

Jornadas de Automática

Planificación de trayectorias 4D para UAVs basada en Fast Marching Square

López, Blanca.^{1,*}, Moreno, Luis.¹

RoboticsLab, Departamento de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, Avenida de la Universidad, nº30, 28911, Leganés, España.

To cite this article: López, B., Moreno, L. 2024. 4D Trajectory Planning for UAVs using Fast Marching Square. *Jornadas de Automática*, 45. <https://doi.org/10.17979/ja-cea.2024.45.10786>

Resumen

Los enjambres de vehículos aéreos no tripulados (UAVs) se están convirtiendo en una herramienta fundamental en diversos campos de aplicación, tales como la vigilancia aérea, la búsqueda y rescate en zonas urbanas y la entrega de mercancía. En este marco, la planificación de trayectorias sigue siendo una de las principales preocupaciones en aplicaciones robóticas complejas. Este trabajo presenta un algoritmo de planificación de rutas 4D, basado en el método de Fast Marching Square (FM2), para sistemas multi-UAV en entornos tridimensionales. El algoritmo 4D-FM2 incorpora una función de velocidad dependiente del tiempo dentro del marco del algoritmo Fast Marching (FM). El algoritmo propuesto fue probado en un escenario urbano simulado y los resultados muestran que este planifica de manera efectiva soluciones seguras y eficientes al problema de planificación de trayectorias para UAVs, evitando obstáculos y otros drones, incluso en situaciones complejas.

Palabras clave: Planificación de trayectorias, UAVs, Robótica inteligente

4D Trajectory Planning for UAVs using Fast Marching Square

Abstract

Swarms of unmanned aerial vehicles (UAVs) are becoming essential tools in various applications, such as aerial surveillance, urban search and rescue, and goods delivery. In this context, trajectory planning remains a significant challenge in complex robotic applications. This paper presents a 4D route planning algorithm based on the Fast Marching Square (FM2) method, designed for multi-UAV systems operating in three-dimensional environments. The 4D-FM2 algorithm integrates a time-dependent speed function into the Fast Marching (FM) framework. The proposed algorithm was tested in a simulated urban environment, and the results demonstrate its effectiveness in planning safe and efficient routes for UAVs, successfully avoiding obstacles and other drones even in complex scenarios.

Keywords: Trajectory and Path Planning, UAVs, Intelligent robotics

1. Introducción

Los vehículos aéreos no tripulados (UAV), o drones, han ganado popularidad recientemente debido a su capacidad para abarcar y agilizar una amplia variedad de aplicaciones civiles y comerciales, ya que pueden realizar tareas que van más allá de la capacidad de acción humana. En este contexto, los UAVs son capaces de realizar actividades como búsqueda y rescate, inspecciones de infraestructuras, distribución de pro-

ductos y vigilancia, entre otras funciones. No obstante, con la presencia simultánea de múltiples vehículos en un mismo espacio aéreo, se hace indispensable desarrollar estrategias para la gestión del tráfico aéreo y asegurar la seguridad de los vuelos durante las operaciones de los drones.

Una de las principales preocupaciones al desarrollar un sistema multiagente es su navegación, la cual se define como el desafío de trasladar a un agente desde un punto inicial hasta un punto final de manera segura, evitando colisiones con

*Autor para correspondencia: bllopezp@ing.uc3m.es
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

obstáculos en el entorno y con otros agentes, todo de forma autónoma. Por este motivo, se requiere cooperación espacio-temporal entre los drones que componen el enjambre. En general, este problema es dividido en dos fases (Huang et al., 2019): la generación de trayectorias desde el punto de partida hasta el objetivo para cada vehículo (es decir, la planificación global de rutas) y una segunda etapa de evasión local de colisiones (es decir, la etapa de gestión de conflictos).

En la literatura se pueden encontrar varias estrategias enfocadas a esa primera fase de planificación, diseñadas específicamente para sistemas multi-UAV. En Hu et al. (2020) se presenta un algoritmo A* dependiente de la velocidad que genera trayectorias con vectores de aceleración que convergen hacia los destinos predefinidos; mientras modelos de control predictivo se utilizan como marco para *solvers* de optimización evolutivos en Yu et al. (2019). En Bahabry et al. (2019), se desarrolla un programa MILP para modelar el problema y lograr una navegación óptima del enjambre, junto con dos algoritmos heurísticos basados en Dijkstra para resolver el problema de planificación de trayectorias. En cuanto a la segunda fase de maniobras de evasión, los algoritmos geométricos, como el enfoque del obstáculo de velocidad (Lombard et al., 2020), están emergiendo como la principal alternativa para este tipo de planificación, que depende tanto de tecnologías ADS-B como de la detección basada en métodos de visión.

Sin embargo, decidir y realizar estas maniobras de evasión de colisiones en tiempo real durante el vuelo puede resultar en conflictos irresolubles. La escalabilidad del enjambre de UAVs y la capacidad de operar en escenarios complejos o congestionados se ven particularmente afectadas por este problema (Du et al., 2019). Dado que las situaciones de conflicto entre vehículos no pueden anticiparse durante la fase inicial de planificación global de rutas en 3D debido a su dependencia de la ejecución dinámica de las misiones, en la literatura se han presentado diferentes opciones para incorporar implícitamente la dimensión temporal en el problema de planificación de rutas. Estas son las conocidas como estrategias de planificación de trayectorias 4D, y han sido estudiadas tanto para la navegación de drones como para la aviación civil. Estos enfoques implican principalmente el uso de métodos de optimización, como algoritmos evolutivos o de enjambre (Guo et al., 2023), esquemas de optimización de programación de tareas o despegue (Chen et al., 2021), y métodos de control de horizonte retrocedente (RHC) (Seenivasan et al., 2020).

Este trabajo presenta una estrategia de planificación de trayectorias 4D para la navegación de drones basada en el método de planificación Fast Marching Square (FM2). FM2 se basa en el fenómeno de propagación de la onda para calcular los caminos más cortos o mapas de distancia dentro de un entorno representado mediante una cuadrícula de ocupación. La principal ventaja de utilizar FM2 como un algoritmo de planificación de trayectorias 4D es su capacidad para considerar implícitamente el movimiento y el perfil de velocidad del UAV a través del comportamiento de propagación de la luz. Esta característica única permite que el algoritmo considere la presencia de otros vehículos durante el proceso de planificación única y exclusivamente cuando se espera que el vehículo a planificar llegue al mismo lugar y en el mismo instante temporal que otro dron. Este enfoque minimiza los bloqueos espaciales innecesarios entre trayectorias y previene desvíos indeseados

del dron, generando trayectorias suaves y libres de conflictos desde el proceso de planificación.

El contenido del trabajo se estructura de la siguiente manera. La Sección 2 describe el algoritmo de planificación de trayectorias en 4D presentado en este trabajo. La Sección 3 muestra los resultados obtenidos al aplicar nuestro método a la planificación de trayectorias de UAVs en un escenario urbano simulado. Finalmente, la Sección 4 presenta las conclusiones principales. Cabe destacar que los términos dron y UAV se utilizan indistintamente en este documento.

2. Metodología

En esta sección, se presenta el algoritmo Fast Marching como método de planificación de trayectorias, así como los métodos utilizados para desarrollar el algoritmo de planificación de trayectorias 4D basado en Fast Marching Square (4D-FM2) y su aplicación para sistemas multi-UAV.

2.1. Fast Marching Square como algoritmo de planificación

Una característica deseable al generar trayectorias realizables para un vehículo es obtener caminos suaves, cortos y libres de obstáculos. En la naturaleza, existen fenómenos que se comportan de manera similar, como la propagación de la luz. Según el principio de Fermat, una onda de luz que viaja desde una fuente puntual a través de un medio material llegará a cualquier punto en el espacio de la manera más rápida posible. Este comportamiento se vuelve especialmente interesante para la planificación de trayectorias, dado que el potencial artificial creado por la propagación de la onda y el gradiente del campo vectorial asociado podría utilizarse para calcular una ruta óptima desde un punto inicial hasta un punto final.

El algoritmo Fast Marching (FM) fue introducido por Sethian (Sethian, 1996). Es un método utilizado para resolver numéricamente la ecuación de Eikonal originalmente en una malla ortogonal rectangular, que modela la propagación de una onda en un medio no homogéneo mediante el tiempo de llegada de esa onda a cualquier punto en el espacio. Esto se expresa mediante la Ecuación 1,

$$1 = F(\rho)|\nabla T(\rho)|, \rho \in \mathbb{R}^N, \quad (1)$$

donde ρ representa cualquier punto en el entorno, $T(\rho)$ es el tiempo que tardaría la onda en llegar del punto inicial al punto ρ , $F(\rho)$ es la velocidad de propagación de la onda en ρ y N constituye la dimensión del problema. Por lo tanto, la magnitud del gradiente de la función de llegada $T(\rho)$ es inversamente proporcional a la velocidad de propagación.

La descripción completa de este algoritmo, junto con detalles sobre su implementación, está bien explicada en la literatura (González, 2015). De manera resumida, para generar una trayectoria desde un punto inicial hasta una ubicación objetivo en un entorno bidimensional o tridimensional, el método FM comienza tomando como entrada la matriz de ocupación binaria del escenario. Luego, calcula los tiempos de llegada desde el punto de partida a todos los puntos del espacio alcanzable. Finalmente, desde el punto de meta especificado y mediante técnicas de descenso de gradiente, calcula el camino que conduce al punto de origen por la ruta más corta.

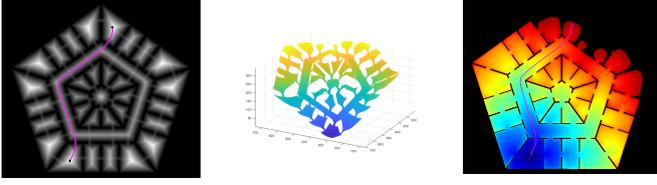


Figura 1: Ejemplo de uso de FM2. De izda. a dcha.: Mapa de velocidades obtenido en la primera etapa de FM2; Superficie de Lyapunov de los tiempos de llegada $T(\rho)$ en la segunda etapa de FM2; Tiempos de llegada representados en el mapa de ocupación. La trayectoria final se representa en magenta.

La variante FM2 consiste en aplicar el método FM dos veces. De esta manera, FM2 pasa a trabajar no con un mapa de ocupación binario si no con un mapa de velocidad o viscosidad W del entorno. Los valores de este mapa dependen, para cada punto, de su distancia al obstáculo más cercano: aquellos más alejados de los obstáculos permitirán que la onda de propagación tenga velocidades más altas que los puntos más cercanos a los obstáculos. Por lo tanto, las trayectorias generadas tenderán a estar alrededor de áreas con valores de velocidad más altos (es decir, más alejados de los obstáculos) y, por ende, serán más seguras y suaves. Se pueden aplicar modificaciones adicionales a este mapa de velocidades para lograr comportamientos deseados. En el caso de la planificación de trayectorias para UAVs, se establecieron niveles de vuelo, zonas de no vuelo y corredores de vuelo libre en algunos de nuestros trabajos anteriores (López et al., 2021). Un ejemplo del uso de FM2 en planificación se muestra en la Figura 1.

2.2. Planificación de trayectorias 4D basada en Fast Marching Square

El problema de planificación de trayectorias en 4D puede abordarse mediante el método FM considerando una función de velocidad dependiente del tiempo $F(\rho)$. De esta manera, la ecuación de la Eikonal se expresaría como en la Ecuación 2,

$$1 = F(\rho, T(\rho))|\nabla T(\rho)|, \rho \in \mathbb{R}^N. \quad (2)$$

Considerando un conjunto creciente de K instantes de tiempo $t_0 < t_1 < \dots < t_k < \dots < t_K$ y un conjunto de K mapas de velocidad $F_0(\rho), F_1(\rho), \dots, F_k(\rho), \dots, F_K(\rho)$ asociados a estos instantes temporales, el algoritmo FM2 puede configurarse para realizar interpolación lineal en cada intervalo de tiempo, de manera que el mapa de viscosidad resultante utilizado para la planificación de trayectorias se obtiene iterativamente y resulta en una combinación de todos los mapas de velocidad considerados. Esto se expresa en la Ecuación 3,

$$F(\rho, t) = (1 - \alpha)F_k(\rho) + \alpha F_{k+1}(\rho), \quad (3)$$

para $t = (1 - \alpha)t_k + \alpha t_{k+1}$,

donde $\alpha \in [0, 1]$ y $0 \leq k < K$. Para intervalos de tiempo anteriores a t_0 y posteriores a t_K , $F(\rho, t) = F_0$ y $F(\rho, t) = F_K$, respectivamente. De acuerdo con estas afirmaciones, para poder implementar este enfoque de FM2 en trayectorias 4D (4D-FM2) en el problema de planificación de rutas para equipos

multi-UAV, el primer paso consiste en definir el conjunto de tiempos t_{set} y mapas de velocidad F_{set} necesarios.

En este caso de aplicación concreto, el objetivo es planificar una ruta libre de conflictos y colisiones con respecto a obstáculos del entorno y otros agentes del sistema. Por tanto, estos conjuntos de intervalos de tiempos y mapas de velocidad serán los encargados de registrar en qué posiciones están los otros agentes y en qué instantes de tiempo. De esta manera, el planificador considerará como obstáculo (i.e. velocidad de propagación igual a 0 en los mapas de velocidad F_k) estas zonas y en dichos tiempos (i.e. en los instantes t_k correspondientes), a la hora de generar una nueva trayectoria para un dron dado. Hay que considerar que las trayectorias de los UAV se planifican secuencialmente sin ningún criterio específico, más allá del orden de llegada de la solicitud para calcular una nueva trayectoria. Para obtener una solución robusta, se ha de establecer una escala de referencia temporal común para todas las trayectorias de la misión, de forma que se obtenga un conjunto coherente de tiempos relevantes t_{set} . Por lo tanto, este conjunto estará compuesto por valores de tiempo proporcionales a una escala temporal predefinida dada por t_s .

Es importante destacar que la presencia de otro vehículo en el momento t_k y en una determinada posición del entorno se tendrá en cuenta durante la planificación únicamente si se prevé que el vehículo a planificar llegue a esa ubicación al mismo tiempo. Es decir, se considerará si la propagación de onda del método FM2 llega a esa ubicación en dicho tiempo. Esta es una contribución significativa del uso de FM2 como algoritmo de planificación de trayectorias en 4D.

Con el fin de garantizar cierto grado de seguridad, se consideran ciertos márgenes de confianza a la hora de introducir los bloqueos dados por otros agentes en los conjuntos t_{set} y F_{set} , tanto en distancia como en tiempo. De esta manera se considerará la presencia de otro vehículo en el intervalo $t_k \pm t_f$, donde t_f es un valor de tiempo constante y positivo. Por otro lado, para garantizar una distancia de seguridad mínima d_{sec} entre los drones, se debe marcar convenientemente una zona de seguridad alrededor de cada waypoint registrado en F_{set} . Cada waypoint es un punto que conforma la trayectoria de un vehículo. Como se ha mencionado, aquellos waypoints correspondientes a las trayectorias de otros vehículos sólo se seleccionan y registran con precisión cada t_s pasos, por lo que el tamaño de esta zona de seguridad debe ser lo suficientemente amplio como para garantizar d_{sec} en cualquier momento. Este tamaño de separación de seguridad d_{plus} viene dado entonces por la Ecuación 4,

$$d_{plus} = d_{sec} + v_m \cdot t_s. \quad (4)$$

Con el fin de asegurar una interpolación suave entre las matrices de F_{set} , una buena práctica consiste en *oscurecer* estas matrices F_{set} , es decir, reducir su valor de viscosidad original en un factor $0 < \varepsilon < 1$, alrededor de las zonas potencialmente conflictivas ya marcadas con un valor de velocidad relativa cero. Esto debe hacerse tanto espacialmente (en celdas vecinas más cercanas que una distancia dada d_u) como temporalmente (para $t_k \pm t_u$, donde t_u es un valor temporal positivo constante superior a t_f). Un ejemplo visual en 2D de cómo considerar estas ocupaciones dadas por otros agentes en el entorno se muestra en la Figura 2. Otra práctica útil consiste en añadir matrices

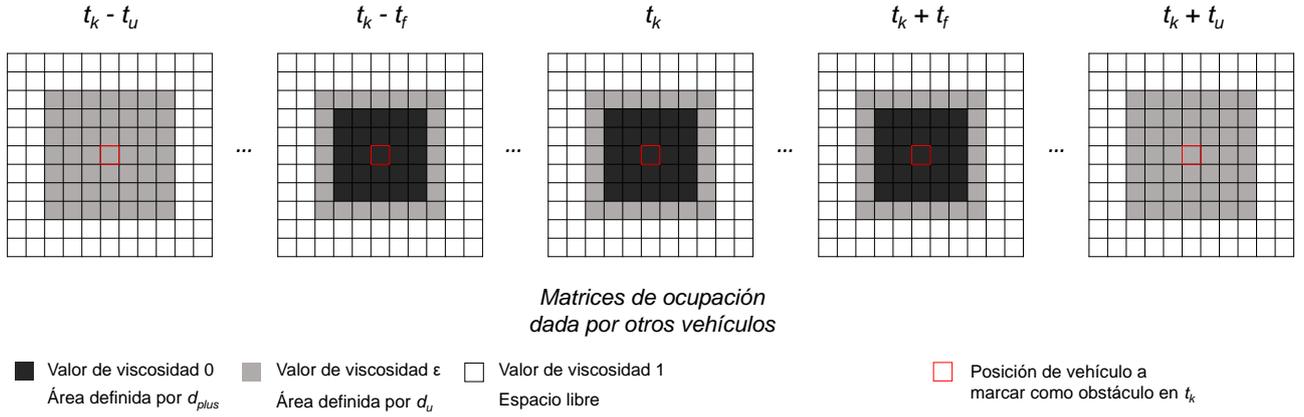


Figura 2: Representación 2D de cómo se añade la posición de otros vehículos a las matrices de viscosidad de planificación de trayectorias. En este ejemplo, sólo hay que considerar un dron en el instante de tiempo t_k .

F_k adicionales iguales al mapa de velocidad básico W (y su t_k asociado) a los conjuntos t_{set} y F_{set} entre cada dos marcas de tiempo consecutivas de los conjuntos originales, e.g. t_k y t_{k+1} , que estén separadas por un tiempo superior a un umbral dado t_l y también en la última marca de tiempo t_K . Recordando que W registra obstáculos del entorno únicamente, de este modo se evita el bloqueo innecesario proveniente de la presencia de otros drones en el espacio aéreo durante el proceso de interpolación entre matrices F_{set} . Dicha interpolación se ha añadido a nuestro algoritmo de planificación de trayectorias base FM2 gracias al solucionador Fast Marching proporcionado por Mirebeau and Portegies (2019).

Los drones considerados en este estudio son multirrotor, son tratados como un punto en el espacio con 3 GDL y tienden a volar a una velocidad constante de misión v_m . Basándonos en esta premisa, una de las ventajas del método FM2 es que no solo genera una trayectoria tridimensional suave, sino que también puede proporcionar un perfil de velocidad para el vehículo que seguirá dicha trayectoria. Estos valores de velocidad pueden extraerse al relacionarlos con la velocidad de propagación de la onda FM2, la cual puede ajustarse de acuerdo con el mapa de viscosidad $F(\rho, t)$ (escalado de cero a uno) y la velocidad de misión deseada v_m . Por lo tanto, el tiempo estimado de llegada (ETA) a cada waypoint (WP) de la trayectoria de una misión puede derivarse de la propagación de ondas dado por $T(\rho)$ en la segunda etapa FM2, denominada como la matriz D de tiempos de llegada.

Para calcular la trayectoria a partir de estos datos, hay que realizar el descenso de gradiente en D . Con el fin de manejar la discretización del procedimiento dada por las matrices de viscosidad discretas, el algoritmo Fast Marching utiliza un esquema numérico eficiente para discretizar la ecuación diferencial parcial que gobierna la propagación del frente de onda. A continuación, calcula a partir de D una curva de movimiento continua, que es una aproximación de la trayectoria geodésica desde el punto de origen hasta la meta. Para obtener trayectorias 4D, hay que extraer estos valores temporales de D accediendo a los valores de la matriz correspondientes a los índices dados por las coordenadas espaciales de la trayectoria. Por esta razón, mirar una matriz discreta a través de los índices de una trayectoria geodésica continua aproximada conduce a incoherencias de velocidad y tiempo.

Para resolver este problema, en este trabajo se ha llevado a cabo un método personalizado de búsqueda de trayectorias discretas a través de un algoritmo de descenso de gradiente restringido. Basándose en la matriz D y en el tiempo de llegada asignado al punto final, la trayectoria 4D resultante hacia la posición de origen se obtiene iterativamente seleccionando el siguiente waypoint de las celdas vecinas de la posición actual mediante reglas espacio-temporales predeterminadas. Estas reglas garantizan que la solución es consistente tanto en distancia entre WPs, tiempos de llegada y velocidad de misión v_m . Dado que D no presenta mínimos locales por definición, encontrar la solución está garantizado. Siguiendo este proceso, se genera una nueva trayectoria de cuatro dimensiones que contiene información sobre las ubicaciones espaciales y el ETA de cada waypoint. A partir de estos datos, se puede derivar una aproximación de la velocidad v_j que experimentará el vehículo en cada posición de la trayectoria para cumplir con el plan de misión mediante la Ecuación 5

$$v_j = \frac{d_{j+1,j}}{t_{j+1,j}} \quad \forall 1 \leq j \leq NW - 1, \quad (5)$$

donde NW es el número de waypoints, $d_{j+1,j}$ es la distancia euclídea entre dos waypoints consecutivos y $t_{j+1,j}$ es el tiempo esperado que tardará el vehículo en recorrer dicha distancia. Consideramos que $v_{NW} = v_{NW-1}$. Como la trayectoria se ha extraído discretamente de D , para garantizar un perfil de velocidad suave que evite desviaciones temporales indeseables, este perfil inicial puede suavizarse ligeramente y los tiempos de llegada resultantes recalcularse de la misma manera. Para ello se utiliza un suavizador spline discretizado rápido, no supervisado y robusto, proporcionado por Garcia (2010).

En este trabajo, al calcular una trayectoria para una nueva misión, se pretende que su plan temporal correspondiente comience lo antes posible con respecto a un determinado valor de tiempo de despegue. Puede haber casos en que esto no sea factible, e.g. cuando el área alrededor de los waypoints iniciales esté ocupada por otros vehículos en ese instante de tiempo. Además, otros bloqueos marcados en las matrices de viscosidad F_{set} pueden provocar que para ese instante de lanzamiento no sea posible encontrar una trayectoria correcta. Este problema se evita en gran medida mediante el mencionado proceso de *oscurecimiento* llevado a cabo por el factor ϵ , pero no com-

pletamente. En estas situaciones se aplica un retardo temporal de misión t_d . Finalmente, para garantizar que un nuevo plan de misión es válido y no interfiere con otras trayectorias calculadas previamente, se comprueba expresamente que ningún waypoint de la nueva trayectoria se encuentra a una distancia inferior a d_{sec} espacialmente y más cerca que t_f temporalmente. El marco general del algoritmo 4D Fast Marching Square presentado en este trabajo se muestra en la Figura 3.

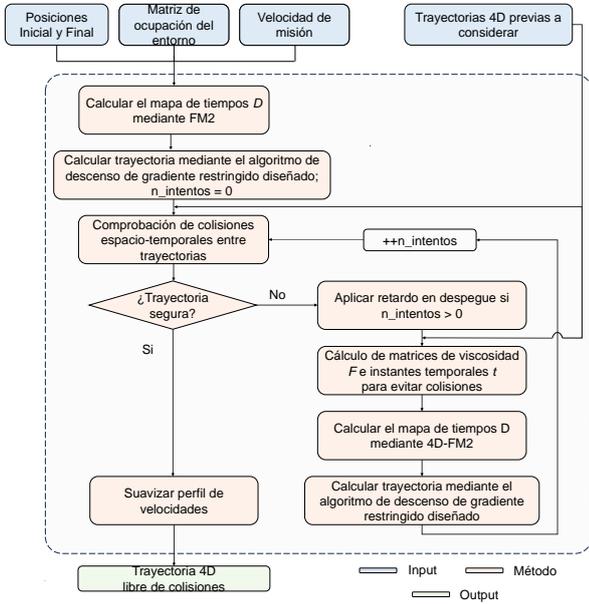


Figura 3: Framework del algoritmo 4D-FM2 propuesto.

3. Resultados

Con el fin de probar el planificador 4D Fast Marching Square presentado en este trabajo, se llevan a cabo una serie de simulaciones para analizar su rendimiento. El algoritmo se implementa en MATLAB y las pruebas se realizan en un ordenador con 16 GB de RAM y una CPU AMD Ryzen 5 3600 con Windows 10. Los resultados de estas simulaciones se presentan en esta sección. Los experimentos se realizan para distintos números de misiones simultáneas: 5, 10, 15 y 20. Cada una de estas corresponde a un vehículo UAV que vuela desde una posición inicial hasta una ubicación objetivo.

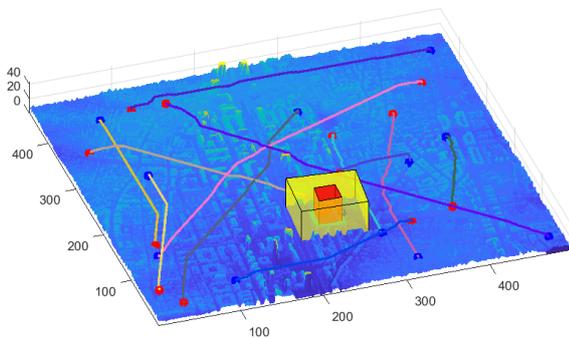


Figura 4: Entorno de simulación. Alrededores del Estadio Santiago Bernabéu en Madrid (España), junto a 12 trayectorias planificadas como ejemplo y dos zonas de restricción.

Para cada simulación, se realizan 7200 iteraciones, y cada una equivale a 1 segundo de tiempo de vuelo real. Cuando finaliza una misión, se planifica una nueva para dicho vehículo.

Los puntos inicial y final de cada misión se generan de manera aleatoria. Se estudia el rendimiento de cada conjunto de misiones simultáneas a diferentes velocidades de misión v_m , incluidas 2, 4 y 6 celdas/s. Para el movimiento simulado de los drones, se ha supuesto que vuelan siguiendo el perfil de velocidad previamente calculado.

El entorno simulado para las pruebas corresponde a los alrededores de la zona del Estadio Santiago Bernabéu de Madrid (España). Presenta una resolución de 5 metros por celda sobre una matriz de $467 \times 497 \times 44$. Un conjunto de 12 trayectorias, junto al entorno es representado en la Figura 4. Los parámetros analizados en las secciones anteriores se han elegido basándose en la experimentación. La distancia de seguridad d_{sec} se fija en 7 celdas; t_s se define en 2 segundos para experimentos con v_m igual a 2 celdas/s y 1.25 segundos en caso contrario; t_f se fija en 5 segundos, mientras que t_u es igual a 8 segundos. El factor *darkening* ϵ se define en 0.65, mientras que la distancia de seguridad d_u se establece en $1,5 \cdot d_{sec}$. La constante de tiempo t_l es de $5 \cdot t_s$. El tiempo de retraso t_d es 20 segundos, estableciéndose un máximo de 2 minutos de retraso para una misión. Si no se puede calcular una trayectoria para una misión determinada en función de estos parámetros, la misión se considerará fallida y se asignará una nueva.

Se recogieron una serie de medidas estadísticas durante las simulaciones, siendo estas: Número de misiones completadas por cada UAV, tiempo de retraso requerido por misión, número de UAVs volando a la vez por iteración, número total de misiones completadas por el equipo, tiempo computacional requerido por misión y velocidad media de vuelo durante cada misión. Para cada velocidad de misión v_m y para cada conjunto de misiones simultáneas, algunas de estas medidas mencionadas se presentan directamente y otras en forma de valor medio. Estos resultados se presentan a continuación.

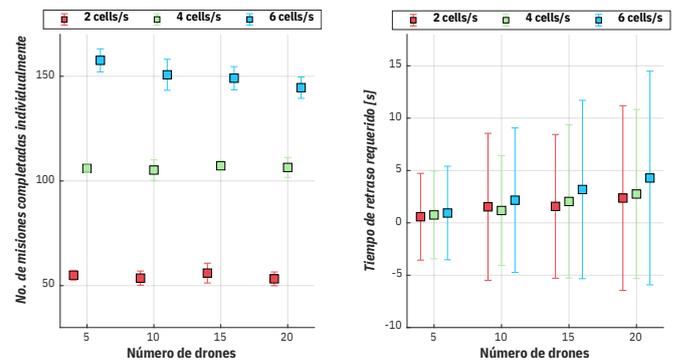


Figura 5: Medidas estadísticas recogidas en simulaciones. De izda. a dcha.: N° de misiones realizadas por cada UAV; Tiempo de retraso necesario para una misión.

En la Figura 5 se observa que a valores de v_m mayores el número de misiones completadas por cada UAV también aumenta. Esto es porque los vehículos pueden completar misiones en menor tiempo. También se extrae de esta Figura que a medida que aumenta el número de drones y la velocidad, el tiempo de espera para iniciar una misión aumenta gradualmente, aunque siempre se mantiene en valores razonables. Además, viendo la Figura 6a, aunque el número de vehículos que pueden volar al mismo tiempo está siempre cerca del valor máximo, disminuye de forma inversa a v_m . Por tanto, cuanto

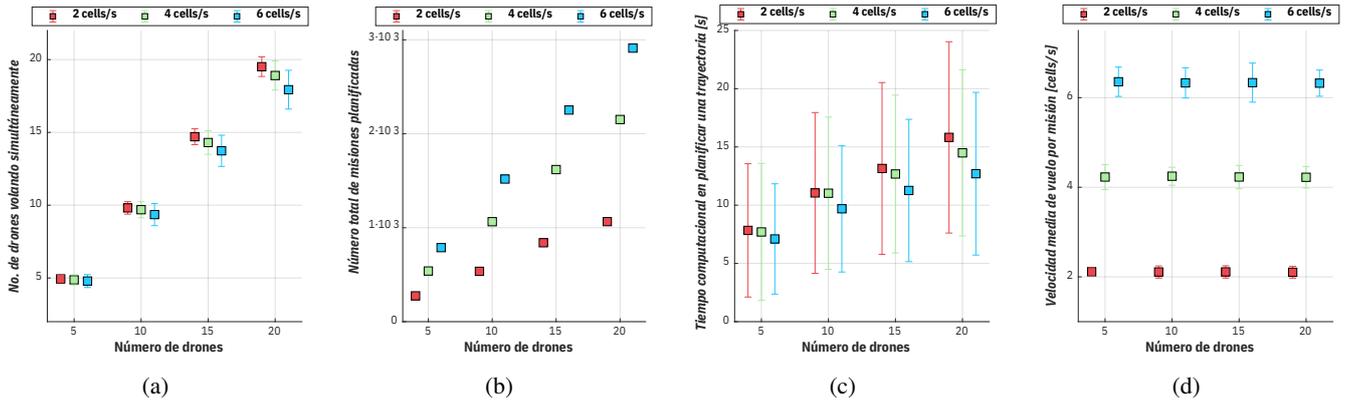


Figura 6: Medidas estadísticas recogidas en simulación. De izda. a dcha.: N° de UAVs volando simultáneamente; N° de misiones completadas por el equipo; Tiempo computacional para obtener una trayectoria; Velocidad media de vuelo por misión.

mayor es v_m y el número de misiones simultáneas, más congestionado está el espacio aéreo. El número total de misiones completadas por el grupo se ve incrementado tanto respecto a v_m como al número de agentes (ver Figura 6b).

En cuanto al tiempo necesario para calcular una nueva trayectoria, la Figura 6c indica que esta medida aumenta proporcionalmente al número de drones, ya que hay que considerar más trayectorias durante la etapa de planificación. Sin embargo, este valor de tiempo es menor para velocidades más altas. Esto se debe a que la propagación de ondas llevada a cabo en el método 4D-FM2 puede realizarse a mayor velocidad, acelerando el proceso computacional. Por lo tanto, se ha comprobado que, en general, el algoritmo 4D-FM2 puede planificar una nueva misión en menos de 25 segundos, incluso en el caso menos favorable, siendo el tiempo medio inferior a este valor. Por último, la Figura 6d muestra que la velocidad a la que los vehículos realizan sus misiones, obtenida a partir de los valores de ETA dados por la matriz D , es efectivamente muy próxima a la velocidad establecida v_m . Sin embargo, se observa que estos valores de velocidad son ligeramente superiores a los esperados. Esto puede deberse a imprecisiones inherentes al solucionador de Fast Marching utilizado.

Por último, cabe destacar que la distancia mínima de seguridad entre vehículos d_{sec} se preservó durante todo momento en las simulaciones. También, respecto al número de misiones que podían resultar fallidas, cuando v_m se fijó en 2 celdas/s, todas las misiones se calcularon correctamente sin ningún problema. A velocidades más altas, solo se produjeron hasta 2 misiones fallidas por cada conjunto de misiones simultáneas, lo que puede considerarse insignificante si se tiene en cuenta el número total de misiones completadas.

4. Conclusiones

Este trabajo ha presentado un algoritmo de planificación de trayectorias 4D para sistemas multi-UAV basado en el método Fast Marching Square. Esto se ha conseguido al trabajar con una función de velocidad dependiente del tiempo, lo que permite considerar de manera eficiente la presencia de otros vehículos en determinados instantes temporales, permitiendo establecer las maniobras de evitación directamente en la etapa de planificación. El método 4D Fast Marching Square (4D-FM2) se evaluó en un escenario urbano simulado en 3D para varios números de misiones concurrentes y velocidades de UAV. Los resultados demostraron que la distancia mínima

de seguridad se mantuvo sistemáticamente a lo largo de las simulaciones y el método demostró un buen rendimiento general en términos de manejo de tráfico aéreo.

Agradecimientos

This work was supported by the European Commission's Innovation and Networks Executive Agency through the European H2020 LABYRINTH project. Additionally, it was supported by the Acción Estratégica en Exoesqueletos Robóticos y Algoritmos Avanzados de Planificación de Movimientos.

Referencias

- Bahabry, A., Wan, X., Ghazzai, H., Menouar, H., Vesonder, G., Massoud, Y., 2019. Low-altitude navigation for multi-rotor drones in urban areas. *IEEE Access* 7, 87716–87731.
- Chen, Y., Xu, Y., Hu, M., Huang, F., Nie, Q., 2021. A 4d-trajectory planning method based on hybrid optimization strategy for demand and capacity balancing. In: 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC). IEEE, pp. 1–9.
- Du, Y., Zhang, X., Nie, Z., 2019. A real-time collision avoidance strategy in dynamic airspace based on dynamic artificial potential field algorithm. *IEEE Access* 7, 169469–169479.
- García, D., 2010. Robust smoothing of gridded data in one and higher dimensions with missing values. *Comput. Stat. & data analysis* 54, 1167–1178.
- González, J. G., 2015. Fast marching methods in path and motion planning: improvements and high-level applications. Ph.D. thesis, Univ. Carlos III de Madrid.
- Guo, T., Mei, Y., Tang, K., Du, W., 2023. A knee-guided evolutionary algorithm for multi-objective air traffic flow management. *IEEE Transactions on Evolutionary Computation*.
- Hu, Y., Yao, Y., Ren, Q., Zhou, X., 2020. 3d multi-uav cooperative velocity-aware motion planning. *Future Generation Computer Syst.* 102, 762–774.
- Huang, S., Teo, R. S. H., Tan, K. K., 2019. Collision avoidance of multi-uavs: A review. *Annual Reviews in Control* 48, 147–164.
- Lombard, A., Durand, L., Galland, S., 2020. Velocity obstacle based strategy for multi-agent collision avoidance of unmanned aerial vehicles. In: 2020 IEEE International Conference on Sensing, Communication and Networking (SECON Workshops). IEEE, pp. 1–6.
- López, B., Muñoz, J., Quevedo, F., Monje, C. A., Garrido, S., Moreno, L. E., 2021. Path planning and collision risk management strategy for multi-uav systems in 3d environments. *Sensors* 21 (13), 4414.
- Mirebeau, J.-M., Portegies, J., 2019. Hamiltonian fast marching: a numerical solver for anisotropic and non-holonomic eikonal pdes. *Image Processing On Line* 9, 47–93.
- Seenivasan, D. B., Olivares, A., Staffetti, E., 2020. Multi-aircraft optimal 4d online trajectory planning in the presence of a multi-cell storm in development. *Transportation Research Part C: Emerging Tech.* 110, 123–142.
- Sethian, J. A., 1996. A fast marching level set method for monotonically advancing fronts. *Proc. of the National Academy of Sci.* 93 (4), 1591–1595.
- Yu, W., Yanyang, W., Xiangju, Q., Ligu, S., 2019. Exploring mission planning method for a team of carrier aircraft launching. *Chinese Journal of Aeronautics* 32 (5), 1256–1267.