Jornadas de Automática

Efficient Coordination of Heterogeneous Unmanned Aerial Systems in Cooperative Surveillance

E. Kouchaki[a], L. García-Junceda[b], M.A. de Frutos[b], J.R. Martínez-de Dios[a,*], A. Ollero[a]

[a]GRVC Robotics Laboratory, University of Seville, 41092 Seville, Spain
[b]UAV Navigation S.L, 28703, San Sebastián de los Reyes, Madrid, Spain.

To cite this article: E. Kouchaki, L. García-Junceda, M.A. de Frutos, J.R. Martínez-de Dios, A. Ollero, 2024. Efficient Coordination of Heterogeneous Unmanned Aerial Systems in Cooperative Surveillance. Jornadas de Automática, 45. https://doi.org/10.17979/ja-cea.2024.45.10935

Resumen

Este artículo aborda la coordinación y planificación de rutas de múltiples Sistemas Aéreos no Tripulados (UAS) heterogéneos en misiones cooperativas de vigilancia y búsqueda y rescate. El sistema forma parte de un contrato de transferencia de tecnología. Eficiencia, robustez, flexibilidad, reconfigurabilidad y escalabilidad son sus principales requisitos. El esquema propuesto se compone de dos módulos. El primero resuelve el problema *Vehicle Routing Problem* y asigna a cada UAS una lista ordenada de puntos de interés a visitar de modo que se minimice el tiempo total de la misión. El segundo módulo determina una ruta segura y factible para cada UAS minimizando la desviación respecto su ruta evitando zonas de exclusión aérea y cumpliendo las restricciones cinemáticas del UAS. El desempeño presentado muestra el potencial del esquema en aplicaciones como logística, vigilancia y gestión de desastres, entre otras.

Palabras clave: Unmanned Aerial Systems (UAS), Aplicaciones.

Abstract

This paper addresses the coordination and path planning of multiple heterogeneous Unmanned Aerial Systems (UAS) in cooperative surveillance and search & rescue missions. The system is part of a technology transfer contract: efficiency, robustness, flexibility, reconfigurability, and scalability are its main requirements. The proposed scheme is composed of two modules. The first solves the Vehicle Routing Problem and assigns each UAS with an ordered list of points of interest (PoIs) to be visited such that the total mission time is minimized. The second module determines a safe and feasible path for each UAS minimizing the deviation w.r.t. its route while avoiding no-fly zones (NFZs) and fulfilling the UAS kinematic constraints. The presented performance shows the scheme's potential for real-world missions in applications such as logistics, surveillance, and disaster management, among others.

Keywords: Unmanned Aerial Systems (UAS), Applications.

1. Introduction

The cooperation of multiple heterogeneous Unmanned Aerial Systems (UAS) enables exploiting a wide variety of advantages and synergies. Using several UAS equipped with complementary payload sensors (e.g., visual and infrared cameras) enables increasing the mission fulfillment performance such as surveillance or search & rescue. If the UAS are equipped with the same type of sensor, using multiple UAS largely reduces the mission fulfillment times enabling higher mission execution frequencies. However, the flexible and efficient coordination of multiple heterogeneous UAS in real environments with strict as no-fly zones (NFZs) imposed by UAS regulations is not an easy task. That is the objective of the U-PLAN system, that is being developed by the GRVC Robotics Lab from the Univ. de Sevilla in the framework of project ”U-SCUAR: Investigación Avanzada de UAS en el Ámbito de

*Autor para correspondencia: jdedios@us.es

la Categoría Específica", funded by CDTI through the 2022 Aeronautical Technological Plan (PTA). The optimal coordination of a fleet of vehicles has been traditionally addressed as a Vehicle Routing Problem (VRP) (Adewumi and Adeleke, 2018). However, extending this problem to UAS introduces additional complexities, including moving in 3D, the kinematic constraints of UAS, flight endurance or communication constraints, and the need to avoid NFZs and obstacles. In addition, the coordination method should be: computationally efficient to enable dynamic replanning; robust and; scalable in terms of the number of points of interest (PoIs) and the complexity of the mission and environment.

This paper summarizes the design and first prototype of U-PLAN, a system for the efficient and safe coordination of multiple heterogeneous UAS. Two types of missions are considered: visiting a number of PoIs and searching a given area of interest. The proposed scheme is divided in two layers. In the first module, VRP solvers are used to assign each UAS of the fleet with an ordered list of PoIs. In the second module the route for each UAS is used to generate a safe and feasible path, an ordered list of waypoints (WPs), that ensures no collision with NFZs and the fulfillment of the kinematic constraints of the UAS, while reducing the deviation of the path w.r.t. the route assigned to that UAS. This paper summarizes the design of the proposed scheme, its implementation, and extensively validates its modules and the full scheme in different missions.

This paper is organized into the following sections. The main related work is summarized in Section 2. The problem definition, main hypotheses, and general description are presented in Section 3. Section 4 presents the methods and Section 5 summarizes some results. Finally, Section 6 closes the paper with the conclusions and future work.

2. Related Work

A good number of methods have been developed for solving VRP with UAS. Some of the most recent and relevant are summarized. Work Gu et al. (2022) proposed a hierarchical solution for the general VRP with drones in which there were combined truck-UAS routes for a multi-visit task. The optimization model is constructed based on both the UAS flight time and the payload. They decomposed a combined truck–UAS route into the truck segment and UAS segment. Work Saleu et al. (2022) extended the problem of parallel UAS scheduling traveling salesman to parallel UAS and multiple trucks problem by considering several vehicles. Work Corberán et al. (2024) studied the multi-depot UAS routing problem by considering capacity and time constraints. To deal with the complexity, the authors digitized the problem instances presenting an integer formulation for the problem. The authors in Kuiper and Nadjm-Tehrani (2006) present the criteria that characterize desirable mobility properties for the movement of UAS in a reconnaissance scenario, and compare two mobility models for the scenario. There are also several papers performing Coverage Path Planning (CPP) using UAS. Work Cabreira et al. (2019) is a survey that aims to explore and analyze the existing studies related to the different approaches employed in CPP. The surveyed coverage approaches are classified according to a classical taxonomy, such as no decomposition, exact cellular decomposition, and approximate cellular decomposition. The proposed scheme utilizes OR-Tools (Furnon and Perron, 2024) for solving the VRP providing a robust and flexible framework for a wide range of VRP problems. Unlike the hierarchical solutions, which focus on decomposition, OR-Tools provides a more integrated and scalable framework that can handle complex routing problems without the need for decomposition. Additionally, OR-Tools' flexibility in handling the constraints and its ease of use make it an ideal choice for implementing advanced VRP solutions.

Many different trajectory and path planning methods for UAS have been developed. Work Li et al. (2019) employed the inscribed circle (IC) smooth method for smoothing the trajectory of a multi-UAS system. Vertices are smoothed by using a circle with a specific radius set on each vertex and tangent to both paths. The method had a good performance in generating smooth paths for reducing fuel consumption. Work Huang et al. (2016) proposed k-degree smoothing, in which the path is smoothed by establishing four ICs instead of one within the angle of a vertex and passing on three segmented curved paths formed by the perimeter of the circles. However, the main deficiency of these circle-based methods is that the resultant trajectory doesn't pass exactly through the vertex and is not appropriate for the exact visiting points of interest. Work He et al. (2021) leveraged the method of B-Spline to make the generated path, flyable for UAS in a complex three-dimensional environment. Although the interpolation-based methods such as B-Spline, generate paths that hit exactly the PoI, their produced trajectories have a large deviation from the original straight route. Work Zhen et al. (2018) used the Dubins curve to connect the waypoints generated by an ant colony algorithm in multi-UAS cooperative mission planning. Dubins method is a circle-based approach in which the circle passes through the vertex. With it, they could satisfy the UAS maneuverability constraint and reduce the paths length. Inspired by Dubins method, we propose a circle-based path smoothing algorithm in which the circles are defined as NFZs and built symmetrically on the vertices. By applying the proposed path planning method, the paths are smoothed and have lower deviations from the original routes.

3. General Description

The objective of the proposed scheme is to coordinate a set of heterogeneous UAS for the optimal accomplishing of two types of missions: visiting a number of PoIs and searching a given area of interest. The output of the proposed scheme is a set of safe and feasible paths (one for each UAS) that optimally solve visiting or search missions. The UAS are assumed heterogeneous in terms of kinematics and also in terms of energy consumption and endurance. The scenario can contain NFZs that constrain the UAS paths. The NFZs are assumed cylindrical with either circular or polygonal cross-sections. The following hypotheses are considered: i) UAS are heterogeneous and each one has its specifications kinematic constraints and flight duration which all are known, ii) the start and end points of the routes can be different for each UAS (the so-called multi-depot problem), and iii) the PoIs can have different altitudes. The UAS paths should be computed to minimize the total mission time. The mission is considered completed when all the PoIs have been visited.

The mission definition includes the specification of the PoIs, the definition of NFZs, the number of UAS, and the flight specifications and kinematics of each UAS. The proposed scheme is decomposed into two main sub-problems, see Figure 1: 1) multi-UAS route generation and 2) path planning. The first module determines the route for each UAS, the ordered list of PoIs that should be visited by each UAS. All the PoIs should be visited minimizing the mission time. The optimal routes then need to be converted to feasible paths considering the kinematics of each UAS. Also, there are possibly also NFZs in the flight area that need to be avoided by UAS. The second module determines a smooth and safe path for each UAS to be able to perform its route while avoiding NFZs and fulfilling the kinematic constraints of that UAS. The deviation of the UAS path from the route should be as low as possible to keep the optimality of the route. This module utilizes the A^* algorithm for both route smoothing and NFZ avoidance. The output is the path for each UAS expressed as an ordered set of WPs for each UAS.

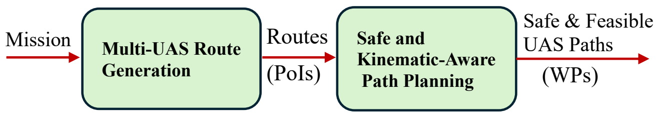

Figura 1: General scheme of the proposed architecture.

4. Methods

4.1. Multi-UAS Route Computation

The Multi-Unmanned Aerial Systems (Multi-UAS) route computation problem is a form of VRP that aims to produce a set of routes for a fleet of UAS to visit some PoIs in an optimal way subject to the constraints of each UAS including its battery life and maximum velocity. In this paper, we use this method for both searching area and visiting point problems. The employed problem formulation is as follows. Given a set of N UAS: $\mathcal{U} = \{U_1, U_2, ..., U_N\}$, a set of M visiting points: $\mathcal{P} = \{P_1, P_2, ..., P_M\}$, a set of $2N$ start depots (Ds) and end depots (De): $\mathcal{D} = \{Ds_1, Ds_2, ..., Ds_N, De_1, De_2, ..., De_N\}$, the goal is to determine a set of routes (one for each UAS): $\mathcal{R} = \{R_1, R_2, ..., R_N\}$ where each R_i represents the computed route of UAS U_i such that every point is visited by only one UAS and the total distance of routes is minimized. The problem is formulated as:

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{i,j\in\mathcal{P}} c_{ij}x_{ij} \\
\text{subject to} \quad & \sum_{j\in\mathcal{P}\setminus 0} x_{0j} = 2N \\
& \sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad (j \in \mathcal{P}\setminus\mathcal{D}) \\
& \sum_{\substack{i\in\mathcal{S}, j\notin\mathcal{S} \\ or\, i\notin\mathcal{S}, j\in\mathcal{S}}} x_{ij} \geq 2b(\mathcal{S}) \qquad (\mathcal{S} \subset \mathcal{P}\setminus\mathcal{D}) \\
& x_{i,j} = 0 \; or \; 1 \qquad (i, j \in \mathcal{P}\setminus\mathcal{D}) \\
& x_{0j} = 0,\ 1 \; or \; 2 \qquad (j \in \mathcal{P}\setminus\mathcal{D})
\end{aligned}
\tag{1}
$$

In that expression, c_{ij} is the cost of the route from node i to node j, x_{ij} is the number of times the path from node i to node j is traversed, $b(\mathcal{S})$ is a lower bound on the number of UAS needed to cover the area S.

A good number of approaches have been proposed in the literature to solve the VRP (Adewumi and Adeleke, 2018). There are also a variety of toolboxes that are available for this purpose such as MATLAB, CVX (CVX Research, 2012), GUROBI (Gurobi Optimization, LLC, 2023), and OR-Tools. In this research, we use OR-Tools, which is an open source toolbox in different languages developed by Google. OR-Tools uses a first solution strategy and then local search to solve VRP as an NP-hard problem. We chose OR-Tools in this work since: i) it is open source and multi-language (C++, .NET, Java, Python); ii) it is easy to provide a graphical interface (compatible with Google Maps); iii) high flexibility to adapt the problem (e.g., add or remove constraints); iv) flexibility for dynamic re-planning and; v) does not require a license if a small number of instances are made.

To implement the OR-Tools for the route computation, the mission definition is fed as a file containing the addresses of PoIs, number of UAS, maximum flight time, and Ground Speed (GS) of each UAS. The solver outputs N number of routes corresponding one by one with the set of UAS which are optimal based on the distance i.e. total flight time.

4.2. Kinematics-Aware Path Planning

After each UAS has been assigned with a list of ordered PoIs, this module determines a safe and feasible path that can be performed by that UAS. The objective is to determine a smooth path on each route for the UAS to be able to perform considering its kinematic constraints avoiding the NFZs that might be within some calculated routes of the route computing module. We utilize a method based on the A^* algorithm to fulfill these two objectives.

Many path planning methods have been proposed including Dijkstra's algorithm (Dhulkefl et al., 2020), Breadth-First Search (BFS), and Depth-First Search (DFS) (Sharma et al., 2017), among others. Also, probabilistic methods trajectory planning algorithms, such as Rapidly-exploring Random Trees (RRT) (Zammit and Van Kampen, 2022), have been employed to navigate autonomously through large spaces. However, the trajectories planned by probabilistic methods can vary significantly from one execution to another, resulting in potentially inconsistent navigation. In this work we preferred to use A^*, which is a powerful approach that can find an optimal solution (Cai et al., 2019). It is efficient and hence enables replanning in dynamic environments and is very widely used. Its advantages in our research can be listed as: i) *optimality and completeness*, since it guarantees to find the shortest path with an admissible heuristic, ii) *efficiency*, as it uses heuristics to guide the search, reducing the number of nodes explored, and iii) *ease of implementation* as in the Euclidean distance can be considered as a heuristic.

First, the NFZs are avoided. For each computed route, the intersection condition is checked for a path between each pair of PoIs and all NFZs defined in the area. If the intersection exists, a graph with bidirectional edges is built on the PoIs on two ends of the path as the start and end nodes and the peripheral nodes of the NFZ. In the case of NFZs with polygonal cross-sections, the peripheral nodes are the vertices. In the case of NFZs with circular cross-sections, the number of adequate points is distributed on its perimeter uniformly to shape the graph accurately. Then A^* algorithm is applied to the graph to find the shortest path. The resulting computed path accordingly does not transverse the NFZ. The pseudocode of the algorithm is shown in Algorithm 1.

Algorithm 1: NFZ avoidance

```
for every optimal route do
    for every path in route do
        i ← start PoI of path
        j ← end PoI of path
        if intersect with NFZs do
            build a graph on i, j, and all peripheral nodes
            run A* algorithm on the graph
            New WPs ← nodes that are on the optimal path
            add New WPs to path
```

Second, the resulting safe paths are smoothed considering the kinematic constraints of the UAS using the capability of A^* in finding the optimal path subject to NFZs. Many methods have been proposed in the literature for path smoothing. Among them, the interpolation-based methods like B-Splines and cubic Splines (Song et al., 2017) and Dubins curves (Ravankar et al., 2018) are widely used due to their capacity of producing continuous and differentiable paths, accuracy, and ease of implementation. However, in our problem, these approaches generate in many cases paths that have a large deviation from the original optimal route computed by OR-Tools. In addition, these deviations can cause the violation of NFZs, making the path unsuitable. On the other hand, the Dubins method smooths the vertices of the path by defining circles on vertices and smoothing the path over them. Although this method produces paths with lower deviations from the optimal routes than the Spline methods, the deviation is still significant under some circumstances, see Figure 5.

For route smoothing, we adopt a method that utilizes A^*. For each route vertex (PoI) a circle is used as an artificial NFZ such that the PoI lies on its perimeter. The center of the circle is set on the bisector of the sharper angle between two paths connected in the PoI as shown in Figure 2. The radius of each circle is specified based on the value of the angle of the corresponding vertex and also the minimum turning radius that the UAS can have during the mission. The following relation exists between the minimum turning radius (R_{min}) of a fixed-wing UAS and its maximum roll angle (ϕ_{max}):

$$R_{min} = \frac{V_G^2}{g \tan \phi_{max}}, \tag{2}$$

where V_G is ground speed and g is the gravitational constant.

In addition, to have a consistent trajectory in all vertices and a minimum deviation from routes, we make a direct proportional relation between the circle radius on the i^{th} PoI (r_i) and the angle between two intersecting paths (θ_i). The selected value of the radius is

$$r_i = \text{máx}[K\theta_i, R_{min}], \tag{3}$$

where K is the proportional constant.

After the construction of the graph, the A^* algorithm finds the optimal paths that make a smooth trajectory. It is shown in Section 5.2 that the adopted approach generates smooth trajectories with lower deviation from the original routes than methods based on Splines and Dubins. The pseudocode for the algorithm is shown in Algorithm 2.

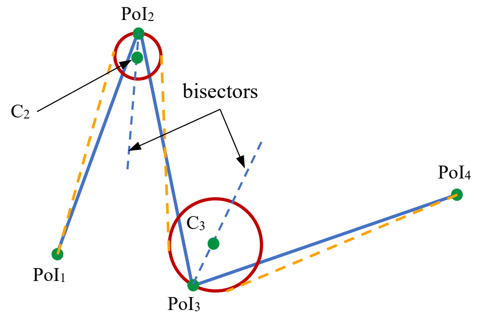

Figura 2: Path smoothing using the proposed method. The original route is in solid lines. The computed paths after smoothing, are in dashed lines. The artificial NFZ are in red. C_2 and C_3 are centers of the circles on the bisectors

Algorithm 2: Route smoothing

```
for every optimal route do
    for every path in route except the last one do
        i ← start WP of path
        j ← end WP of path
        if j ∈ set of POIs do
            build an artificial NFZ on j using Eq. (3) and Fig.2
              apply Alg. 1 to get New WPs
              add New WPs to Path
```

5. Validation

The proposed scheme was implemented in Python. For visualization, we use Google Maps using Map JavaScript API. The scheme was performed on a Processor: 11th Gen Intel(R) Core(TM) i7 @3.60GHz, 3600 Mhz, 8 Core(s), 16 LP. In this section, first the results of each part of Section 4 are presented individually and then the overall architecture is evaluated.

5.1. *Multi-UAS Route Generation*

We consider a mission in which 3 heterogeneous UAS is considered to visit a total of 15 PoIs. The specifications of UAS are presented in Table 1. The PoIs have a variation of different altitudes in the range between 100 m and 200 m. The method described in Section 4.1 is executed and the resulting routes for each UAS are shown in Figure 3 with different colors. The route computation time was 0.02 s. In this case, the total time to complete the mission is determined by the longest route (assigned to UAS3), which is 13 min and 33 s.

Tabla 1: Adopted UAS specifications.

	Velocity Km/h	Battery life min	Max. roll angle degree
UAS1	130	350	28
UAS2	135	300	28
UAS3	130	300	28

The envisioned missions consider less than 5 UAS and 30 PoIs. However, a scalability study is performed to evaluate the behavior of the method with different numbers of PoIs and UAS. Each execution was executed three times and the average execution time was considered. The results are shown in

Figure 4. Although the execution time increases with the number of PoIs and of UAS, the method has good scalability and provides moderate computational times for its feasible implementation, especially if we consider the envisioned missions.

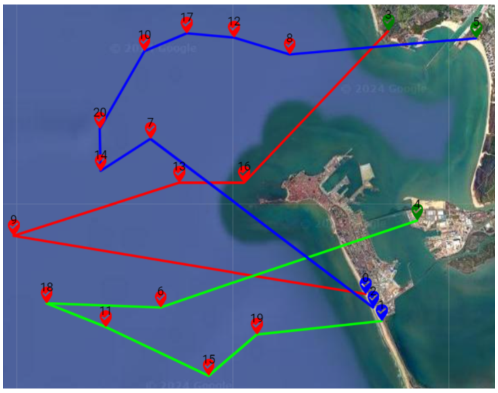

Figura 3: Computed routes for a mission with 3 UAS and 15 PoIs (red markers): red color for the routes for UAS1, green for UAS2, and blue for USA3. The blue and green markers represent respectively the start and end depots.

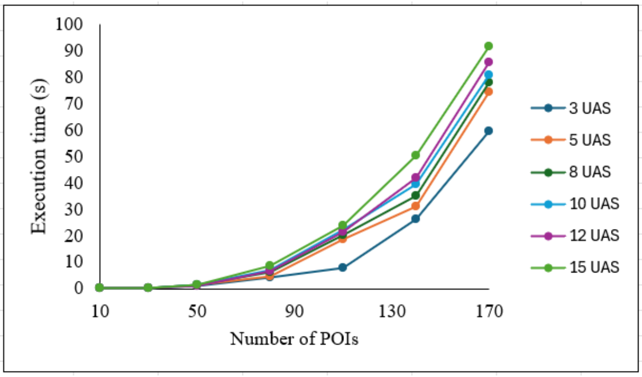

Figura 4: Variation of execution time of route computing with the variation of number of UAS and number of PoIs

5.2. Path Planning

To evaluate the adopted path planning method, its performance is compared with two counterpart approaches widely used in the literature, namely B-Spline interpolation and Dubins. To this end, a single-UAS problem with 6 PoIs including depots is designed and path smoothing is performed on the route using the three mentioned methods. Results are shown in Figure 5. The path generated by B-Spline has a very smooth curvature but a significant higher deviation from the original route than the other two methods. Both Dubins and the developed method perform smoothing only on vertices. They are similar in the wide angles but our method displays better results in sharp angles, which distributes the deviation uniformly on both sides of each angle, resulting in a globally lower deviation. The same conclusions have been obtained in all the routes analyzed, confirming the lower deviation of our method.

Next, we evaluate the developed NFZ avoidance method in Algorithm 1 using different NFZs with different cross-section shapes including polygons and circles in different locations. Figure 6 shows the obtained results. In all cases, the developed algorithm determined the shortest path between two points while avoiding the NFZs. The execution time of the path planning module for each case in these experiments was less than 0.1 s in the >500 experiments performed, which was considered more than sufficient for the problem.

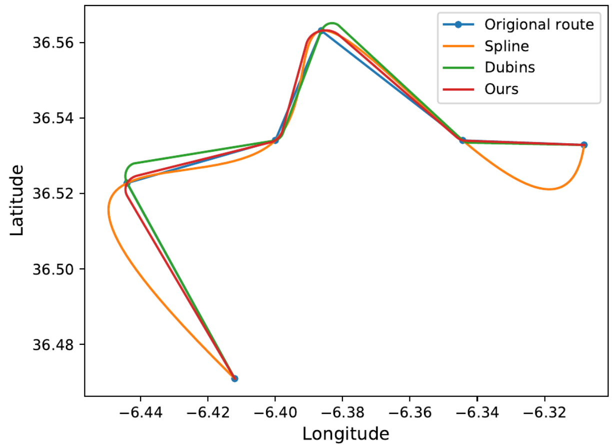

Figura 5: Comparison of the B-Spline, Dubins, and the developed path smoothing methods in one exemplary route.

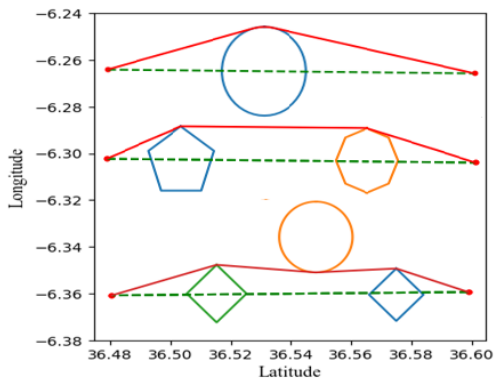

Figura 6: Resulting paths generated by the NFZ avoidance methods with several NFZs with different shapes. The dashed lines are the straight path between each PoI pair and the solid lines are the computed paths.

5.3. Full Multi-UAS System

Next, the performance of the full scheme in fulfilling two different types of missions is analyzed. Figure 7 shows an environment with two NFZs with circular cross-sections and 5 PoIs to be visited using two UAS. The specifications of the UAS are those shown for UAS1 and UAS2 in Table 1. The resulting paths of each UAS are shown in Figure 7. The routes are optimal in terms of minimum mission accomplishment time. UAS1 took the longest time to complete its route, a total of 13 min and 22 s. The paths avoid the NFZs and are smoothed to fulfill the kinematic constraints of the UAS. The proposed scheme took 2.85 s to complete all involved computations.

The proposed scheme can be also applied to search area missions, i.e. missions that aim to coordinate the UAS in searching in a specified area. In this example, a polygonal area at an altitude of 500 m above the ground is searched by UAS1 and UAS2 from Table 1. First, this searching area problem is transformed to a VRP problem. For that, several PoIs are distributed within the area uniformly, based on the fields of view of the UAS's sensors and altitudes. Next, the proposed scheme is straightforwardly applied as described in Section 4. Figure 8 shows the obtained paths. The PoIs are divided between UAS1 and UAS2 to minimize the total mission duration, and the path planning method smooths the paths, particularly at the corners. The time to complete the longest route in this simulation is 29 *min* and 21 *s*. The computational time required by the proposed scheme was 12 s. Similar performance was achieved in the > 300 tests performed with different scenarios, number of UAS, PoIs, and NFZs, evidencing the strong robustness of the proposed scheme.

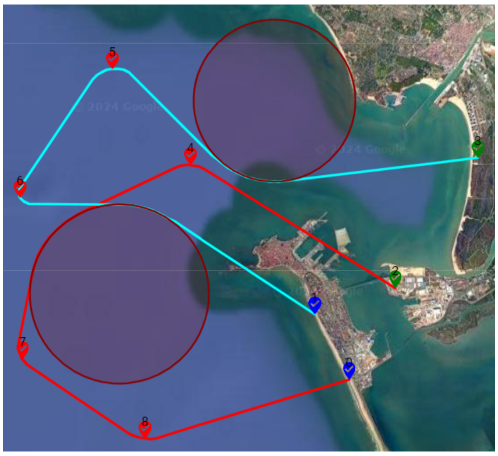

Figura 7: Resulting paths for each UAS for fulfilling the specified mission: red color to represent the path for UAS1, and blue for that for USA2. The start and end depots are shown as blue and green markers respectively. The red markers represent the mission PoIs.

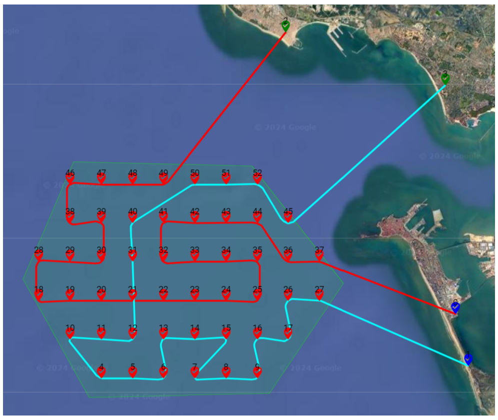

Figura 8: Use of the presented scheme for search area missions: red route by UAS1, and blue route by USA2. The start and end depots are represented as blue and green markers respectively. Generated PoIs are red markers.

6. Conclusions

This paper proposed a scheme for the coordination and path planning of multipl heterogeneous UAS in cooperative missions. It is composed of two modules. The first solves the VRP and assigns, using the OR-Tools library, each UAS with a list of points of interest to be visited minimizing the total mission time. The second module is based on A^* and determines a path for each UAS minimizing the deviation w.r.t. its route avoiding NFZs and fulfilling the UAS kinematic constraints. The validation, performed by building an environment in Python, showed that the implemented VRP solver using OR-Tools is efficient, robust, and has good scalability. For the path planning, the presented method exhibited lower deviation w.r.t. UAS routes than existing methods such as B-Spline and Dubins. The overall system operates robustly both in visiting points and searching area problems. The integration with the autopilots and command and control systems from UAV Navigation-Grupo Oesía SLU and the validation on real UAS in the framework of U-SCUAR project are objects of current development.

Acknowledgements

Partial funding has been obtained from UAV Navigation-Grupo Oesía SLU under contract "U-PLAN: Arquitectura Jerárquica para la Planificación de UAS Heterogéneos"within the framework of project "U-SCUAR: Investigación Avanzada de UAS en el Ámbito de la Categoría Específica", funded by the CDTI through the 2022 Aeronautical Tech. Plan (PTA). This project has been partially funded by Horizon Europe U-ELCOME Ü-space European COMmon dEpLoyment"funded by the EU under contract 101079171. Partial funding has been obtained from project SARA (TED2021-131716B-C22) funded by the Ministry of Science and Innovation.

Referencias

Adewumi, A. O., Adeleke, O. J., 2018. A survey of recent advances in vehicle routing problems. International Journal of System Assurance Engineering and Management 9, 155–172.

Cabreira, T. M., Brisolara, L. B., Paulo R, F. J., 2019. Survey on coverage path planning with unmanned aerial vehicles. Drones 3 (1), 4.

Cai, Y., Xi, Q., Xing, X., Gui, H., Liu, Q., 2019. Path planning for uav tracking target based on improved a-star algorithm. In: 2019 1st International Conference on Industrial Artificial Intelligence (IAI). IEEE, pp. 1–6.

Corberán, T., Plana, I., Sanchis, J. M., Segura, P., 2024. The multidepot drone general routing problem with duration and capacity constraints. International Transactions in Operational Research.

CVX Research, I., Aug. 2012. CVX: Matlab software for disciplined convex programming, version 2.0.

Dhulkefl, E., Durdu, A., Terzioğlu, H., 2020. Dijkstra algorithm using uav path planning. Konya Journal of Engineering Sciences 8, 92–105.

Furnon, V., Perron, L., May 2024. Or-tools routing library.

Gu, R., Poon, M., Luo, Z., Liu, Y., Liu, Z., 2022. A hierarchical solution evaluation method and a hybrid algorithm for the vehicle routing problem with drones and multiple visits. Transportation Research Part C: Emerging Technologies 141, 103733.

Gurobi Optimization, LLC, 2023. Gurobi Optimizer Reference Manual.

He, W., Qi, X., Liu, L., 2021. A novel hybrid particle swarm optimization for multi-uav cooperate path planning. Applied Intelligence 51 (10), 7350–7364.

Huang, L., Qu, H., Ji, P., Liu, X., Fan, Z., 2016. A novel coordinated path planning method using k-degree smoothing for multi-uavs. Applied Soft Computing 48, 182–192.

Kuiper, E., Nadjm-Tehrani, S., 2006. Mobility models for uav group reconnaissance applications. In: 2006 International Conference on Wireless and Mobile Communications (ICWMC'06). IEEE, pp. 33–33.

Li, B., Qi, X., Yu, B., Liu, L., 2019. Trajectory planning for uav based on improved aco algorithm. IEEE Access 8, 2995–3006.

Ravankar, A., Ravankar, A. A., Kobayashi, Y., Hoshino, Y., Peng, C.-C., 2018. Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges. Sensors 18 (9), 3170.

Saleu, R. G. M., Deroussi, L., Feillet, D., Grangeon, N., Quilliot, A., 2022. The parallel drone scheduling problem with multiple drones and vehicles. European Journal of Operational Research 300 (2), 571–589.

Sharma, H., Sebastian, T., Balamuralidhar, P., 2017. An efficient backtracking-based approach to turn-constrained path planning for aerial mobile robots. In: 2017 European Conference on Mobile Robots. pp. 1–8.

Song, B., Wang, Z., Zou, L., 2017. On global smooth path planning for mobile robots using a novel multimodal delayed pso algorithm. Cognitive Computation 9 (1), 5–17.

Zammit, C., Van Kampen, E.-J., 2022. Comparison between a* and rrt algorithms for 3d uav path planning. Unmanned Systems 10 (02), 129–146.

Zhen, Z., Xing, D., Gao, C., 2018. Cooperative search-attack mission planning for multi-uav based on intelligent self-organized algorithm. Aerospace Science and Technology 76, 402–411.