

Centro de Investigación y Estudios Avanzados del IPN

Unidad Tamaulipas

Protocolo de tesis

Título: Estrategias para la exploración coordinada multi-VANT

Candidato: Luis Alberto Ballado Aradias

Asesor: Dr. José Gabriel Ramírez Torres

Co-Asesor: Dr. Eduardo Arturo Rodríguez Tello

14 de agosto de 2023

Resumen

La exploración multi-robot ha surgido como un enfoque prometedor para el mapeo eficiente de entornos desconocidos. Un enfoque colaborativo ofrece una mayor eficiencia de exploración, una obtención de información más rápida y amplias capacidades de cobertura 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 localización, el manejo de mapas y la navegación autónoma.

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 multi-VANT permiten la adquisición simultánea de datos desde múltiples puntos de vista, lo que permite mejorar la generación de mapas de entornos desconocidos. El uso de algoritmos de coordinación inteligente, la toma de decisiones descentralizada mejora la eficiencia de estos sistemas. Además, los avances en los protocolos de comunicación permiten una colaboración fluida, lo que mejora su capacidad para navegar, explorar y adquirir datos de áreas grandes y complejas. Asimismo, la integración de sensores de última generación mejora la precisión y confiabilidad de los sistemas multi-VANT en varios dominios, incluida la gestión de desastres, la agricultura de precisión, la inspección de infraestructura y la vigilancia militar [1,2] o en espectaculares animaciones aéreas [3]. Dichas aplicaciones suelen carecer de autonomía. 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 y navegar dentro de él.

El enfoque de este trabajo es la propuesta de una arquitectura de software capaz de coordinar múltiples vehículos aéreos no tripulados (VANTS) con habilidades para la exploración, generación de mapas de áreas desconocidas y planificación de rutas para explorar eficientemente un área de interés. Este problema implica tomar 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 son considerados en este trabajo.

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

Datos Generales

Título de proyecto

Estrategias para la exploración coordinada multi-VANT

Datos del alumno

Nombre: Luis Alberto Ballado Aradias
Matrícula: 220229860003
Dirección: Juan José de La Garza #909
Colonia: Guadalupe Mainero C.P. 87130
Teléfono (casa): +52 (833) 2126651
Teléfono (lugar de trabajo): +52 (834) 107 0220 + Ext
Dirección electrónica: luis.ballado@cinvestav.mx
URL: https://luis.madlab.mx

Institución

Nombre: CINVESTAV-IPN
Departamento: Unidad Tamaulipas
Dirección: Km 5.5 carretera Cd. Victoria - Soto la Marina.
Parque Científico y Tecnológico TECNOTAM,
Ciudad Victoria, Tamaulipas, C.P. 87130
Teléfono: (+52) (834) 107 0220

Beca de tesis

Institución otorgante: CONAHCYT
Tipo de beca: Maestría Nacional
Vigencia: Septiembre 2022 - Agosto 2024

Datos del asesor

Nombre: Dr. José Gabriel Ramírez Torres
Dirección: Km. 5.5 carretera Cd. Victoria - Soto la Marina
Parque Científico y Tecnológico TECNOTAM
Ciudad Victoria, Tamaulipas, C.P. 87130
Teléfono (oficina): (+52) (834) 107 0220 Ext. 1014
Institución: CINVESTAV-IPN
Departamento adscripción: Unidad Tamaulipas
Grado académico: Doctorado en Mecánica

Nombre: Dr. Eduardo Arturo Rodríguez Tello
Dirección: Km. 5.5 carretera Cd. Victoria - Soto la Marina
Parque Científico y Tecnológico TECNOTAM
Ciudad Victoria, Tamaulipas, C.P. 87130
Teléfono (oficina): (+52) (834) 107 0220 Ext. 1100
Institución: CINVESTAV-IPN
Departamento adscripción: Unidad Tamaulipas
Grado académico: Doctorado en Informática

Descripción del proyecto

El proyecto se centra en la colaboración de múltiples vehículos aéreos no tripulados (VANTS) para tareas de exploración, con el objetivo de desarrollar y evaluar una arquitectura de software descentralizada en el que varios vehículos aéreos no tripulados trabajen independientes y autónomos para explorar entornos desconocidos de manera eficiente.

Antes de profundizar en los detalles del proyecto, es fundamental definir qué es un VANT. Un VANT, también conocido como dron o recientemente como sistema aéreo no tripulado (UAS), se refiere a una aeronave que opera sin un piloto humano a bordo. Los VANTS están equipados con varios sensores, sistemas de comunicación y computadoras a bordo que les permiten operar de forma autónoma o bajo control remoto. Estos aviones pueden ser de diferentes tamaños, desde pequeños modelos hasta máquinas comerciales o militares más grandes[4].

En el contexto del proyecto de colaboración multi-VANT para tareas de exploración descentralizado, estos VANTS autónomos se utilizarán para navegar y explorar entornos desconocidos. Al trabajar juntos de manera coordinada, los VANTS compartirán información, tareas y recursos, optimizando el proceso de exploración para una mayor eficiencia y una cobertura ~~integral~~.

El proyecto tiene como objetivo desarrollar algoritmos, protocolos y estrategias que permitan que estos VANTS se comuniquen de manera efectiva, asignen tareas, eviten obstáculos y exploren en colaboración dado un entorno.

Al aprovechar el potencial de la colaboración multi-VANT, el proyecto tiene como objetivo contribuir a los avances en las técnicas de exploración distribuida con agentes autónomos y expandir las aplicaciones potenciales de los VANTS en varios dominios.

Antecedentes y motivación para el proyecto

Los robots de servicio se están convirtiendo rápidamente en una parte esencial de las empresas ~~centradas en el servicio~~ que buscan formas innovadoras de atender a los clientes mientras mejoran sus resultados de productividad. Los **robots de servicio** generalmente se utilizan para ayudar a los empleados en sus tareas diarias para que puedan concentrarse en actividades más importantes [5]. Entre ellos los VANTS, se han vuelto cada vez más frecuentes verlos en el mundo actual, encontrando aplicaciones en una amplia gama de industrias.

En fotografía y video aéreas, 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, la fumigación de precisión, mejorando la productividad y 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 así los riesgos y costos asociados con las inspecciones manuales. En misiones de búsqueda y rescate, donde ayudan en la localización de personas desaparecidas o en evaluaciones posteriores a un desastre, los VANTS han demostrado ser muy útiles.

La mayoría de estas aplicaciones son sencillas, estáticas, en espacios controlados con rutas pre-determinadas. Para aplicaciones más complejas, donde el robot debe responder de manera autónoma (con mínima intervención humana) a los cambios del medio ambiente, 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[6], la exploración en aguas profundas[7] o áreas afectadas por desastres[8], donde la presencia humana puede no ser segura, permitiéndoles acceder a lugares de difícil acceso [9]. La exploración con robots amplía nuestro conocimiento e impulsa la innovación.

Algunos desarrollos importantes en estas áreas de investigación se han centrado principalmente en sistemas con un único robot. No se puede subestimar la importancia de utilizar múltiples robots en las actividades de exploración. Estos sistemas de múltiples robots ofrecen mayores beneficios que mejoran la efectividad y la eficiencia en este tipo de tareas como lo es la exploración. Múltiples robots permiten la cobertura simultánea de un área más grande, lo que permite una exploración eficiente del entorno [10].

Un sistema multi-VANT puede colaborar, intercambiar información y optimizar sus rutas para minimizar la redundancia y agilizar el proceso de exploración. Además, el uso de múltiples VANTS mejora la solidez de la misión, siendo tolerante en caso de fallas. Si un VANT encuentra dificultades, otros VANTS pueden continuar la exploración, asegurando la continuidad de la misión y reduciendo el riesgo de falla. Además, los sistemas multi-VANT permiten la especialización de tareas, donde diferentes VANTS pueden equiparse con sensores o instrumentos especializados para recopilar datos específicos.

La escalabilidad y adaptabilidad de los sistemas multi-VANT los hace adecuados para actividades de exploración en varios escenarios y entornos, que van desde misiones de pequeña escala a gran escala o complejas.

El uso de sistemas multirobot traen consigo retos inherentes que deben abordarse. 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 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. Es necesario implementar algoritmos para evitar colisiones y estrategias de planificación de rutas para permitir movimientos seguros y eficientes de los robots, especialmente al explorar espacios complejos y desordenados. Por otra parte, la integración y fusión de datos de múltiples robots 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.

La planificación de rutas presenta diversos retos:

- Restricciones físicas del robot (su geometría o forma)
- Dinámica del robot
- Incertidumbres de lecturas de sensores (ruido)

Para crear rutas seguras, debemos respetar las restricciones para que el robot pueda ejecutar los movimientos 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 será en optimizar varios parametros por los 6 DoF (Grados de libertad) que cuenta buscando algoritmos que corran en tiempo real (que se ejecuten rápido) dentro de dis-

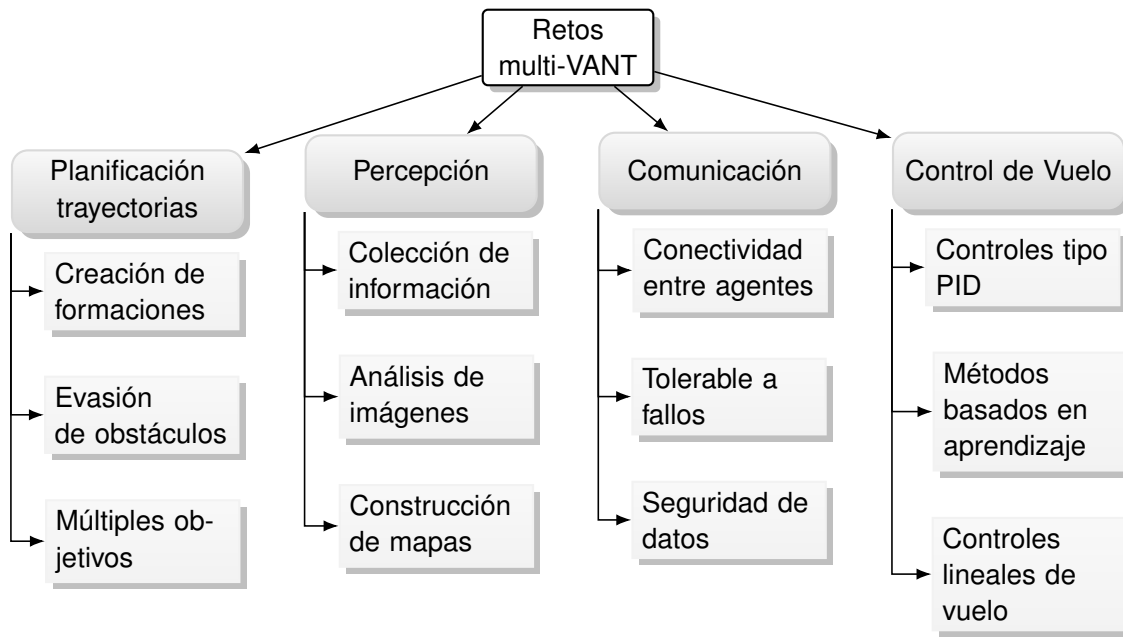


Figura 1: Ilustra los retos multi-VANT

positivos computacionales limitados.

Xu et al. [2023][11] mencionan que la planificación de trayectorias para múltiples VANTS es inherente a lo complejo del entorno y las trayectorias que pueda tomar el VANT. La minimización de la longitud de las rutas, configuraciones que pueda realizar el VANT y la seguridad del trayecto para todos los multi-VANT durante el vuelo son partes clave cuando se crea un planificador multi-VANT. Apesar de las aproximaciones los planificadores globales se deben descomponer para poder considerar la existencia de obstáculos haciendo que la comunicación entre ellos pueda ser afectada. En últimas décadas se han propuesto diversas técnicas de programación (Mixed Integer Linear Programming (MILP), Nonlinear programming (NP) y Dynamic Programming (DP)) teniendo estos métodos una fuerte base en teoría matemática. A pesar de ello estos métodos de programación matemática su escala computacional crece exponencialmente conforme el espacio de búsqueda.

Otros métodos que han sido ampliamente trabajados son los de Campo de Potencial Artificial ampliamente usado como planificador de trayectorias por sus ventajas en tiempo real. Desafortunadamente cuando existen dos campos de repulsión causados por obstáculos son iguales, este método cae en mínimos locales llegando a fallar en encontrar una solución.

Diversas técnicas de Inteligencia Computacional se han propuesto para el problema de planificación de trayectorias (Algoritmos Genéticos GA, Ant Colony Optimization (ACO) Zhao [2020] [12], Particle Swarm Optimization (PSO) y Evolución Diferencial (DE). Estos algoritmos han demostrado crear rutas navegables para los VANT y son ampliamente usados para problemas de planificación de rutas complejos. Trabajos de Deng et al. [2023] [13] han realizado adaptaciones al algoritmo PSO mostrando mejores resultados evitando caer en mínimos locales con ayuda de Algoritmos Genéticos (GA) considerando parámetros como inercia, funciones de activación para la probabilidad de cruce y mutación, mostrando mejorar a rutas rápidas y estables en una ejecución off-line.

Planteamiento del problema

Desarrollar una estrategia de exploración multi-VANT que reduzca el tiempo total de exploración dado un conjunto de \mathcal{V} vehículos aéreos no tripulados. Las capacidades limitadas de energía y sensores abordo de los VANT les permiten navegar de forma autónoma. Teniendo en cuenta sus limitaciones de energía y la necesidad de una exploración eficiente, el objetivo es determinar la trayectoria, las rutas y la asignación de tareas óptimas.

El espacio de todas las posibles configuraciones, esta compuesto por los espacios libres (C_{free}) y espacios ocupado (con obstáculos) C_{obs} .

Sea $\mathcal{W} = \mathbb{R}^3$ el mundo, $\mathcal{O} \in \mathcal{W}$ el conjunto de obstáculos, $\mathcal{A}(q)$ las configuraciones del robot $q \in \mathcal{C}$

- $C_{free} = \{q \in \mathcal{C} | \mathcal{A}(q) \cap \mathcal{O} = \emptyset\}$
- $C_{obs} = \mathcal{C} \setminus C_{free}$

donde $\mathcal{W} = \mathbb{R}^3$ es el espacio de trabajo del robot, $\mathcal{O} \in \mathcal{W}$ es el conjunto de obstáculos, y $\mathcal{A}(q)$ son las configuraciones del robot $q \in \mathcal{C}$.

La solución debe tener en cuenta los obstáculos, los entornos dinámicos, las limitaciones de comunicación y la coordinación entre los VANTS para evitar colisiones. Para lograr una exploración eficiente y completa con un tiempo y recursos mínimos, el problema requiere la creación de algoritmos y técnicas de optimización.

Completar la exploración significa que el robot pueda crear un mapa \mathcal{M} que cubre el volumen \mathcal{V} y los puntos en el mapa. Por la naturaleza del problema, esto se debe resolver de forma rápida sin tiempos de espera.

La función objetivo variará según los objetivos específicos del problema.

- Maximizar la cobertura del área de interés C
- Minimizar el tiempo total requerido para cubrir el área de interés C
- Maximizar la cantidad de información recolectada

Con base en lo anterior, surgen las siguientes preguntas de investigación:

- ¿Qué acciones deberán de realizar los VANTS para explorar el espacio completo lo más rápido posible?
- ¿Cuáles son los mejores algoritmos adecuados para correr en una tarjeta electrónica con recursos limitados?
- ¿Qué tan seguros estaremos que un nuevo VANT a la misión llegue a la frontera y aporte a la misión?

Hipótesis

La eficiencia de exploración y la cobertura de un área objetivo se pueden mejorar empleando un enfoque coordinado, colaborativo y descentralizado. El sistema multi-VANT puede lograr una exploración más completa a través de la asignación efectiva de tareas, la planificación de la

trayectoria y la coordinación. La hipótesis asume que la integración de múltiples VANTS con diversas capacidades conducirá a mejores resultados de exploración, incluida una mayor cobertura de área, una mejor recopilación de datos y un rendimiento general mejorado en comparación con un enfoque de un solo VANT.

Objetivos generales y específicos del proyecto

General

Diseñar una arquitectura de software descentralizada capaz de resolver los problemas de localización, mapeo, navegación y coordinación multi-VANT en ambientes desconocidos y dinámicos para tareas de exploración en interiores.

De manera más específica, se listan los siguientes objetivos:

1. **Construcción propuesta** Evaluar las soluciones en la literatura asociados con la coordinación multi-VANT. Enfocándose en aspectos como la comunicación, evasión de obstáculos, asignación de tareas y sincronización de información. Basándose a esta valoración, construir una arquitectura de software para la coordinación multi-VANT.
2. **Valoración (prueba) propuesta** Emplear una herramienta de simulación de libre uso para robótica, para el desarrollo y puesta en marcha de una propuesta de arquitectura de software capaz de realizar el control multi-VANT y evaluar el desempeño de dicha arquitectura.
3. **Comparación y análisis** Comparar y analizar los resultados obtenidos con enfoques existentes en la coordinación multi-VANT, mostrando las ventajas y desventajas de la estrategia propuesta. Con base a estos análisis proponer recomendaciones y pautas prácticas para la implementación y aplicación de la estrategias de coordinación multi-VANT en escenarios reales, considerando factores como la escalabilidad, la robustez y los recursos computacionales requeridos.

Metodología

La metodología propuesta se divide en tres etapas, iniciando en septiembre del 2023 y terminando en agosto del 2024. A continuación se detallan cada una de las actividades que se plantean realizar en cada una.

Etapa 1. Análisis y diseño de la solución propuesta

En esta etapa se comprende en la revisión de la literatura de manera más completa, que permita contar con la información necesaria para la elección de los mejores algoritmos para abordar cada una de las problemáticas asociadas con la coordinación de trayectorias. Una vez realizada la elección de los algoritmos que se usarán para la propuesta de arquitectura de software, se procederá a revisar y estudiar las arquitecturas para los robots colaborativos. Finalmente, se realizará el diseño de la arquitectura.

Las actividades específicas a realizarse en la etapa 1, son:

E1.A1. Revisión estado del arte Ampliar la revisión de la literatura sobre coordinación y exploración multi-VANT.

E1.A2. Evaluación de aptitudes Revisar y documentar los aspectos relevantes (asi como sus limitantes) que permiten la colaboración, coordinación y balanceo de la carga de trabajo multi-VANT.

E1.A3. Selección de algoritmos Seleccionar los algoritmos para planificación de trayectorias y exploración en ambientes desconocidos representativos para un entorno de computación restringida.

E1.A4. Elaboración de solución Definir la arquitectura de software para escenarios en aplicaciones multi-VANT apegadas a las especificaciones de computadora de placa reducida (Raspberry Pi, Esp32 ... etc.).

E1.A5. Documentación Etapa 1 Elaborar la documentación de la revisión del estado del arte y del trabajo realizado que formará parte de la tesis.

E1.A6. Revisión de tesis Etapa 1 Revisión y corrección de avances con los asesores.

Etapa 2. Implementación y validación

Esta etapa se centra en el desarrollo e implementación del diseño de la arquitectura de software para la coordinación multi-VANT.

Las actividades específicas a realizarse en la etapa 2, son:

E2.A1. Selección Simulador Al tener definida la arquitectura de software y conocer las estructuras de datos que se utilizaran, evaluar los diversos simuladores para robótica de libre uso. (Revisar temas de modelos 3D, dinámica del robot, representación del ambiente 3D, simulación de sensores).

E2.A2. Visualización de datos Conocer las herramientas para la visualización y telemetría y creación de un modelo 3D de acuerdo al simulador seleccionado.

E2.A3. Control de desplazamientos Crear movimientos y control de un VANT y múltiples VANTS, algoritmos que forman parte de la capa reactiva del VANT.

E2.A4. Desarrollo de algoritmos de exploración De acuerdo con la revisión del estado del arte se implementará el algoritmo propuesto para la exploración con un VANT

E2.A5. Implementación un solo VANT Realizar pruebas y corregir errores con base a los desarrollos realizados.

E2.A6. Simulación un solo VANT Realizar pruebas de simulación con un solo VANT de la solución propuesta.

E2.A7. Desarrollo de coordinación Al contar con la exploración y navegación exitosa de un solo VANT, se procede al desarrollo de coordinación multi-VANT.

E2.A8. Implementación multi-VANT Realizar pruebas y corrección de errores con base a los desarrollos realizados para la coordinación multi-VANT.

E2.A9. Simulación multi-VANT Realizar pruebas de simulación multi-VANT de la solución propuesta.

E2.A10. Documentación Etapa 2 Elaborar la documentación del desarrollo e implementación de la propuesta de arquitectura de software para la coordinación multi-VANT que formará parte de la tesis.

E2.A11. Revisión de tesis Etapa 2 Revisión y corrección de capítulos con los asesores.

Etapla 3. Evaluación experimental, resultados y conclusiones

Partiendo del prototipo y las simulaciones desarrolladas en la etapa anterior, en esta etapa se realizan todas las actividades relacionadas con la evaluación, recabación de resultados y la escritura de los capítulos restantes de la tesis. Además se realizará el proceso de graduación y actividades relacionadas.

Las actividades específicas a realizarse en la etapa 3, son:

E3.A1. Experimentación de solución Experimentos para evaluar el desempeño de la solución propuesta creada en la etapa anterior.

E3.A2. Recopilación de resultados Recabar la información de los resultados, realizar su análisis y generar la documentación correspondiente.

E3.A3. Documentación Etapa 3 Elaborar la documentación de los resultados obtenidos y conclusiones que formará parte de la tesis.

E3.A4. Revisión de tesis Revisión y corrección de tesis con los asesores.

E3.A5. Divulgación De acuerdo a los progresos dentro de la tesis, se estará en total disposición a espacios donde se pueda hacer divulgación científica dentro del estado cubriendo los requisitos de retribución social de la institución.

E3.A6. Proceso de titulación Comenzar el proceso de titulación.

Infraestructura

Para el desarrollo de este proyecto de investigación, se hará uso de un equipo de cómputo con las siguientes características:

- iMac (21.5-inch, Late 2015)
- Procesador 2.8 GHz Quad-Core Intel Core i5
- Memoria Ram 8 GB 1867 MHz DDR3
- Graphics Intel Iris Pro Graphics 6200 1536 MB
- Almacenamiento 1 TB

Cronograma de actividades (plan de trabajo)

	Cuatrimestre 1 ^a				Cuatrimestre 2 ^b				Cuatrimestre 3 ^c			
	1	2	3	4	1	2	3	4	1	2	3	4
Etap 1												
E1.A1. Revisión literatura relevante ^d												
E1.A2. Selección de algoritmos												
E1.A3. Diseño de la arquitectura de software												
E1.A4. Documentación Etapa 1												
E1.A5. Revisión de tesis Etapa 1												
Etap 2												
E2.A1. Selección Simulador												
E2.A2. Visualización de datos ^e												
E2.A3. Control de desplazamientos ^f												
E2.A4. Desarrollo de algoritmo de exploración												
E2.A5. Implementación y simulación ^g												
E2.A6. Desarrollo de coordinación												
E2.A7. Implementación y simulación ^h												
E2.A8. Documentación Etapa 2												
E2.A9. Revisión de tesis Etapa 2												
Etap 3												
E3.A1. Experimentación de solución												
E3.A2. Recopilación resultados												
E3.A3. Documentación Etapa 3												
E3.A4. Revisión de tesis												
E3.A5. Divulgación ⁱ												
E3.A6. Proceso de titulación												

^aCorrespondiente a los meses de Septiembre, Octubre, Noviembre, Diciembre del 2023

^bCorrespondiente a los meses de Enero, Febrero, Marzo, Abril del 2024

^cCorrespondiente a los meses de Mayo, Junio, Julio, Agosto del 2024

^dRevisión de alertas de trabajos relacionados sobre la exploración y colaboración multi-VANT, evaluación de aptitudes en trabajos recientes

^eVisualización Octomap en Simulador

^fUn VANT

^gSe considera un solo agente que resuelva la tarea de exploración autónoma con evasión de obstáculos

^hSe considerarán los múltiples-VANT que resuelva la tarea de exploración autónoma con evasión de obstáculos

ⁱAbierto a espacios de divulgación de acuerdo con las actividades de retribución social

Estado del arte

Las aplicaciones de la robótica 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 un comportamiento inteligente. Dentro de la robótica móvil inteligente se han propuesto estrategias de comportamiento reactivas, algoritmos que imitan el comportamiento de insectos y el cómo se desplazan en un entorno.

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, minimizando la colisión y mejorando la eficiencia de sus misiones. Estos algoritmos tienen en cuenta varios factores, como las restricciones del robot y las ubicaciones del objetivo, para generar trayectorias seguras y eficientes.

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 [1987][14], dieron respuesta a problemáticas de navegación eficiente y de poca memoria (Algoritmos tipo bug).

Matemáticamente el problema de encontrar rutas es resuelto con grafos, siendo un grafo una representación matemática de vértices y aristas. Siendo el vértice la posición del robot y las aristas un camino donde encontramos los trabajos de Hart et al. [1968][15], al mejorar el algoritmo de Dijkstra para el robot Shakey, que debía navegar en una habitación que contenía obstáculos fijos. El objetivo principal del algoritmo A^* es la eficiencia en la planificación de rutas. Otros algoritmos propuestos por Stentz [1994][16] han demostrado operar de manera eficiente ante obstáculos dinámicos, a comparación del algoritmo A^* que vuelve a ejecutarse al encontrarse con un obstáculo, el algoritmo D^* usa la información previa para buscar una ruta hacia el objetivo. La propuesta de LaValle [1998][17] del algoritmo RRT son ampliamente usados 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, uniendo el árbol existente. Las modificaciones al algoritmo RRT por Karaman and Frazzoli [2011][18] incorporando una heurística de costo por recorrer, permite encontrar rutas casi óptimas de manera eficiente. Siendo ampliamente usado en problemas de navegación autónoma y planificación de movimiento.

Recientes trabajos de Yang et al. [2022][19], siguen demostrando la capacidad de implementación de algoritmos como los grafos de visibilidad para tareas en entornos conocidos y no conocidos haciendo una representación del ambiente usando polígonos, logrando un rápido planificador que también resuelve los obstáculos nuevos en el ambiente con mayores resultados a comparación de estrategias como A^* , D^* e inclusive RRT*.

Además de la planificación de rutas, la coordinación de múltiples robots requiere una comunicación efectiva. 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 [2023][20] 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.

Método	Completez	Óptimo	Escalable	Notas
Grafo de visibilidad	✓	✓	✗	<ul style="list-style-type: none"> ■ Mucho espacio libre ■ Mala escalabilidad ■ El robot pasa cerca de obstáculos
Diagramas de Voronoi	✓	✗	✗	<ul style="list-style-type: none"> ■ Espacio libre máximo ■ Rutas conservadoras ■ Mala escalabilidad
Campos de potencial artificial	✓	✗	Depende del ambiente	<ul style="list-style-type: none"> ■ Fácil de implementar ■ Suceptible a mínimos locales
Dijkstra/A*	✓	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	✓	Grafo	✓	<ul style="list-style-type: none"> ■ Eficiente para problemas con consultas múltiples ■ Completez probabilística ■ Camino irregular
RRT	✓	✗	✓	<ul style="list-style-type: none"> ■ Eficiente para problemas de consulta única ■ Completez probabilística ■ Camino irregular

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

Entre los trabajos más relevantes se encuentra la tesis doctoral de Elizondo Leal [2013][21] que tiene como objetivo general una estrategia e implementación de la coordinación de múltiples

robots móviles con un enfoque de auto-ofertas. Trabajos con la coordinación de robots para el problema de box pushing proponiendo una nueva estrategia inspirada en el algoritmo frente de onda. Por otra parte trabajos de tesis de maestría de Sandoval García [2013][22] cuyos objetivos son la generación de mapas fotográficos utilizando vehículos aéreos no tripulados de baja altitud.

Otras investigaciones relevantes se encuentran en las tesis del CINVESTAV Unidad Guadalajara. La tesis de maestría de Villanueva Grijalva [2015][23] se centra en las posibilidades de navegación autónoma de un VANT, por otra parte el trabajo de Campos-Macías et al. [2020][24], tiene como objetivo de una propuesta de arquitectura para un VANT en tareas de exploración, dicho trabajo no cubre la cooperación multi-VANT.

El campo de exploración con múltiples vehículos aéreos no tripulados es un área nueva y con mucho crecimiento en los últimos años. Una variedad de preguntas para permitir la exploración autónoma se han formulado, desde la planificación de rutas para múltiples robots en tareas de exploración Sharma and Tiwari [2016][10], estrategias para la coordinación y protocolos de comunicación. Diversos estudios multi-VANT se han realizado para tareas como el monitoreo ambiental Alsamhi et al. [2019][25], agricultura de precisión Mukhamediev et al. [2023][26] y operaciones de búsqueda y rescate Shakhathreh et al. [2019][1].

La dirección en que apunta el estado del arte, se puede atribuir a los avances en tecnología en la última década. Investigadores de diversas áreas, que incluyen las ciencias computacionales e ingeniería han contribuido al crecimiento de éste campo. Las bases para la exploración autónoma e innovaciones son heredadas de algoritmos ya empleados en la robótica móvil. [ver cuadro 1]

Una de los trabajos pioneros en la exploración con robots, es la propuesta de fronteras Yamauchi [1997][27]. Donde establece que una frontera es la línea entre las zonas exploradas y las no exploradas dado 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. [2019][28], combina la estrategia basada en fronteras con técnicas de planificación de trayectorias Lazy Theta* en un VANT.

Trabajos como el de Tranzatto et al. [2022][29] han logrado optimizar problemas de alta dimensionalidad con el control de navegación para un robot con cuatro patas, haciendo uso de aprendizaje por refuerzo con ayuda de simulaciones corriendo en paralelo en un cuarto simulado, logrando obtener los pesos que le ayudan a resolver el problema de navegación, pero al momento de pasar a efectuar un despliegue de software el robot no pudo hacer un paso correcto. Los huecos entre la simulación y la realidad debido a los anchos de banda que sufren las lecturas de sensores, teniendo una comunicación deficiente en la arquitectura. Tomando en cuenta los ruidos estocásticos y realizando simulaciones híbridas han logrado ganar el DARPA Subterranean Challenge[30] usando una exploración basada en grafos y un mapa de ocupación (OctoMap) para simular el entorno tridimensional.

Navegaciones interesantes Loquercio et al. [2021][31] que propone una arquitectura con capas de proyección, decisión y control posterior a un procesamiento de imagen con uso de algoritmos para la estimación de un mapa, lograrán demostrar que pueden navegar en entornos extremadamente complejos a altas velocidades haciendo uso de arquitecturas de tipo sensor, mapear, planear.

La organización de software asociada con los niveles de control para un robot tiene el nombre de Arquitectura de Software. Existen diversas capas de control. En niveles bajos de control queremos que los movimientos del robot sean estables sin oscilaciones, que no colisionen con obstáculos al mismo tiempo tener una estabilidad en sus movimientos. Esperamos que el comportamiento

autónomo resuelva aspectos como moverse al mismo tiempo evadir obstáculos. Arquitecturas de software que permiten este tipo de control regularmente se ejecutan en paralelo y son conocidas como behavior-based architectures Arkin [1998][32].

Los controles de bajo nivel responden muy bien a técnicas de teoría de control Ramirez and Zeghloul [2001][33]. Considerando como entrada del sistema la posición que deseamos y su orientación. El error será la diferencia entre la posición deseada y la posición actual. Es necesario que la retroalimentación de este tipo de controles sea de alta velocidad para evitar que los errores aumenten a lo largo del tiempo evitando la inestabilidad.

A lo largo del desarrollo de la robotica móvil se han demostrado que estrategias de control basadas en comportamientos (behavior-based) presentan mejores desempeños Brooks [1986][34]. El robot sensa su entorno y reacciona con los comportamientos requeridos. Factores como estos, aumentan la autonomía y solucionan los problemas comunes como el de evitar obstáculos.

Es común pensar que la ingeniería de control y el control biológico presentan diversas similitudes. Por un lado los sistemas en ingeniería tienen un valor de referencia, pudiendo describirlos como sistemas lineales. Los controles biológicos no son lineales. Bajo las inspiraciones de la naturaleza se han propuesto algoritmos heurísticos que buscan aprovechar el comportamiento de la naturaleza para intereses propios de optimización.

Las metaheurísticas Bio-inspiradas son una clase de algoritmos de optimización inspirados de sistemas y procesos biológicos que nos ayudan a 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 cruce, 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, su adopción en el campo de la robótica ha sido 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 e iteraciones para converger en una solución óptima, lo que puede no ser factible en aplicaciones de robótica en tiempo real. 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 deterministas:** 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.
- **Optimización basada en modelos:** muchos problemas de robótica se pueden resolver de manera efectiva utilizando técnicas de optimización basadas en modelos. Con modelos dinámicos conocidos y restricciones ambientales, los métodos basados en modelos, como el control óptimo o la optimización de la trayectoria, pueden proporcionar soluciones analíticas o numéricas con un rendimiento garantizado. Estos enfoques pueden explotar la estructura del problema y las restricciones específicas, lo que lleva a soluciones más eficientes y confiables en comparación con las metaheurísticas de propósito general.
- **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 grandes poblaciones o extensas iteraciones para 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.

La adquisición de datos es el primer paso en la representación de mapas 3D con VANTS. Los VANTS pueden llevar a cabo vuelos sobre un área de interés, capturando imágenes desde diferentes ángulos y alturas. Estas técnicas aprovechan la información de correspondencia entre las imágenes para calcular la posición y orientación relativa de las cámaras y reconstruir la estructura tridimensional del entorno.

Los VANTS pueden utilizar sensores LiDAR (Light Detection and Ranging) para capturar datos 3D. Los sensores LiDAR emiten pulsos de luz láser y miden el tiempo que tarda en reflejarse en los objetos. Esto permite obtener información precisa sobre la distancia y la posición tridimensional de los objetos en el entorno. Los datos de un sensor de tipo LiDAR pueden combinarse con las imágenes capturadas de una cámara para generar mapas 3D completos y detallados.

Los primeros trabajos multi-VANT se encuentran en las aportaciones de Shen et al. [2011][35] que hacen uso de un VANT con la propuesta de dos planificadores de trayectorias con un control proporcional con retroalimentación y basados en RRT* haciendo una representación del mundo en 2D a base de un sensor tipo LiDAR, por otra parte los trabajos de Grzonka et al. [2012][36] también hacen una representación del entorno en 2D haciendo usos de algoritmos que trabajan en mapas densos tipo grid, hacen uso del algoritmo D* lite para su planificación de trayectorias. Fraundorfer et al. [2012][37] hacen uso de una exploración con fronteras a partir de una navegación autónoma aplicando un algoritmo tipo bug para el seguimiento de una ped, hacen uso de 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.

Con la llegada de las primeras cámaras capaces de obtener valores de profundidad (RGB-D), mayores capacidades de almacenamiento en menos espacio nos permiten ver el entorno como realmente es \mathbb{R}^3 . Con la propuesta de estructura de datos basada en grafos octrees por Meagher [1982][38] con una baja complejidad en el orden logaritmico. En 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 Volumetricos Probabilisticos (PVM) representan un entorno 3D usado para tareas de navegación autónoma. Los trabajos de Hornung et al. [2013][39] y el Centro Aeroespacial Alemán(DLR) introducen los OctoMaps que se utiliza para representar mapas tridimensionales ocupados y desconocidos en entornos de robótica y navegación. Hacen uso del Octree para modelar la probabilidad de ocupación del espacio. En recientes trabajos Min et al. [2020][40] proponen dar solución a los cuellos de botella que se presentan en el OctoMap buscando acelerar los tiempos de computo en la construcción de mapas a partir de la implementación de Aceleradores Gráficos GPU. Obteniendo resultados superiores a los reportados a la fecha.

Los trabajos de Cieslewski et al. [2017][41] hacen uso de representación del entorno con ayuda de voxel 3D, planifican trayectorias de exploración para un VANT basado en fronteras con un VANT utilizando un modo reactivo generando una ruta hacia las nuevas fronteras a explorar. Para las fronteras lejanas en el rango del sensor la velocidad es máxima hacia el área desconocida, en caso contrario para fronteras cercanas que la velocidad es menor

Usenko et al. [2017][42] proponen el uso del mapa centrando al robot en un círculo tridimensional de tamaño fijo, plantea el problema como replanificación local como la optimización de una función de costo con un término que penaliza las desviaciones de posición y velocidad de la trayectoria. La trayectoria local es representada por una curva de Bezier uniforme, simplificando el cálculo. Hacen uso de un paquete de optimización no lineal

Mohta et al. [2017][43] hacen uso de un mapa híbrido formado con la combinación de un mapa 3D con un mapa global en 2D, usan un planificador A^* en un grafo híbrido con la información 3D y 2D, formulan una programación cuadrática para la generación de trayectorias agregando un término en la función de costo entre la trayectoria y los segmentos de línea del camino. La trayectoria se representa como un polinomio de séptimo orden, para la asignación de tiempo a cada segmento utilizan los tiempos ajustando un perfil de velocidad trapezoidal a través de los segmentos

Lin et al. [2017][44] 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][45] presentan algoritmos para la exploración autónoma, hacen una exploración 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][46] aborda el problema de quedarse en mínimos locales agregando

objetivos, 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][47] hacen uso de distancias euclidianas para facilitar la información de distancia de los obstáculos resultando costosas de procesar en tiempo real, propone reducir la trayectoria dentro del espacio libre con restricciones, plantean una programación cuadrática (QP) utilizando una base de Bernstein para representar la trayectoria como curvas de Bezier por partes,

Florence et al. [2018][48] hacen uso de un mapa global 2D para guiar la exploración basada en consultad de proximidad, hacen uso de un planificador 2D con el algoritmo A*, hacen uso de una primitiva de movimiento 3D que maximiza el progreso euclidiano hacia el objetivo considerando las probabilidades de colisión

Selin et al. [2019][49] hacen uso del algoritmo RRT insertando valores altos a los vertices con mayores ganancias de información que son usados como objetivos de planificación de rutas.

McGuire et al. [2019][50] presenta una solución de navegación mínima para enjambres pequeños multi-VANTS que exploran entornos desconocidos sin señal de GPS de forma centralizada. Éste trabajo propone un algoritmo Swarm Gradient Bug (SGBA), una solución de navegación mínima que permite a un enjambre de diminutos VANTS explorar autonomamente un entorno desconocido y regresar posteriormente al punto de partida. SGBA 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.

Collins and Michael [2020][51] 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

En recientes trabajos Cieslewski [2021][52] ha demostrado descentralizar la tarea de SLAM para la creación de mapas en tareas de exploración eliminando el bloque de optimización haciendo uso de técnicas de machine learning teach and repeat.

Campos-Macías et al. [2020][24] presenta una arquitectura para un VANT con la habilidad de explorar y navegaciones hacia objetivos con ayuda se su propia representación de octomaps y planificador global de tipo RRT.

Zhou et al. [2023][53] presentan una arquitectura descentralizada multi-VANT, hacen uso de una descomposición HGrid para la representación del entorno, logran equilibrar la repartición de tareas formulando el problema de 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.

REFERENCIA	MAPA	Planificador de rutas	Generación trayectoria	MULTI-VANT
Cieslewski et al. [2017][41]	Octomap	Basado en fronteras	Control directo de velocidad	✗
Usenko et al. [2017][42]	Cuadrícula egocéntrica	Offline RRT*	Curvas de Bezier	✗
Mohta et al. [2017][43]	mapa 3D-Local y 2D-Global	A*	Progración cuadrática	✗
Lin et al. [2017][44]	3D voxel array TSDF	A*	Optimización cuadrática	✗
Papachristos et al. [2017][45]	Octomap	NBVP	Control directo de velocidad	✗
Oleynikova et al. [2018][46]	Voxel Hashing TSDF	NBVP	Optimización cuadrática	✗
Gao et al. [2018][47]	Mapa de cuadrícula	Método de marcha rápida	Optimización cuadrática	✗
Florence et al. [2018][48]	Busqueda basada en visibilidad	2D A*	Control MPC	✗
Selin et al. [2019][49]	Octomap	NBVP	Control directo de velocidad	✗
McGuire et al. [2019][50]	NA	SGBA	Control directo de velocidad	✗
Collins and Michael [2020][51]	KD Tree + Mapa en Voxel	Búsqueda en Grafo	Movimientos suaves	✗
Campos-Macías et al. [2020][24]	Octree	RRT	Basado en contornos	✗
Zhou et al. [2023][53]	Octomap HGrid	NBVP	Control directo de velocidad	✓

Cuadro 2: Trabajos relacionados

Contribuciones o resultados esperados

1. Documentación, y códigos liberados
 - Algoritmo para la exploración multi-VANT
 - Algoritmo para la planificación de rutas
 - Algoritmo para crear formaciones
 - Protocolos de comunicación y coordinación multi-VANT
2. Simulación de solución
 - Simulaciones detalladas en diversos escenarios 3D
 - Métricas como tiempo de respuesta, consumo de energía y la capacidad de adaptación a diferentes escenarios.
3. Tesis impresa.

Fecha de inicio

Fecha de terminación

Septiembre de 2023

Agosto de 2024

Firma del alumno: _____

Comité de aprobación del tema de tesis

Dr. José Gabriel Ramírez Torres

Dr. Eduardo Arturo Rodríguez Tello

Dr. Ricardo Landa Becerra

Dr. Mario Garza-Fabre

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] Konstantinos Dalamagkidis. Definitions and terminology. In *Handbook of Unmanned Aerial Vehicles*, pages 43–55. Springer Netherlands, August 2014. doi: 10.1007/978-90-481-9707-1_92. URL https://doi.org/10.1007/978-90-481-9707-1_92.
- [5] Intel Ed. Fireworks have a new competitor: Drones, Ago 2023. URL <https://www.intel.la/content/www/xl/es/robotics/service-robot.html>.
- [6] NASA Ed. Mars 2020 mission perseverance rover, Ago 2023. URL <https://www.intel.la/content/www/xl/es/robotics/service-robot.html>.
- [7] 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.
- [8] 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/carpethiancc.2016.7501177. URL <https://doi.org/10.1109/carpethiancc.2016.7501177>.
- [9] 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>.
- [10] 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>.
- [11] 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>.
- [12] Hong Zhao. Optimal path planning for robot based on ant colony algorithm. In *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pages 671–675, 2020. doi: 10.1109/IWCMC48107.2020.9148277.

- [13] Lixia Deng, Huanyu Chen, Xiaoyiqun Zhang, and Haiying Liu. Three-dimensional path planning of uav based on improved particle swarm optimization. *Mathematics*, 11(9), 2023. ISSN 2227-7390. doi: 10.3390/math11091987. URL <https://www.mdpi.com/2227-7390/11/9/1987>.
- [14] 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>.
- [15] 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.
- [16] 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.
- [17] 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>.
- [18] 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>.
- [19] Fan Yang, Chao Cao, Hongbiao Zhu, Jean Oh, and Ji Zhang. Far planner: Fast, attemptable route planner using dynamic visibility update, 2022.
- [20] 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.
- [21] Juan Carlos Elizondo Leal. *Estrategia Descentralizada para la Exploración Multi-Robot, incluyendo Restricciones en Rango de Comunicación*. PhD thesis, CINVESTAV Unidad Tamaulipas, Junio 2013.
- [22] Humberto Sandoval García. Generación de mapas utilizando vehículos aéreos no tripulados de baja altitud. Master’s thesis, CINVESTAV Unidad Tamaulipas, Febrero 2013.
- [23] Omar Abraham Villanueva Grijalva. Control de un vehículo aéreo no tripulado (auv) para reconocimiento en interiores. Master’s thesis, CINVESTAV Unidad Guadalajara, Septiembre 2015.
- [24] 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>.
- [25] 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.
- [26] 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.

- [27] 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.
- [28] 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>.
- [29] 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.
- [30] Tomáš Rouček, Martin Pecka, Petr Čížek, Tomáš Petříček, Jan Bayer, Vojtěch Šalanský, Teymur Azayev, Daniel Heřt, Matěj Petrlik, Tomáš Báča, Vojtech Spurný, Vít Krátký, Pavel Petráček, 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>.
- [31] Antonio Loquercio, Elia Kaufmann, René Ranftl, Matthias Müller, 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/scirobotics.abg5810>.
- [32] Ronald C. Arkin, editor. *Behavior-Based Robotics*. MIT Press, 1998. doi: 10.1007/978-3-319-32552-1. URL <https://mitpress.mit.edu/9780262529204/behavior-based-robotics/>.
- [33] Gabriel Ramirez and Said Zeghloul. Collision-free path planning for nonholonomic mobile robots using a new obstacle representation in the velocity space. *Robotica*, 19(5):543–555, 2001. doi: 10.1017/S0263574701003484.
- [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] 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.
- [36] 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.
- [37] 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.

- [38] 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).
- [39] 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>.
- [40] Heajung Min, Kyung Min Han, and Young J. Kim. Accelerating probabilistic volumetric mapping using ray-tracing graphics hardware, 2020.
- [41] 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.
- [42] 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>.
- [43] 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>.
- [44] 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>.
- [45] 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.
- [46] 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>.
- [47] 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.
- [48] Peter R. Florence, John Carter, Jake Ware, and Russ Tedrake. Nanomap: Fast, uncertainty-aware proximity queries with lazy search over local 3d data, 2018.
- [49] 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.

- [50] 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>.
- [51] 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.
- [52] 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.
- [53] 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.