

Estrategias para la exploración coordinada multi-VANT

Luis Alberto Ballado Aradias

Asesor: Dr. José Gabriel Ramírez-Torres

Co-Asesor: Dr. Eduardo Arturo Rodríguez-Tello

11 de diciembre de 2023

Resumen

La exploración multi-robot ha surgido como un enfoque prometedor para la generación eficiente del mapa de un medio ambiente desconocido. Un enfoque colaborativo ofrece mejores resultados de exploración con una rápida obtención de información, logrando sus objetivos con un alto grado de consistencia y resiliencia a fallos en comparación con implementaciones donde se emplea un único robot. Sin embargo, la exploración multi-robot plantea diversos desafíos que deben abordarse para su correcta implementación, como la comunicación, la colaboración y la fusión de datos.

En la última década se ha tenido un aumento en la investigación y el desarrollo en el campo de los vehículos aéreos no tripulados (VANTS), lo que ha dado lugar a importantes avances e innovaciones en esta área. Los sistemas con múltiples vehículos aéreos no tripulados (multi-VANT) permiten la adquisición simultánea de datos desde múltiples puntos de vista privilegiados, en comparación con robots terrestres, lo que permite mejorar la generación de mapas de entornos desconocidos. El uso de algoritmos de coordinación inteligente y la toma descentralizada de decisiones proporciona una mayor confiabilidad a estos sistemas, ya que cualquier falla o mal funcionamiento de un VANT puede ser compensado por los VANTS restantes. Además, un buen protocolo de comunicación permite una colaboración fluida entre los robots, mejorando su capacidad para moverse de un lugar a otro evitando obstáculos a su paso (navegar), descubrir un entorno desconocido adquiriendo información a su paso (explorar). Asimismo, la integración de sensores de última generación los convierte en herramientas útiles en diversos dominios [1, 2, 3]. La mayor parte de las aplicaciones que hacen uso de VANTS suelen carecer de autonomía, requiriendo la asistencia y vigilancia constantes de un usuario humano. Para que un robot se considere autónomo deberá tomar decisiones y realizar tareas sin necesidad de que alguien le diga qué hacer o guiarlo paso a paso. Tener la capacidad de percibir su entorno y usar la información para decidir cómo moverse son considerados altos niveles de autonomía. Para llegar a ello, el robot debe resolver primero problemas como su localización, construir el mapa de su entorno y posteriormente usarlo para navegar dentro de él.

El enfoque de este trabajo es la propuesta de una arquitectura descentralizada de software capaz de coordinar múltiples vehículos aéreos no tripulados con habilidades para la exploración, generación de mapas y planificación de rutas para explorar eficientemente un área de interés. Este problema implica la toma de decisiones complejas, como asignar tareas de exploración a los robots, evitar colisiones y planificar rutas óptimas. Factores como la comunicación entre robots, la incertidumbre del entorno y las limitaciones de recursos de energía forman parte de los criterios considerados en este trabajo.

Palabras claves: estrategias multi-VANT, exploración multi-VANT, planificación de rutas multi-VANT, arquitectura de software multi-VANT.

Antecedentes y motivación para el proyecto

Los robots de servicio, son máquinas autónomas diseñadas con el objetivo de prestar servicio a los humanos, convirtiéndose poco a poco en una parte esencial en nuestras vidas. Los podemos encontrar en diversos ámbitos, como en el entretenimiento, limpieza, logística, entre otras soluciones innovadoras [4].

Los VANTS, que se han vuelto cada vez más frecuentes, son considerados robots de servicio debido a su capacidad para realizar tareas que gracias a la movilidad que sus seis grados de libertad (DoF) les permite desplazarse. En fotografía y video, los VANTS pueden obtener sorprendentes tomas aéreas para fines de filmación, bienes raíces, turismo y entretenimiento. En la agricultura, los VANTS se utilizan para el control de cultivos, mejorando la gestión de recursos. En el mantenimiento de infraestructuras, los VANTS juegan un papel importante ayudando en la inspección de puentes, edificios, líneas eléctricas y tuberías, reduciendo los riesgos y costos asociados con las inspecciones manuales. En misiones de búsqueda y rescate, donde ayudan en evaluaciones posteriores a un desastre, los VANTS han demostrado ser muy útiles debido a las ventajas en términos de movilidad.

La mayoría de las aplicaciones que desempeñan los VANTS son sencillas, estáticas, en espacios controlados y con rutas predeterminadas, o bien, en contextos de operación por control remoto por un usuario (piloto). Para aplicaciones más complejas, donde el robot debe responder de manera autónoma con la mínima intervención humana, es decir, que solo se les ordene las tareas que deban realizar y no guiarlos en cada uno de sus movimientos. Para lograr esto, se requiere que el robot cuente con habilidades de identificación de contextos, planificación de tareas y manejo de mapas.

La importancia de la exploración con robots radica en su capacidad para superar los riesgos que enfrentan los humanos al exponerse a entornos desconocidos y peligrosos. Los robots se pueden diseñar para resistir a condiciones extremas, como las misiones espaciales [5], la exploración en aguas profundas [6] o áreas afectadas por desastres [7], donde la presencia humana puede no ser segura, permitiéndoles acceder a lugares de difícil acceso [8]. La vasta mayoría de los desarrollos en esta área de investigación se ha centrado en sistemas que emplean un único robot.

No se puede subestimar la importancia de utilizar sistemas con múltiples robots en las actividades de exploración, ya que su escalabilidad y adaptabilidad los hace adecuados en varios escenarios y entornos, que van desde misiones de pequeña escala a misiones de gran escala o demasiado complejas para un único robot. Los sistemas multi-robot ofrecen beneficios que mejoran la efectividad y resiliencia en este tipo de tareas. Al emplear múltiples robots permite la cobertura simultánea de una mayor área, lo que resulta en una exploración rápida del entorno [9], lo que es extremadamente importante en aplicaciones donde el tiempo es crítico, como las misiones de búsqueda y rescate.

Un sistema multi-VANT, puede colaborar, intercambiar información y minimizar rutas para agilizar el proceso de exploración. Además, el uso de múltiples VANTS aseguran que la misión no se vea comprometida, agregando tolerancia en caso de que un VANT encuentre dificultades, otros VANTS pueden continuar la exploración, asegurando la continuidad de la misión, reduciendo el riesgo de fracaso. Además, los sistemas multi-VANT permiten la especialización de tareas, donde diferentes VANTS pueden equiparse con sensores o instrumentos especializados para recopilar datos específicos.

El uso de sistemas multi-robot trae consigo retos inherentes que deben abordarse (Figura 1 ilustra los retos multi-VANT). La coordinación y colaboración entre múltiples robots presenta desafíos en términos de comunicación, asignación de tareas y sincronización. Establecer canales de comunicación efectivos entre los robots es crucial para compartir información, coordinar acciones y evitar colisiones. Se requieren algoritmos de asignación de tareas para distribuir diferentes tareas

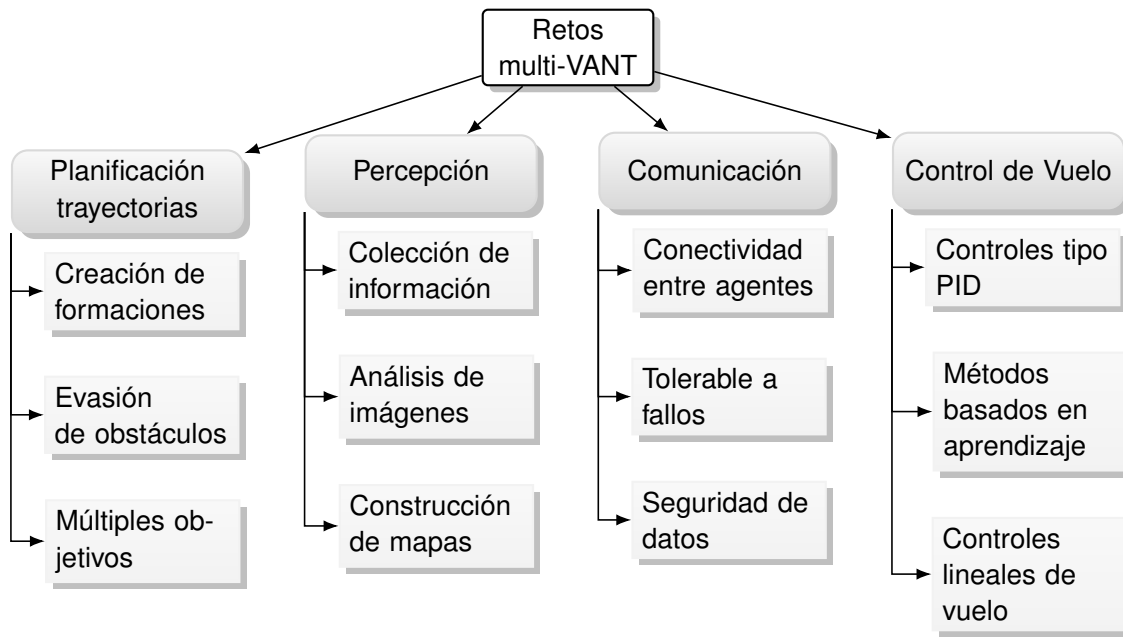


Figura 1: Retos multi-VANT

de exploración entre los robots, teniendo en cuenta factores como la ubicación, las capacidades y los niveles de energía para optimizar la división del trabajo. Además, es fundamental garantizar la sincronización y evitar colisiones entre los robots en entornos dinámicos. Por otra parte, la integración y fusión de datos de múltiples sensores plantea desafíos en términos de sincronización, confiabilidad y consistencia de datos, para combinar de manera efectiva los datos recopilados por los robots individuales en una representación coherente del entorno.

Durante la planificación de rutas seguras, se deben tomar en cuenta las restricciones de movimiento propias del robot, para que este pueda ejecutar la trayectoria en el mundo real. Los problemas que emergen de la planificación de trayectorias es la escalabilidad y eficiencia computacional. Considerando mover un VANT en 3D que puede trasladarse y rotar, el problema consiste en optimizar trayectorias en 6 grados de libertad (DoF) empleando algoritmos que deben ejecutarse en tiempo real dentro de dispositivos computacionalmente limitados.

La dificultad en la planificación de trayectorias para múltiples VANTS es inherente a la complejidad espacial del entorno y las maniobras que pueda realizar el VANT Xu et al.[10]. La minimización de la longitud de las rutas y la seguridad del trayecto para todos los robots durante el vuelo son partes clave cuando se crea un planificador de trayectorias multi-VANT.

En la última década se han propuesto diversas soluciones globales de planificación basadas en distintas técnicas de programación (Mixed Integer Linear Programming (MILP) [11], Nonlinear programming (NP) [12] y Dynamic Programming (DP) [13]), pero su complejidad computacional crece exponencialmente conforme aumenta el espacio de búsqueda.

Otros métodos que han sido ampliamente trabajados son los Campos de Potencial Artificial por sus ventajas de cómputo en tiempo real. Desafortunadamente, este método cae en mínimos locales de la función potencial, llegando a fallar para encontrar una solución que lo guíe al punto destino.

Estado del arte

Las aplicaciones de la robótica industrial se han centrado en realizar tareas simples y repetitivas. La necesidad de robots con capacidad de identificar cambios en su entorno y reaccionar sin la intervención humana, da origen a los robots inteligentes. Aunado a ello, si deseamos que el robot se mueva libremente, los cambios en su entorno pueden aumentar rápidamente y complicar el problema de desarrollar un robot que muestre un comportamiento inteligente.

El despliegue rápido de robots en situaciones de riesgo, búsqueda y rescate ha sido un área ampliamente estudiada en la robótica móvil. Recientemente, concursos como el DARPA Subterranean (SubT) Challenge, buscan acelerar la investigación y desarrollo de tecnología en escenarios complejos subterráneos, donde factores como exploración, planeación y coordinación son clave para lograr los objetivos del Reto [14], la aplicación de una estrategia de exploración es pieza clave para el cumplimiento del reto.

La exploración de un ambiente desconocido empleando multi-VANT es un área relativamente nueva y con mucho crecimiento en los últimos años. Se han abordado una variedad de temas para lograr la exploración autónoma, desde la planificación de rutas para múltiples robots terrestres en tareas de exploración [9], estrategias para la coordinación y protocolos de comunicación [15]. Diversos estudios de aplicaciones multi-VANT se han realizado para resolver tareas como el monitoreo ambiental [16], la agricultura de precisión [17] y operaciones de búsqueda y rescate [1], por mencionar algunas.

La dirección en que apunta el estado del arte actualmente se puede atribuir a los avances en tecnología en la última década que nos han permitido dotar de autonomía a VANTS de menor tamaño. Investigadores de diversas áreas, que incluyen las ciencias computacionales y la ingeniería, han contribuido al progreso de este campo.

Sin embargo, la necesidad de una arquitectura de software descentralizada que englobe todos estos componentes y que sea capaz de coordinar múltiples VANTS, surge como una área de investigación prometedora que garantice el funcionamiento correcto los componentes necesarios para realizar tareas de exploración coordinada multi-VANT.

En esta revisión del estado del arte se citan trabajos relacionados a la coordinación multi-VANT para tareas de exploración, que sirven de guía para la realización de este trabajo.

Planificación de trayectorias

Uno de los desafíos clave en la colaboración de múltiples VANTS es la planificación de rutas. Se han desarrollado diversos algoritmos para optimizar la planificación de rutas dentro de la robótica móvil (tabla 1), minimizando los riesgos de colisión y mejorando la eficiencia en sus misiones. Estos algoritmos tienen en cuenta varios factores como las restricciones del robot y la ubicación del objetivo, para generar trayectorias seguras.

El objetivo principal de los algoritmos de navegación es el de guiar al robot desde el punto de inicio al punto destino. Los trabajos por Lumelsky and Stepanov[18], dieron respuesta a problemáticas de navegación eficiente, que no requieren de una representación del medio ambiente y emplean, por lo tanto, pocos recursos computacionales y de memoria (algoritmos tipo bug).

Matemáticamente, el problema de planificación de rutas es resuelto a través del modelado del medio ambiente utilizando grafos, siendo un grafo una representación matemática de vértices y aristas. Hart et al.[19], al mejorar el algoritmo de Dijkstra para el robot Shakey, logró navegar en

una habitación que contenía obstáculos fijos. El objetivo principal del algoritmo A* es la eficiencia en la planificación de rutas. A su vez, el algoritmo D*, propuesto por Stentz[20], ha demostrado operar de manera eficiente ante obstáculos dinámicos; en comparación con el algoritmo A* que vuelve a ejecutarse al encontrarse con un obstáculo no previsto inicialmente, el algoritmo D* usa la información previa para buscar una nueva ruta hacia el objetivo.

Método	Completez	Óptimo	Escalable	Notas
Grafo de visibilidad [21]	✓	✓	✗	<ul style="list-style-type: none"> ■ Mucho espacio libre ■ Mala escalabilidad ■ El robot pasa cerca de obstáculos
Diagramas de Voronoi [22]	✓	✗	✗	<ul style="list-style-type: none"> ■ Espacio libre máximo ■ Rutas conservadoras ■ Mala escalabilidad
Campos de potencial artificial [23]	✓	✗	Depende del ambiente	<ul style="list-style-type: none"> ■ Fácil de implementar ■ Susceptible a mínimos locales
Dijkstra/A* [24]	✓	Grafo	✗	<ul style="list-style-type: none"> ■ Más rápido que la búsqueda desinformada ■ A* usa una función heurística para impulsar la búsqueda de manera eficiente ■ Mala escalabilidad
PRM [25]	✓	Grafo	✓	<ul style="list-style-type: none"> ■ Eficiente para problemas con consultas múltiples ■ Completez probabilística ■ Camino irregular
RRT [26]	✓	✗	✓	<ul style="list-style-type: none"> ■ Eficiente para problemas de consulta única ■ Completez probabilística ■ Camino irregular

Tabla 1: Métodos para planificación de trayectorias usados en robótica móvil

Por otra parte, el algoritmo RRT (Rapidly Exploring Random Trees), propuesto por LaValle[27], es ampliamente usado para la planificación de rutas en robots modernos. El algoritmo construye de forma incremental una estructura de árbol mediante un muestreo aleatorio en el espacio de configuraciones, uniéndolo aleatoriamente nuevas posiciones al árbol existente hasta alcanzar la posición final. Las modificaciones realizadas al algoritmo RRT por Karaman and Frazzoli[28], incorporando una heurística de costo por recorrer, permite encontrar rutas casi óptimas de manera eficiente. Este enfoque es ampliamente usado en problemas de navegación autónoma y planificación de movimiento.

En recientes trabajos de Yang et al.[29], muestran la capacidad de implementación de algoritmos clásicos de planificación de trayectorias, como los grafos de visibilidad, para tareas en entornos conocidos y no conocidos, utilizando una representación del ambiente a base de polígonos, logrando un rápido planificador que también resuelve obstáculos nuevos en el ambiente, logrando resultados comparables a las estrategias más recientes como D* e inclusive RRT*.

Trabajos como el de Tranzatto et al.[30] en el que han logrado optimizar problemas de alta dimensionalidad como el control de navegación para un robot con cuatro patas, haciendo uso de aprendizaje por refuerzo y con ayuda de simulaciones, logrando obtener un esquema de control que le permiten al robot resolver el problema de navegación. Sin embargo, al momento de probar el esquema en un robot real, el robot no pudo efectuar un paso correcto. Este problema se debe a la distancia que existe entre la simulación y la realidad, en particular al no considerar las incerti-

dumbres en las lecturas de los sensores.

Las simulaciones permiten demostrar el correcto funcionamiento de los esquemas de control. A través de simulaciones híbridas e introduciendo ruidos estocásticos en las simulaciones, es posible lograr resultados muy prometedores, como en el caso de éxito en el DARPA Subterranean Challenge [14], que utiliza una exploración basada en grafos y un mapa de ocupación (Octomap) para simular el entorno tridimensional.

Un enfoque muy recurrente para abordar problemas complejos, como la planificación de trayectorias, la fusión de datos y la distribución de tareas, es el uso de las metaheurísticas bio-inspiradas. Estas estrategias se inspiran en sistemas y procesos biológicos para resolver problemas complejos de optimización. Existen varios tipos de metaheurísticas bio-inspiradas:

1. **Algoritmos Genéticos (GA).** Propuestos por J. Holland, se basan en los principios de selección natural, usando operadores como la cruza, mutación y selección. Manteniendo una población de las posibles soluciones iterando para encontrar la solución cercana a la solución óptima.
2. **Particle Swarm Optimization (PSO).** Propuestos por Eberhart y Kennedy, inspirado en el comportamiento de parvadas de pájaros y cardumen de peces, el algoritmo involucra una población de partículas que se mueven en un espacio de búsqueda. Cada partícula ajusta su posición según su propia solución y la solución de toda la población.
3. **Ant Colony Optimization (ACO).** Propuesto por M. Dorigo, inspirado en el comportamiento de búsqueda de alimento de las hormigas, imita la comunicación y toma de decisiones colectiva de las hormigas, puede ser usado para encontrar caminos dentro de un grafo.
4. **Firefly Algorithm (FA).** Propuesto por X. Yang, sigue el modelo de los patrones intermitentes de las luciérnagas, el algoritmo emula el comportamiento de atracción y repulsión de las luciérnagas.

Las metaheurísticas han demostrado ser efectivas para resolver una amplia gama de problemas de optimización; sin embargo, su adopción en el campo de la robótica móvil se ve limitada por varias razones.

- **Complejidad y restricciones en tiempo real:** la robótica a menudo implica la toma de decisiones en tiempo real, donde los robots deben responder rápidamente a entornos cambiantes. Las metaheurísticas suelen requerir extensos recursos computacionales y temporales para converger en una solución óptima, lo que puede no ser factible en aplicaciones de robótica en tiempo real, particularmente en vehículos aéreos con limitado poder de cómputo. El control y la planificación en tiempo real en robótica a menudo requieren algoritmos de baja complejidad computacional, como la planificación clásica o los enfoques de control reactivo.
- **Soluciones rápidas:** en aplicaciones de robótica, especialmente las que involucran tareas críticas para la seguridad o control preciso, se prefieren las soluciones deterministas y predecibles a las soluciones estocásticas que ofrecen las metaheurísticas. Las metaheurísticas brindan soluciones aproximadas con diversos grados de optimización, que pueden no ser adecuadas para tareas que requieren un control preciso o garantías de seguridad.
- **Algoritmos de tareas específicas:** la robótica a menudo implica tareas y dominios específicos que se han estudiado ampliamente, lo que da como resultado algoritmos específicos de tareas adaptados a esos dominios. Estos enfoques personalizados a menudo son más eficientes y efectivos para resolver los problemas específicos abordados en robótica, lo que hace que las metaheurísticas de propósito general sean menos atractivas.

- **Limitaciones de hardware y energía:** los sistemas de robótica suelen tener recursos de hardware limitados y, a menudo, están limitados por el consumo de energía. Las metaheurísticas, que a menudo requieren una mayor cantidad de recursos o extensos tiempos de ejecución para alcanzar la convergencia, pueden no ser adecuadas para plataformas robóticas con recursos limitados.

Sin embargo, es importante tener en cuenta que ciertamente hay áreas dentro de la robótica donde las metaheurísticas se han aplicado con éxito, como la planificación de rutas de robots en entornos complejos, la robótica de enjambres o la asignación de tareas en sistemas de múltiples robots. Los enfoques híbridos que combinan metaheurísticas con optimización basada en modelos o algoritmos específicos de tareas pueden aprovechar las fortalezas de ambos y proporcionar soluciones efectivas para aplicaciones en la robótica.

Coordinación multi-robot

Además de la planificación de rutas, la coordinación de múltiples robots para la exploración requiere de una estrategia de comunicación efectiva, que garantice la coherencia del mapa que se va generando. Se han investigado diferentes protocolos de comunicación y estrategias de intercambio de información para permitir la colaboración. Algunos enfoques utilizan comunicación directa entre los robots, mientras que otros emplean una arquitectura de red donde los múltiples robots se comunican a través de una infraestructura descentralizada Chen and Hung[31], mostrando la tolerancia a fallas en equipos para tareas de búsqueda y rescate.

En el Centro de Investigación y Estudios Avanzados del Instituto Politécnico Nacional Unidad Tamaulipas se han realizado investigaciones en el área de exploración multi-robot y diseño de prototipos de VANTS, lo cual sirve como antecedente para este trabajo. Este relevante desarrollo, propuesto por Elizondo-Leal[32], tiene como objetivo principal el despliegue de una estrategia de coordinación para múltiples robots móviles basado en un enfoque de auto-ofertas.

En recientes trabajos Cieslewski [2021][33] ha demostrado descentralizar la tarea de SLAM para la creación de mapas en tareas de exploración multi-VANT eliminando el bloque de optimización, haciendo uso de técnicas de machine learning teach and repeat.

Arquitectura de software multi-robot

A lo largo del desarrollo de la robótica móvil se ha mostrado que estrategias de control basadas en comportamientos (behavior-based) pueden presentar un mejor desempeño que las arquitecturas secuenciales (Brooks[34]). Estas arquitecturas están organizadas en capas paralelas, cada una codificando un comportamiento distinto, en la que el robot sensa su entorno y reacciona a los estímulos. La respuesta final del robot es una combinación de las distintas respuestas activadas por los sensores. Este enfoque permite al robot resolver situaciones complejas a través de soluciones computacionales compactas y que requieren de pocos recursos, como la navegación y la evasión de obstáculos no previstos.

En lo que respecta a la organización de la arquitectura de control, Loquercio et al.[35] obtiene resultados muy interesantes empleando una arquitectura secuencial con capas de proyección, decisión y control posterior a un procesamiento de imagen, con uso de algoritmos para la estimación de un mapa, logrando navegar en entornos extremadamente complejos a altas velocidades.

Campos-Macías et al. [2020][36] presenta una arquitectura de control para un VANT, con la habilidad de explorar y navegar hacia objetivos, utilizando una representación de octomaps y un

planificador global de tipo RRT.

Zhou et al. [2023][37] presentan una arquitectura descentralizada multi-VANT, hacen uso de una descomposición HGrid para la representación del entorno, logran equilibrar la repartición de tareas formulando la distribución de tareas de exploración como un problema tipo Vehicle Routing Problem. Cada VANT actualiza constantemente la ruta extrayendo información para la planificación de la exploración. Proponen una arquitectura en tres capas (Percepción, Coordinación y Exploración), la generación de trayectoria es basada por curvas de bezier generando trayectorias suaves y seguras en tiempo real.

Exploración

Las bases para la exploración autónoma e innovaciones son heredadas de algoritmos ya empleados en la robótica móvil (tabla 1). Uno de los primeros trabajos en la exploración con robots, es la propuesta de fronteras por Yamauchi[38], donde define como una frontera a la línea entre las zonas exploradas y las no exploradas de un área de interés. Durante la navegación la información percibida por el robot crece, moviendo las fronteras hasta que no existan más fronteras. En el trabajo de Faria et al.[39], se combina la estrategia basada en fronteras con técnicas de planificación de trayectorias Lazy Theta* en un VANT.

Fraundorfer et al.[40] hacen uso de una exploración con fronteras a partir de una navegación autónoma aplicando un algoritmo tipo bug para el seguimiento de una pared, empleando campos de potencial artificial para una planificación local en un mapa de ocupación tipo grid. Estos trabajos demostraron la navegación autónoma de vehículos aéreos no tripulados y que estos pueden seguir puntos de referencia en el mapa, evitar obstáculos y llevar a cabo tareas de exploración en entornos complejos.

Los trabajos de Papachristos et al.[41], Selin et al.[42], Cieslewski et al.[43] se enfocan a la exploración haciendo uso de estrategias basadas en fronteras.

Representación medio ambiente 3D

Los primeros trabajos de VANT autónomos se encuentran en las aportaciones de Shen et al.[44], que hacen uso de un VANT con la propuesta de dos planificadores de trayectorias con un control proporcional con retroalimentación y basados en RRT*, para conseguir una representación del mundo en 2D empleando un sensor tipo LiDAR. Por otra parte, los trabajos de Grzonka et al.[45] también hacen una representación del entorno en 2D, haciendo uso de algoritmos que trabajan en mapas densos tipo grid, y del algoritmo D* lite para la planificación de trayectorias.

Con la llegada de las primeras cámaras capaces de obtener valores de profundidad (RGB-D), y con mayores capacidades de almacenamiento en menos espacio, nos permiten tratar el medio ambiente a través de representaciones tridimensionales. Podemos citar por ejemplo la propuesta de estructura de datos basada en grafos octrees por Meagher[46] con una baja complejidad en el orden logarítmico, por consiguiente en el año 2013 se introdujo un nuevo concepto para la representación de mapas 3D basados en esos principios, haciendo que la representación de entornos 3D se realice de manera eficiente para aplicaciones en robótica donde se necesitan algoritmos rápidos. Los Mapas Volumétricos Probabilísticos (PVM) representan un entorno 3D usado para tareas de navegación autónoma. Los trabajos de Hornung et al.[47] y el Centro Aeroespacial Alemán (DLR) introducen los Octomaps, que se utilizan para representar mapas tridimensionales como subdivisiones marcadas como ocupadas, desocupadas y desconocidas, para aplicaciones de navegación. En recientes trabajos Min et al.[48] proponen dar solución a los cuellos de botella que se presentan en el Octomap buscando acelerar los tiempos de cómputo en la construcción de mapas a partir de la

implementación de Aceleradores Gráficos GPU.

Los trabajos de Cieslewski et al.[43] hacen uso de la representación del entorno por medio de voxel, la versión tridimensional de la rejilla de ocupación empleada en robótica móvil terrestre, para planificar trayectorias de exploración para un VANT, basado en fronteras y con un esquema reactivo de navegación.

Usenko et al.[49] proponen el uso del mapa centrando al robot en un círculo tridimensional de tamaño fijo, plantea el problema de replanificación local como la optimización de una función de costo con un término que penaliza las desviaciones de posición y velocidad de la trayectoria de referencia. La trayectoria de referencia es construida como curva de Bezier uniforme, simplificando el cálculo.

Mohta et al.[50] hacen uso de un mapa híbrido formado con la combinación de un mapa 3D con un mapa global en 2D.

Florence et al.[51] propone un framework para el manejo de datos para mapas 3D a partir de la información de una cámara de profundidad (RGB-D). La propuesta hace uso de una estructura con cuadrículas de ocupación y datos de profundidad. Al consultar sólo las partes necesarias del mapa, muestran una eficiencia y precisión para manejo de mapas 3D.

Diversos trabajos en los que se incluyen a Gao et al.[52], Lin et al.[53], Oleynikova et al. [2018][54], añaden una estructura adicional a su representación del medio ambiente, usando distancias que permiten la evasión de obstáculos de forma segura. Sin embargo estas soluciones pueden ser costosas requiriendo un mayor procesamiento de computo.

Collins and Michael[55] usan una representación local del mapa como un KD-Tree de un mapa representado en voxels, mientras que un grafo topológico representa todo el entorno explorado.

Generación de trayectorias

En el trabajo de Mohta et al.[50] usan un planificador A* en un grafo híbrido con la información 3D y 2D, formulan un problema de programación cuadrática para la generación de trayectorias agregando un término en la función de costo sobre el error entre la trayectoria y los segmentos de línea del camino. La trayectoria se representa como un polinomio de séptimo orden, utilizando un perfil de velocidad trapezoidal.

Lin et al. [2017][53] hacen uso de un planificador global offline para generar rutas, en la navegación usan un planificador local seleccionando las nuevas guías y un algoritmo A* para buscar la distancia mínima hacia esas nuevas guías. Utilizan un polinomio por partes de octavo orden para la representación de la trayectoria.

Papachristos et al. [2017][41] presentan algoritmos para la exploración autónoma, construyendo un árbol aleatorio de exploración rápida RRT a partir de nuevos puntos, buscando el camino que minimice la incertidumbre del robot con los puntos de referencia del mapa, mientras una segunda ejecución del algoritmo RRT encuentra el camino hacia el punto de vista seleccionado minimizando la incertidumbre del robot y los puntos de referencia.

Oleynikova et al. [2018][54] aborda el problema de mínimos locales de la función de potencial empleada para la navegación, agregando objetivos secundarios para escapar de dichos mínimos. Los autores hacen uso de tablas hash que proporcionan una representación del entorno con rápidos

tiempos de inserción y consulta de complejidad constante.

Gao et al. [2018][52] propone un algoritmo para la generación de trayectorias utilizando el método de marcha rápida y los polinomios en la base de Bernstein. Al considerar la dinámica del VANT, garantizan la prevención de colisiones manteniendo movimientos suaves y continuos.

Selin et al. [2019][42] presenta un enfoque para la planificación de una exploración autónoma en entornos tridimensionales a gran escala. La planificación combina la estrategia basada en fronteras con el enfoque de ganancia de información para dirigir al robot a áreas inexploradas, logrando evitar obstáculos a su paso. Mediante un ajuste del plan de exploración en función del tamaño y complejidad del ambiente, muestran su eficiencia maximizando la cobertura de la exploración a medida que minimizan el tiempo de procesamiento.

McGuire et al. [2019][56] presenta una solución de navegación para enjambres de pequeños multi-VANTS que exploran entornos desconocidos sin señal de GPS de forma centralizada. Éste trabajo propone el algoritmo Swarm Gradient Bug (SGBA), que maximiza la cobertura al hacer que los robots se muevan en diferentes direcciones lejos del punto de partida. Los robots navegan por el entorno y enfrentan obstáculos estáticos sobre la marcha mediante la odometría visual y algoritmos tipo BUG para el seguimiento de paredes. Además, se comunican entre sí para evitar colisiones y maximizar la eficiencia de la búsqueda. Para regresar al punto de partida, los robots realizan una búsqueda de gradiente hacia una señal Bluetooth de baja potencia.

Tabla 2: Trabajos relacionados

REFERENCIA	APLICACIÓN	GENERACION MAPA	PLANIFICACION DE RUTA	GENERACION TRAYECTORIA	SENSOR RGB-D	DINAMICA VANT	multi-VANT
Villanueva-Grijalva [2015][57]	N/A	N/A	N/A	Ley de Control	✗	✓	✗
Marín Mota [2016][58]	Exploración	Octomap	Basado en fronteras	Control directo de velocidad	✓	✗	✓
Cieslewski et al. [2017][43]	Exploración	Octomap	Basado en fronteras	Control directo de velocidad	✓	✗	✓
Usenko et al. [2017][49]	Punto Objetivo	Cuadrícula egocéntrica	Offline RRT*	Curvas de Bezier	✗	✗	✓
Mohta et al. [2017][50]	Punto Objetivo	3D-Local y 2D-Global	A*	Programación cuadrática	✗	✗	✓
Lin et al. [2017][53]	Punto Objetivo	3D voxel array TSDF	A*	Optimización cuadrática	✗	✗	✗
Papachristos et al. [2017][41]	Exploración	Octomap	Next Best View Planner (NBVP)	Control directo de velocidad	✗	✗	✗
Oleynikova et al. [2018][54]	Punto Objetivo	Voxel Hashing TSDF	Next Best View Planner (NBVP)	Optimización cuadrática	✓	✓	✓
Gao et al. [2018][52]	Punto Objetivo	Mapa de cuadrícula	Método de marcha rápida	Optimización cuadrática	✗	✗	✓
Florence et al. [2018][51]	Punto Objetivo	Busqueda basada en visibilidad	2D A*	Control predictivo por modelo (MPC)	✓	✓	✓
Selin et al. [2019][42]	Exploración	Octomap	Next Best View Planner (NBVP)	Control directo de velocidad	✗	✗	✗
McGuire et al. [2019][56]	Exploración	NA	Swarm Gradient Bug Algorithm (SGBA)	Control directo de velocidad	✗	✗	✗
Collins and Michael [2020][55]	Punto Objetivo	KD Tree + Mapa en Voxel	Búsqueda en Grafo	Movimientos suaves	✓	✓	✓
Zhang et al. [2020][59]	Punto Objetivo	N/A	Rapidly Exploring Random Trees (RRT)	Movimientos suaves	✓	✓	✓
Campos-Macías et al. [2020][36]	Punto Objetivo	Octree	Rapidly Exploring Random Trees (RRT)	Basado en contornos	✓	✓	✓
Cieslewski [2021][33]	Exploración	Octomap	Basado en fronteras	Control directo de velocidad	✓	✓	✗
Zhou et al. [2023][37]	Exploración	Octomap HGrid	Next Best View Planner (NBVP)	Control directo de velocidad	✓	✓	✓
Bartolomei et al. [2023][60]	Exploración	Octomap	Basado en fronteras	Control directo de velocidad	✓	✓	✓

Referencias

- [1] Hazim Shakhatreh, Ahmad H. Sawalmeh, Ala Al-Fuqaha, Zuochao 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. doi: 10.1109/access.2019.2909530. URL <https://doi.org/10.1109/access.2019.2909530>.
- [2] Syed Agha Hassnain Mohsan, Nawaf Qasem Hamood Othman, Alsharif Li, Yanlong, Mohammed H., and Muhammad Asghar Khan. Unmanned aerial vehicles (uavs): practical aspects, applications, open challenges, security issues, and future trends. *Intelligent Service Robotics*, 16(1), 2023. doi: <https://doi.org/10.1007/s11370-022-00452-4>.
- [3] Lauren Hirsch and Michael J. De La Merced. Fireworks have a new competitor: Drones, Jul 2023. URL <https://www.nytimes.com/2023/07/01/business/dealbook/fourth-of-july-fireworks-drones.html>.
- [4] Intel Ed. What are service robots?, Ago 2023. URL <https://www.intel.la/content/www/xl/es/robotics/service-robot.html>.
- [5] NASA Ed. Mars 2020 mission perseverance rover, Ago 2023. URL <https://www.intel.la/content/www/xl/es/robotics/service-robot.html>.
- [6] William Stone, Nathaniel Fairfield, and George A. Kantor. Autonomous underwater vehicle navigation and proximity operations for deep phreatic thermal explorer (depthx). In G. Griffiths and K. Collins, editors, *Proceedings of Masterclass in AUV Technology for Polar Science*, London, March 2006. Society for Underwater Technology.
- [7] Frank E. Schneider and Dennis Wildermuth. Assessing the search and rescue domain as an applied and realistic benchmark for robotic systems. In *2016 17th International Carpathian Control Conference (ICCC)*. IEEE, May 2016. doi: 10.1109/carpathiancc.2016.7501177. URL <https://doi.org/10.1109/carpathiancc.2016.7501177>.
- [8] Michael Jones, Soufiene Djahel, and Kristopher Welsh. Path-planning for unmanned aerial vehicles with environment complexity considerations: A survey. *ACM Comput. Surv.*, 55(11), feb 2023. ISSN 0360-0300. doi: 10.1145/3570723. URL <https://doi.org/10.1145/3570723>.
- [9] Sujata Sharma and Ritu Tiwari. A survey on multi robots area exploration techniques and algorithms. In *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*. IEEE, March 2016. doi: 10.1109/icctict.2016.7514570. URL <https://doi.org/10.1109/icctict.2016.7514570>.
- [10] Liang Xu, Xianbin Cao, Wenbo Du, and Yumeng Li. Cooperative path planning optimization for multiple uavs with communication constraints. *Knowledge-Based Systems*, 260:110164, 2023. ISSN 0950-7051. doi: <https://doi.org/10.1016/j.knosys.2022.110164>. URL <https://www.sciencedirect.com/science/article/pii/S0950705122012606>.
- [11] W.A. Kamal, D.-W. Gu, and I. Postlethwaite. MILP AND ITS APPLICATION IN FLIGHT PATH PLANNING. *IFAC Proceedings Volumes*, 38(1):55–60, 2005. doi: 10.3182/20050703-6-cz-1902.02061. URL <https://doi.org/10.3182/20050703-6-cz-1902.02061>.
- [12] Brian Geiger, Joseph Horn, Anthony DeLullo, Albert Niessner, and Lyle Long. Optimal path planning of UAVs using direct collocation with nonlinear programming. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, June 2006. doi: 10.2514/6.2006-6199. URL <https://doi.org/10.2514/6.2006-6199>.

- [13] Adel MOKRANE, Amal CHOUKCHOU BRAHAM, and Brahim CHERKI. Uav path planning based on dynamic programming algorithm on photogrammetric Dems. In *2020 International Conference on Electrical Engineering (ICEE)*, pages 1–5, 2020. doi: 10.1109/ICEE49691.2020.9249903.
- [14] Tomáš Rouček, Martin Pecka, Petr Čížek, Tomáš Petříček, Jan Bayer, Vojtěch Šalanský, Teymur Azayev, Daniel Heřt, Matěj Petrлік, Tomáš Báča, Vojtech Spurný, Vít Krátkyý, Pavel Petrářek, Dominic Baril, Maxime Vaidis, Vladimír Kubelka, François Pomerleau, Jan Faigl, Karel Zimmermann, Martin Saska, Tomáš Svoboda, and Tomáš Krajník. System for multi-robotic exploration of underground environments CTU-CRAS-NORLAB in the DARPA subterranean challenge. *Field Robotics*, 2(1):1779–1818, March 2022. doi: 10.55417/fr.2022055. URL <https://doi.org/10.55417/fr.2022055>.
- [15] Jennifer Gielis, Ajay Shankar, and Amanda Prorok. A critical review of communications in multi-robot systems. *Current Robotics Reports*, 3(4):213–225, August 2022. doi: 10.1007/s43154-022-00090-9. URL <https://doi.org/10.1007/s43154-022-00090-9>.
- [16] Saeed H. Alsamhi, Ou Ma, Mohammad Samar Ansari, and Faris A. Almalki. Survey on collaborative smart drones and internet of things for improving smartness of smart cities. *IEEE Access*, 7:128125–128152, 2019. doi: 10.1109/ACCESS.2019.2934998.
- [17] Ravil I. Mukhamediev, Kirill Yakunin, Margulan Aubakirov, Ilyas Assanov, Yan Kuchin, Adilkhan Symagulov, Vitaly Levashenko, Elena Zaitseva, Dmitry Sokolov, and Yedilkhan Amirgaliyev. Coverage path planning optimization of heterogeneous uavs group for precision agriculture. *IEEE Access*, 11:5789–5803, 2023. doi: 10.1109/ACCESS.2023.3235207.
- [18] Vladimir J. Lumelsky and Alexander A. Stepanov. Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape. *Algorithmica*, 2(1-4):403–430, November 1987. doi: 10.1007/bf01840369. URL <https://doi.org/10.1007/bf01840369>.
- [19] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. doi: 10.1109/TSSC.1968.300136.
- [20] A. Stentz. Optimal and efficient path planning for partially-known environments. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 3310–3317 vol.4, 1994. doi: 10.1109/ROBOT.1994.351061.
- [21] Sunan Huang and Rodney Swee Huat Teo. Computationally efficient visibility graph-based generation of 3d shortest collision-free path among polyhedral obstacles for unmanned aerial vehicles. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1218–1223, 2019. doi: 10.1109/ICUAS.2019.8798322.
- [22] 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, 2017. doi: 10.1109/CCDC.2017.7978644.
- [23] Ana Batinovic, Jurica Goricanec, Lovro Markovic, and Stjepan Bogdan. Path planning with potential field-based obstacle avoidance in a 3d environment by an unmanned aerial vehicle. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 394–401, 2022. doi: 10.1109/ICUAS54217.2022.9836159.
- [24] Jinyang Wang, Yuhua Li, Ruixuan Li, Hao Chen, and Kejing Chu. Trajectory planning for UAV navigation in dynamic environments with matrix alignment dijkstra. *Soft Computing*, 26(22):12599–12610, June 2022. doi: 10.1007/s00500-022-07224-3. URL <https://doi.org/10.1007/s00500-022-07224-3>.

- [25] Fei Yan, Yan Zhuang, and Jizhong Xiao. 3d prm based real-time path planning for uav in complex environment. In *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1135–1140, 2012. doi: 10.1109/ROBIO.2012.6491122.
- [26] Yue Li, Wei Han, Yong Zhang, and Wanhui Mu. A method of uav path planning based on an improved rrt algorithm. In *2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC)*, pages 1–5, 2018. doi: 10.1109/GNCC42960.2018.9018916.
- [27] Steven M. LaValle. Rapidly-exploring random trees : a new tool for path planning. *The annual research report*, 1998. URL <https://api.semanticscholar.org/CorpusID:14744621>.
- [28] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research*, 30(7):846–894, June 2011. doi: 10.1177/0278364911406761. URL <https://doi.org/10.1177/0278364911406761>.
- [29] Fan Yang, Chao Cao, Hongbiao Zhu, Jean Oh, and Ji Zhang. Far planner: Fast, attemptable route planner using dynamic visibility update, 2022.
- [30] Marco Tranzatto, Frank Mascarich, Lukas Bernreiter, Carolina Godinho, Marco Camurri, Shehryar Khattak, Tung Dang, Victor Reijgwart, Johannes Loeje, David Wisth, Samuel Zimmermann, Huan Nguyen, Marius Fehr, Lukas Solanka, Russell Buchanan, Marko Bjelonic, Nikhil Khedekar, Mathieu Valceschini, Fabian Jenelten, Mihir Dharmadhikari, Timon Homberger, Paolo De Petris, Lorenz Wellhausen, Mihir Kulkarni, Takahiro Miki, Satchel Hirsch, Markus Montenegro, Christos Papachristos, Fabian Tresoldi, Jan Carius, Giorgio Valsecchi, Joonho Lee, Konrad Meyer, Xiangyu Wu, Juan Nieto, Andy Smith, Marco Hutter, Roland Siegwart, Mark Mueller, Maurice Fallon, and Kostas Alexis. Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge, 2022.
- [31] Bor-Sen Chen and Ting-Wei Hung. Integrating local motion planning and robust decentralized fault-tolerant tracking control for search and rescue task of hybrid uavs and biped robots team system. *IEEE Access*, 11:45888–45909, 2023. doi: 10.1109/ACCESS.2023.3273787.
- [32] Juan Carlos Elizondo-Leal. *Estrategia Descentralizada para la Exploración Multi-Robot, incluyendo Restricciones en Rango de Comunicación*. PhD thesis, CINVESTAV Unidad Tamaulipas, Junio 2013.
- [33] Titus Cieslewski. *Decentralized Multi-Agent Visual SLAM*. PhD thesis, University of Zurich, Febrero 2021. URL https://rpg.ifi.uzh.ch/docs/thesis_Cieslewski_final.pdf.
- [34] R. Brooks. A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, 2(1):14–23, 1986. doi: 10.1109/JRA.1986.1087032.
- [35] Antonio Loquercio, Elia Kaufmann, René Ranftl, Matthias MÅ¼ller, Vladlen Koltun, and Davide Scaramuzza. Learning high-speed flight in the wild. *Science Robotics*, 6(59), oct 2021. doi: 10.1126/scirobotics.abg5810. URL <https://doi.org/10.1126%2Fscirobotics.abg5810>.
- [36] Leobardo Campos-Macías, Rodrigo Aldana-López, Rafael Guardia, José I. Parra-Vilchis, and David Gómez-Gutiérrez. Autonomous navigation of MAVs in unknown cluttered environments. *Journal of Field Robotics*, 38(2):307–326, may 2020. doi: 10.1002/rob.21959. URL <https://doi.org/10.1002/rob.21959>.
- [37] Boyu Zhou, Hao Xu, and Shaojie Shen. Racer: Rapid collaborative exploration with a decentralized multi-uav system. *IEEE Transactions on Robotics*, 39(3):1816–1835, 2023. doi: 10.1109/TRO.2023.3236945.

- [38] B. Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 'Towards New Computational Principles for Robotics and Automation'*, pages 146–151, 1997. doi: 10.1109/CIRA.1997.613851.
- [39] Margarida Faria, António Sérgio Ferreira, Héctor Pérez-Leon, Ivan Maza, and Antidio Viguria. Autonomous 3d exploration of large structures using an UAV equipped with a 2d LIDAR. *Sensors*, 19(22):4849, November 2019. doi: 10.3390/s19224849. URL <https://doi.org/10.3390/s19224849>.
- [40] 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, 2012. doi: 10.1109/IROS.2012.6385934.
- [41] Christos Papachristos, Shehryar Khattak, and Kostas Alexis. Uncertainty-aware receding horizon exploration and mapping using aerial robots. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4568–4575, 2017. doi: 10.1109/ICRA.2017.7989531.
- [42] Magnus Selin, Mattias Tiger, Daniel Duberg, Fredrik Heintz, and Patric Jensfelt. Efficient autonomous exploration planning of large-scale 3-d environments. *IEEE Robotics and Automation Letters*, 4(2):1699–1706, 2019. doi: 10.1109/LRA.2019.2897343.
- [43] Titus Cieslewski, Elia Kaufmann, and Davide Scaramuzza. Rapid exploration with multi-rotors: A frontier selection method for high speed flight. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2135–2142, 2017. doi: 10.1109/IROS.2017.8206030.
- [44] Shaojie Shen, Nathan Michael, and Vijay Kumar. Autonomous multi-floor indoor navigation with a computationally constrained mav. In *2011 IEEE International Conference on Robotics and Automation*, pages 20–25, 2011. doi: 10.1109/ICRA.2011.5980357.
- [45] Slawomir Grzonka, Giorgio Grisetti, and Wolfram Burgard. A fully autonomous indoor quadrotor. *IEEE Transactions on Robotics*, 28(1):90–100, 2012. doi: 10.1109/TRO.2011.2162999.
- [46] Donald Meagher. Geometric modeling using octree encoding. *Computer Graphics and Image Processing*, 19(2):129–147, June 1982. doi: 10.1016/0146-664x(82)90104-6. URL [https://doi.org/10.1016/0146-664x\(82\)90104-6](https://doi.org/10.1016/0146-664x(82)90104-6).
- [47] 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. doi: 10.1007/s10514-012-9321-0. URL <https://octomap.github.io>. Software available at <https://octomap.github.io>.
- [48] Heajung Min, Kyung Min Han, and Young J. Kim. Accelerating probabilistic volumetric mapping using ray-tracing graphics hardware, 2020.
- [49] Vladyslav Usenko, Lukas von Stumberg, Andrej Pangercic, and Daniel Cremers. Real-time trajectory replanning for MAVs using uniform b-splines and a 3d circular buffer. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, sep 2017. doi: 10.1109/iros.2017.8202160. URL <https://doi.org/10.1109/iros.2017.8202160>.
- [50] Kartik Mohta, Michael Watterson, Yash Mulgaonkar, Sikang Liu, Chao Qu, Anurag Makeneni, Kelsey Saulnier, Ke Sun, Alex Zhu, Jeffrey Delmerico, Konstantinos Karydis, Nikolay Atanasov, Giuseppe Loianno, Davide Scaramuzza, Kostas Daniilidis, Camillo Jose Taylor, and Vijay Kumar. Fast, autonomous flight in GPS-denied and cluttered environments.

Journal of Field Robotics, 35(1):101–120, December 2017. doi: 10.1002/rob.21774. URL <https://doi.org/10.1002/rob.21774>.

- [51] Peter R. Florence, John Carter, Jake Ware, and Russ Tedrake. Nanomap: Fast, uncertainty-aware proximity queries with lazy search over local 3d data, 2018.
- [52] Fei Gao, William Wu, Yi Lin, and Shaojie Shen. Online safe trajectory generation for quadrotors using fast marching method and bernstein basis polynomial. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 344–351, 2018. doi: 10.1109/ICRA.2018.8462878.
- [53] Yi Lin, Fei Gao, Tong Qin, Wenliang Gao, Tianbo Liu, William Wu, Zhenfei Yang, and Shaojie Shen. Autonomous aerial navigation using monocular visual-inertial fusion. *Journal of Field Robotics*, 35(1):23–51, July 2017. doi: 10.1002/rob.21732. URL <https://doi.org/10.1002/rob.21732>.
- [54] Helen Oleynikova, Zachary Taylor, Roland Siegwart, and Juan Nieto. Safe local exploration for replanning in cluttered unknown environments for microaerial vehicles. *IEEE Robotics and Automation Letters*, 3(3):1474–1481, jul 2018. doi: 10.1109/lra.2018.2800109. URL <https://doi.org/10.1109/lra.2018.2800109>.
- [55] Matthew Collins and Nathan Michael. Efficient planning for high-speed mav flight in unknown environments using online sparse topological graphs. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11450–11456, 2020. doi: 10.1109/ICRA40945.2020.9197167.
- [56] K. N. McGuire, C. De Wagter, K. Tuyls, H. J. Kappen, and G. C. H. E. de Croon. Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. *Science Robotics*, 4(35):eaaw9710, 2019. doi: 10.1126/scirobotics.aaw9710. URL <https://www.science.org/doi/abs/10.1126/scirobotics.aaw9710>.
- [57] Omar Abraham Villanueva-Grijalva. Control de un vehículo aéreo no tripulado (auv) para reconocimiento en interiores. Master’s thesis, CINVESTAV Unidad Guadalajara, Septiembre 2015. URL <https://repositorio.cinvestav.mx/handle/cinvestav/405>.
- [58] Luis Alfonso Marín Mota. Exploración 3d de interiores mediante múltiples mavs. Master’s thesis, CINVESTAV Unidad Zacatenco, Diciembre 2016. URL <https://www.cs.cinvestav.mx/TesisGraduados/2016/TesisLuisMarin.pdf>.
- [59] Ji Zhang, Chen Hu, Rushat Gupta Chadha, and Sanjiv Singh. Falco, fast likelihood based collision avoidance with extension to human guided navigation. *Journal of Field Robotics*, 37(8), Abril 2020. ISSN 1556-4967. doi: 10.1002/rob.21952. URL <http://dx.doi.org/10.1002/rob.21952>.
- [60] Luca Bartolomei, Lucas Teixeira, and Margarita Chli. Fast multi-uav decentralized exploration of forests. *IEEE Robotics and Automation Letters*, 8(9):5576–5583, 2023. doi: 10.1109/LRA.2023.3296037.