwater tunnels. *Journal of Field Robotics*, 24(1-2):03–21, 2007.
- [209] René Prissang. Three-dimensional predictive deposit modelling based on the linear octree data structure. In *Computer Graphics in Geology*, pages 229–238. Springer, 1992.
- [210] Carlos Saona-Vazquez, Isabel Navazo, and Pere Brunet. The visibility octree: a data structure for 3d navigation. *Computers & Graphics*, 23(5):635–643, 1999.
- [211] Fakir S. Nooruddin and Greg Turk. Simplification and repair of polygonal models using volumetric techniques. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):191–205, 2003.
- [212] Patrick Min. binvox. <http://www.patrickmin.com/binvox> or <https://www.google.com/search?q=binvox>, 2004 - 2019. Accessed: yyyy-mm-dd.

- [213] Alaa Khamis, Ahmed Hussein, and Ahmed Elmogy. Multi-robot task allocation: A review of the state-of-the-art. In *Cooperative Robots and Sensor Networks 2015*, pages 31–51. Springer, 2015.
- [214] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics (NRL)*, 52(1):7–21, 2005.
- [215] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.
- [216] Smriti Chopra, Giuseppe Notarstefano, Matthew Rice, and Magnus Egersstedt. A distributed version of the hungarian method for multirobot assignment. *IEEE Transactions on Robotics*, 33(4):932–947, 2017.
- [217] G. Cox and G. Cox. *Combustion Fundamentals of Fire*. Combustion treatise. Academic, 1995.
- [218] J Arnaldos, A Giménez, X Navalón, E Pastor, E Planas, and L Zárate. Manual d'enginyeria per a la prevenció i extinció d'incendis forestals. *Centre d'Estudis del Risc Tecnològic, Barcelona*, 2003.
- [219] Jean-Luc Dupuy and Michel Larini. Fire spread through a porous forest fuel bed: a radiative and convective model including fire-induced flow effects. *International Journal of Wildland Fire*, 9(3):155–172, 1999.
- [220] Luís Guadalupe Zárate López. *Estudio de las características físicas y geométricas de la llama en los incendios forestales*. Universitat Politècnica de Catalunya, 2004.
- [221] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.
- [222] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2149–2154. IEEE, 2004.
- [223] Abdulla Al-Kaff, Francisco Miguel Moreno, and Ahmed Hussein. Ros-based approach for unmanned vehicles in civil applications. In *Robot Operating System (ROS)*, pages 155–183. Springer, 2019.

- [224] Abdulla Al-Kaff, Ricardo Alonso, Mostafa Osman, and Ahmed Hussein. Skyonyx: Autonomous uav research platform for air transportation system (atsys). In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3550–3555. IEEE, 2018.
- [225] Lorenz Meier, Petri Tanskanen, Friedrich Fraundorfer, and Marc Pollefeys. Pixhawk: A system for autonomous flight using onboard computer vision. In *2011 IEEE International Conference on Robotics and Automation*, pages 2992–2997. IEEE, 2011.
- [226] Sukhrob Atoev, Ki-Ryong Kwon, Suk-Hwan Lee, and Kwang-Seok Moon. Data analysis of the mavlink communication protocol. In *2017 International Conference on Information Science and Communications Technologies (ICISCT)*, pages 1–3. IEEE, 2017.
- [227] Anis Koubaa, Azza Allouch, Maram Alajlan, Yasir Javed, Abdelfettah Belghith, and Mohamed Khalgui. Micro air vehicle link (mavlink) in a nutshell: A survey. *IEEE Access*, 7:87658–87680, 2019.
- [228] L Meier. Px4 firmware. [urlhttps://px4.io/](https://px4.io/), 2020. 01-10-2020.
- [229] Andrea Maccapani and Maurizio Goiak. Detect and avoid: an industrial perspective technical and certification considerations for automatism and safety functions associated with unmanned aircraft systems. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 68–73. IEEE, 2018.
- [230] Ema Falomir, Serge Chaumette, and Gilles Guerrini. A 3d mobility model for autonomous swarms of collaborative uavs. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 196–204. IEEE, 2019.
- [231] Jialong Zhang, Jianguo Yan, Xiaojun Xing, Dongli Yuan, Xiaolei Hou, Xiaoqiao Qi, and Pu Zhang. The collision avoidance control algorithm of the uav formation flight. In *Proc. ARAA*, pages 1–7, 2017.
- [232] Chee Yong Tan, Sunan Huang, Kok Kiong Tan, and Rodney Swee Huat Teo. Three-dimensional collision avoidance design on unmanned aerial vehicle. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 521–530. IEEE, 2018.
- [233] Marc Ille and Toru Namerikawa. Collision avoidance between multi-uav-systems considering formation control using mpc. In *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 651–656. IEEE, 2017.

- [234] Jiannan Zhao, Cheng Hu, Chun Zhang, Zhihua Wang, and Shigang Yue. A bio-inspired collision detector for small quadcopter. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2018.
- [235] Abdulla Al-Kaff, Fernando García, David Martín, Arturo De La Escalera, and José María Armingol. Obstacle detection and avoidance system based on monocular camera and size expansion algorithm for uavs. *Sensors*, 17(5):1061, 2017.
- [236] Liang Lu, Carlos Sampedro, Javier Rodriguez-Vazquez, and Pascual Campoy. Laser-based collision avoidance and reactive navigation using rrt* and signed distance field for multicopter uavs. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1209–1217. IEEE, 2019.
- [237] Catrina Lim, Boyang Li, Ee Meng Ng, Xin Liu, and Kin Huat Low. Three-dimensional (3d) dynamic obstacle perception in a detect-and-avoid framework for unmanned aerial vehicles. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 996–1004. IEEE, 2019.
- [238] Erwin Perez, Alexander Winger, Alexander Tran, Carlos Garcia-Paredes, Niran Run, Nick Keti, Subodh Bhandari, and Amar Raheja. Autonomous collision avoidance system for a multicopter using stereoscopic vision. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 579–588. IEEE, 2018.
- [239] JinWen Hu, Man Wang, ChunHui Zhao, Quan Pan, and Chang Du. Formation control and collision avoidance for multi-uav systems based on voronoi partition. *Science China Technological Sciences*, 63(1):65–72, 2020.
- [240] Woo-Hyun Ko and PR Kumar. Probability-based collision detection and resolution of planned trajectories for unmanned aircraft system traffic management. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 946–951. IEEE, 2019.
- [241] Tully Foote. tf: The transform library. In *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*, pages 1–6. IEEE, 2013.
- [242] William H Harman. Tcas- a system for preventing midair collisions. *The Lincoln Laboratory Journal*, 2(3):437–457, 1989.
- [243] Azad Ghaffari. Analytical design and experimental verification of geofencing control for aerial applications. *IEEE/ASME Transactions on Mechatronics*, 2020.

- [244] Elie Hermand, Tam W Nguyen, Mehdi Hosseinzadeh, and Emanuele Garone. Constrained control of uavs in geofencing applications. In *2018 26th Mediterranean Conference on Control and Automation (MED)*, pages 217–222. IEEE, 2018.
- [245] Mariem Maiouak and Tarik Taleb. Dynamic maps for automated driving and uav geofencing. *IEEE Wireless Communications*, 26(4):54–59, 2019.
- [246] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- [247] Alexander L Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L Littman. Pac model-free reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 881–888, 2006.
- [248] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.
- [249] Martin Riedmiller. Neural fitted q iteration—first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pages 317–328. Springer, 2005.
- [250] Seyed Sajad Mousavi, Michael Schukat, and Enda Howley. Deep reinforcement learning: an overview. In *Proceedings of SAI Intelligent Systems Conference*, pages 426–440. Springer, 2016.
- [251] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.
- [252] Antonio Gulli and Sujit Pal. *Deep learning with Keras*. Packt Publishing Ltd, 2017.
- [253] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [254] C. Justo, A. Al-Kaff, F. Moreno and Á. Madridano, and J.M. Armingol. Intelligent cooperative system for traffic monitoring in smart cities*. In *2020 Intelligent Vehicles Symposium (IV)*, 2020.
- [255] Derrick Downing and Stephen Ganley. Micropilot autopilot. 2015.

- [256] Richard Bellman. A markovian decision process. *Journal of mathematics and mechanics*, pages 679–684, 1957.
- [257] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017.

