







Article

Multi UAV Coverage Path Planning in Urban Environments

Javier Muñoz ^{*}, Blanca López , Fernando Quevedo , Concepción A. Monje , Santiago Garrido 
and Luis E. Moreno 

Robotics Lab, Universidad Carlos III de Madrid, Av. Madrid 30, 28911 Leganés, Spain; bilopezp@ing.uc3m.es (B.L.); fquevedo@ing.uc3m.es (F.Q.); cmonje@ing.uc3m.es (C.A.M.); sgarrido@ing.uc3m.es (S.G.); moreno@ing.uc3m.es (L.E.M.)

* Correspondence: jamunozm@ing.uc3m.es

Abstract: Coverage path planning (CPP) is a field of study which objective is to find a path that covers every point of a certain area of interest. Recently, the use of Unmanned Aerial Vehicles (UAVs) has become more proficient in various applications such as surveillance, terrain coverage, mapping, natural disaster tracking, transport, and others. The aim of this paper is to design efficient coverage path planning collision-avoidance capable algorithms for single or multi UAV systems in cluttered urban environments. Two algorithms are developed and explored: one of them plans paths to cover a target zone delimited by a given perimeter with predefined coverage height and bandwidth, using a boustrophedon flight pattern, while the other proposed algorithm follows a set of predefined viewpoints, calculating a smooth path that ensures that the UAVs pass over the objectives. Both algorithms have been developed for a scalable number of UAVs, which fly in a triangular deformable leader-follower formation with the leader at its front. In the case of an even number of UAVs, there is no leader at the front of the formation and a virtual leader is used to plan the paths of the followers. The presented algorithms also have collision avoidance capabilities, powered by the Fast Marching Square algorithm. These algorithms are tested in various simulated urban and cluttered environments, and they prove capable of providing safe and smooth paths for the UAV formation in urban environments.

Keywords: UAVs; coverage path planning; path planning; fast marching



Citation: Muñoz, J.; López, B.; Quevedo, F.; Monje, C.A.; Garrido, S.; Moreno, L.E. Multi UAV Coverage Path Planning in Urban Environments. *Sensors* **2021**, *21*, 7365. <https://doi.org/10.3390/s21217365>

Academic Editor: Heoncheol Lee

Received: 28 September 2021

Accepted: 3 November 2021

Published: 5 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Unmanned Aerial Vehicles (UAVs), due to their great versatility, have been recently used for tasks such as smart farming [1], surveillance [2–4], wildfire control [5,6], delivery [7–9], and natural disaster situations [10,11] among others. UAVs are mainly classified into two categories: fixed-wing and rotary wing. Fixed-wing UAVs have rigid wings which allow flying based on the lift created by forward speed, while rotary-wing UAVs use rotary blades to fly. Rotary-wing UAVs are more versatile since they allow vertical take off and landing, great maneuverability, low-altitude flights, and hovering tasks [12].

The Coverage Path Planning (CPP) problem is considered a motion planning subtopic, which objective is to find a path which covers every point of a certain area of interest while avoiding obstacles [13]. This area of interest is commonly decomposed into cells of different sizes and resolutions, based on the computation capabilities available or the UAV's sensors range, which commonly are LiDAR sensors or cameras. The Coverage Path Planning problem is usually divided into two main categories. One category consists of finding the optimal path that covers a predefined target area, while the other one consists of calculating a smooth path that connects every point in the given set of designated waypoints.

For the CPP category, which focuses on covering a target area, the first step is to consider the parameters that define the area. The perimeter of the target area can be represented by a series of vertices, where each vertex is defined by a pair of coordinates and its internal angle. Once the target area is defined, it is necessary to apply some kind of cellular decomposition

technique to ensure the complete coverage [13]. The most common cellular decomposition methods used for the CPP problem are exact and approximate cellular decomposition.

Exact cellular decomposition techniques divide the target area into sub-areas, which are usually covered using back-and-forth motions. Following this division, the CPP problem can be considered as a graph search to create a path that connects all sub-areas. Therefore, the final path is a combination of the back-and-forth motions inside each sub-area and the path that connects adjacent sub-areas [14]. An example of exact cellular decomposition can be seen in Figure 1.

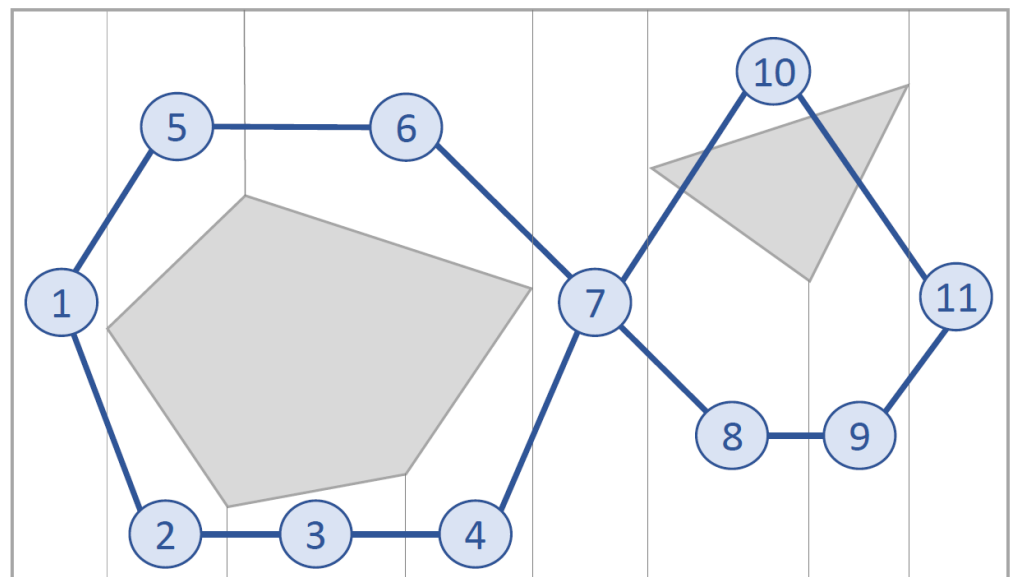


Figure 1. Sub-areas obtained from exact cellular decomposition [12]. Reprinted from ref. [12].

Some of the CPP works that use the exact cellular decomposition technique are: coverage of concave polygonal areas by decomposing them into non-concave sub areas and using back-and-forth motions [15–17], coverage for fixed-wing UAVs analyzing wind to decrease flight time [18,19], optimal coverage for fixed-wing UAVs able to avoid obstacles with arbitrary shapes [20,21], spiral and back-and-forth motions in concave areas [22], a cooperative strategy to simultaneously cover sub-areas of the given target area by a team of UAVs [23], and a spiral CPP algorithm for missions in coastal regions using multiple heterogeneous UAVs [24,25], among others.

Approximate cellular decomposition techniques divide the target area into regular cells [13]. These cells usually have a square form, and are commonly used to generate coverage paths [14]. The size of the cells is commonly determined by the field of view (FOV) of the sensor, usually a LiDAR sensor or a camera, attached to the UAV and the distance from the UAV to the terrain as seen in Figure 2.

A few CPP articles that feature approximate cellular decomposition techniques are: an approach for image mosaicing in precision agriculture with irregular-shaped areas [26], an optimal CPP algorithm where the area is represented by points of interest [27], another approach for precision agriculture that uses a team of heterogeneous quadcopters [28], a meta-heuristic algorithm named Harmony Search (HS) [29], online non-uniform CPP using different grid sizes [30,31], high-resolution aerial sensing with multiple heterogeneous UAVs for non-convex areas [32], pheromone-based methods [33], and a combination of digital pheromones and evolutionary strategies for coordination of multiple UAVs [34], among others.

Other CPP approaches use waypoints to define the targets to cover: [35] uses waypoints for 3D CPP for autonomous structural inspection operations using aerial robots and [27] develops an optimal CPP algorithm where the area is represented by points of interest.

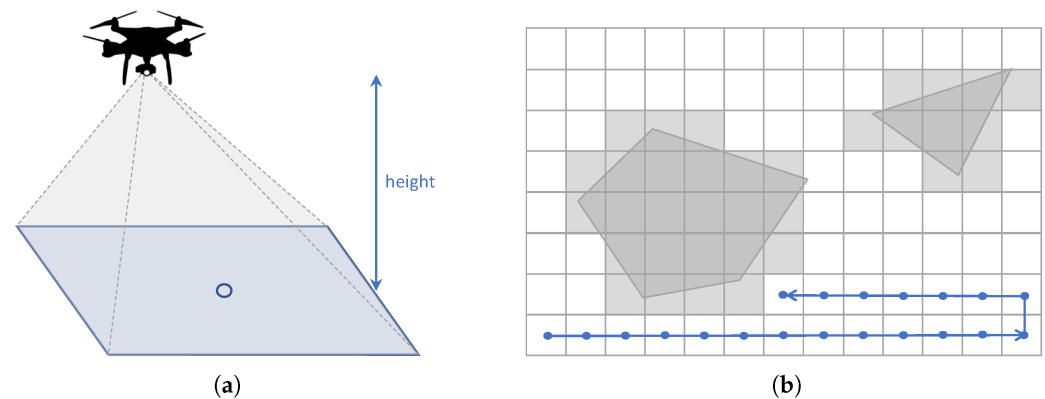


Figure 2. (a) Projected area of the UAV; (b) Cells obtained by applying approximate cellular decomposition [12]. Reprinted from ref. [12].

Most approaches focus on calculating coverage paths and optimizing the computation time and the time needed to cover the area, but they usually only consider empty surfaces with no obstacles. However, there exist some methods that take into account coverage areas with obstacles [26,36,37]. The main disadvantage of these works compared to our method, is that they usually decompose the area to cover into big cells, and completely avoid the cells that contain any obstacles.

More recent publications present different approaches for planning CPP missions. Muñoz et al. [38] propose a Differential Evolution-based CPP method for search and rescue tasks in marine environments. Chen et al. [39] propose a clustering-based CPP method, which consists in dividing the target area into clusters and assign each area to a different UAV based on the area properties and the UAV's scanning width and flying speed. However, this work can be seen as a task assignment method, and could be used as a previous step to our proposed target area CPP method. Cho et al. [40] highlight a two-phase method for solving the CPP problem of multiple-UAV areas in maritime search and rescue. The first phase decomposes the search area into a graph, and the second phase uses the mixed-integer linear programming (MILP) to calculate an optimal path that minimizes the completion time. Melo et al. [41] introduce a 3D dynamic algorithm for CPP capable of covering a set of 3D waypoints while allowing optimization of energy usage.

This work proposes two methods for multi UAV optimal CPP in urban environments that use the approximate cellular decomposition approach and have collision-avoidance capabilities. One method generates a coverage path given a target area using back-and-forth motions, while the other method creates a smooth path that covers a set of given waypoints. Furthermore, both methods are applied to teams of UAVs by developing an algorithm that follows a leader-follower approach [42], creating UAV formations. For the collision-avoidance capabilities and the UAV formation algorithm, the FM² method is used. Since the flying of drones over cities is forbidden in Spain at the moment, the algorithms are tested in simulated urban environments.

The main advantage of the proposed algorithms over previous approaches, is that the UAVs fly as a formation instead of sub-dividing the target area. Furthermore, this formation can be deformed accordingly so it avoids any obstacles that may be in the way. The state of the art methods are really efficient when performing clustering and CPP, but cannot be applied to urban or cluttered environments that may contain obstacles such as communication towers, cranes, small buildings or skyscrapers.

The rest of this paper is organized as follows. Section 2 contains all the main aspects of the proposed solution to the coverage path planning problem for search operations. Section 3 analyses the overall performance of the strategy and evaluates its applicability to real world case scenarios. Section 4 provides a discussion about the main advantages of our proposal, and compares it with similar works in the literature. Lastly, Section 5 highlights the main conclusions and contributions drawn throughout the progress of this work.

2. Description of the Approach

The development of the algorithm has been divided into three major stages, which will be detailed hereinafter. Firstly, the method was used to achieve the leader-follower drone formation in order to avoid obstacles while keeping a sufficient safety distance among vehicles. Secondly, the process was followed to obtain the coverage path for the leader drone. Finally, the approach through which the follower UAVs obtain their path was based on the already computed collision-free leader path and the aforementioned leader-follower formation strategy. The proposed algorithm aims to be applied in both search target areas and in linear coverage missions, in which a series of waypoints must be visited by the vehicles. The second stage mentioned above varies slightly for these two scenarios and thus, both applications are explained separately in this section. It must be mentioned that in this project UAVs are considered as a point-size object flying through free airspace.

In this work, we only consider the planning for the coverage path exclusively, as we do not take into account the planification needed from the UAV depot to the start of the CPP mission, and back to the depot after the CPP mission has finished. The algorithm used to plan these paths and manage traffic in cluttered urban environments is shown in our previous publication [43].

2.1. UAV Formation Approach Using Fast Marching Square

Now that the basis for the CPP algorithms has been detailed, it is necessary to define the formation strategy that will be followed by the drones. In this case, a leader-follower approach is implemented, using a virtual leader in the case of an even number of drones. The drones follow a triangular formation, which can be deformed to avoid obstacles using the Fast Marching Square (FM²) method [44]. This path planning method is a variation of the original Fast Marching method [45]. It creates a velocity map given a binary map of occupation by propagating a wavefront taking all the obstacles as source points. To better illustrate the differences between the two methods, paths are calculated and shown in Figures 3 and 4. Both figures show that the path obtained with the FM² method is more smooth and reproducible by a robot, thanks to the calculation of the velocity map.

The velocity map values range from 0 to 1, representing the maximum speed allowed for the drone. Obstacles will imply speeds equal to zero, whilst points in space far enough away from obstacles will allow maximum speeds. When computing a path for the vehicles to follow, FM² method will get the shortest path from the initial position to the goal position that lets the vehicles navigate at higher speeds. Using this velocity map, the partial goals of the followers of the formation can be adjusted so obstacles are avoided safely. An example of this formation approach is shown in Figure 5.

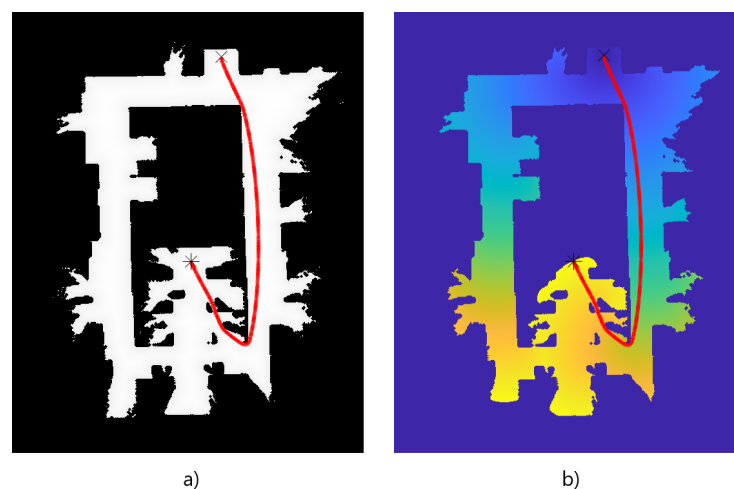


Figure 3. (a) initial binary map, (b) time of arrival of the propagating wavefront. The path obtained with the Fast Marching Method (FMM) is shown as a red line.

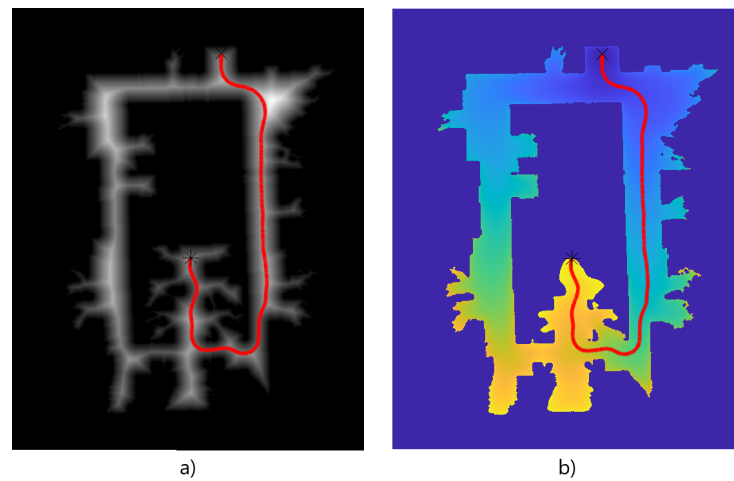


Figure 4. (a) velocity map, (b) time of arrival of the wavefront. The path obtained with Fast Marching Square (FM²) is shown as a red line.

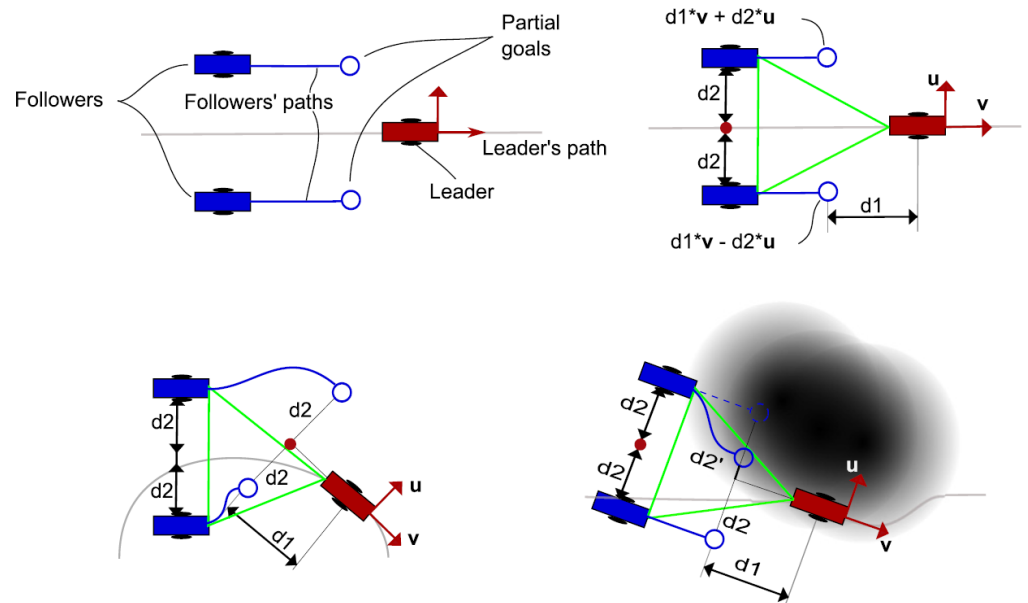


Figure 5. UAVs formation algorithm. From top left to bottom right: main components of the formation approach; reference definition of the triangle-shaped UAV formation; modification of the partial goals of the followers according to the leader position; modification of partial goals according to the obstacles of the environment.

For a given number of drones n , the desired positions in the formation are calculated as follows. First, a multiplier is applied to calculate the distance in the perpendicular axis to the leader's path. If n is even, a virtual leader is used, whereas if n is odd, one of the drones is used as the leader. i is an auxiliary variable used to calculate the multiplier variable for each UAV.

$$\begin{aligned}
 \text{If } n \text{ is odd} \quad & \text{multiplier}(2i : 2i + 1) = i && \text{for } i = 1 : \left(\frac{n}{2} - 0.5\right) \\
 \text{If } n \text{ is even} \quad & \text{multiplier}(2i : 2i + 1) = i - 0.5 && \text{for } i = 1 : \frac{n}{2}
 \end{aligned} \tag{1}$$

Now that the multiplier value for each drone is known, the next partial goals for the followers are calculated as follows:

$$partialGoal(n) = leaderGoal - multiplier(n)d_1\mathbf{v} \pm B(n)multiplier(n)d_2\mathbf{u} \quad (2)$$

where $partialGoal(n)$ is the next partial goal for drone n , $leaderGoal$ is the next point in the leader's path, $B(n)$ is the velocity value at the position of drone n , d_1 and d_2 are the respective parallel and perpendicular distances to the leader's path, and \mathbf{v} and \mathbf{u} are the vectors that define said parallel and perpendicular directions. Then, the UAVs are added as obstacles for the other UAVs in the team, and their paths from their current positions to their new partial goals are calculated using FM^2 . The steps of the algorithm are detailed in Algorithm 1 and in Figure 6.

Algorithm 1 Routine of the FM^2 formation strategy

- 1: Define number of UAVs of the formation
 - 2: Calculate the multiplier values based on the number of UAVs following Equation (1)
 - 3: $W_0 \leftarrow$ Occupation matrix of the environment
 - 4: $W \leftarrow$ Velocity map computed from W_0
 - 5: Calculate the leader's path using FM^2
 - 6: Calculate first partial objectives for all follower UAVs following Equation (2)
 - 7: **while** $step < length(leaderPath)$ **do**
 - 8: Calculate the follower paths to their partial objectives using FM^2
 - 9: Move all UAVs to their next partial objective
 - 10: $leaderGoal \leftarrow leaderPath(step)$
 - 11: Calculate the next partial objectives for all follower UAVs following Equation (2)
 - 12: $step \leftarrow step + stepSize$
 - 13: **end while**
-

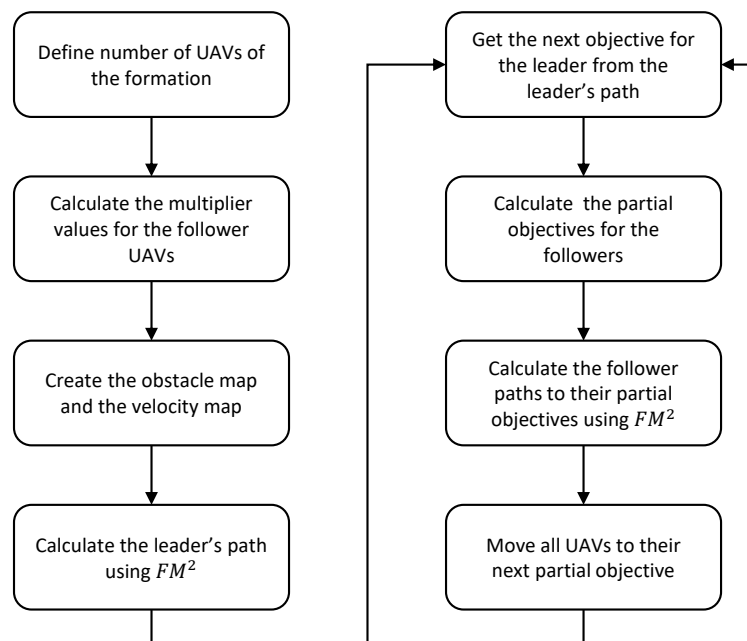


Figure 6. Flowchart for the FM^2 formation strategy.

This strategy is applied to both methods proposed in the following subsections.

2.2. Multi UAV Coverage Path Planning Given a Target Area

Given the number of desired UAVs for the formation and a target area defined by a set of perimeter points, the proposed algorithm aims to cover the whole area while maintaining the desired flight level relative to the uneven ground level and avoiding obstacles.

The first step is to create a binary mask representing the target area to cover based on the given perimeter points. An example of coverage binary mask is shown in Figure 7. From now on, all figures will be trimmed down so the operations performed and the paths obtained can be seen more clearly.

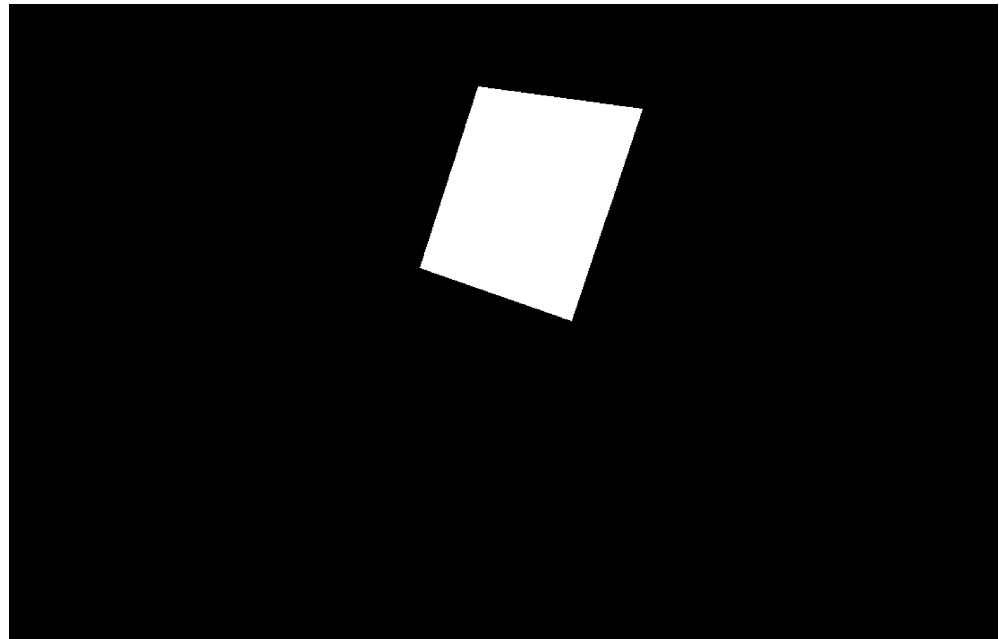


Figure 7. Binary mask that defines the target area to cover, defined by a given set of perimeter points.

Then, the coverage angle is obtained by detecting the longest edge of the perimeter. This will guarantee that the coverage path has the minimum number of turns necessary. The longest edge detected for the binary mask created in the first step can be seen in Figure 8a. The path is then computed by using back-and-forth motions, with a predefined coverage bandwidth. The obtained path is shown in Figure 8b.

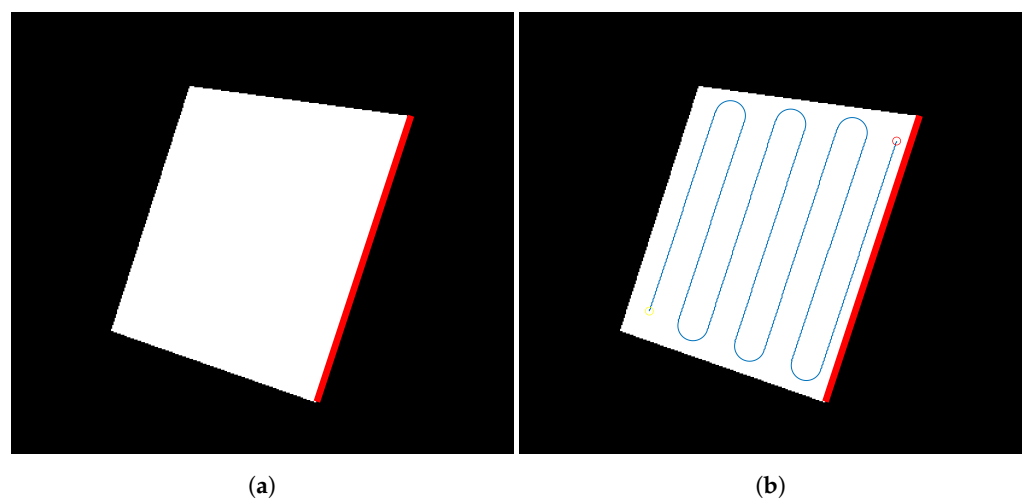


Figure 8. Initial 2D coverage path calculation process. (a) Selected best edge so the coverage path has the minimum number of turns. (b) Computed path using back-and-forth motions.

Once the initial path is obtained, it is time to consider the obstacles in the environment. For that purpose, a series of processes are followed. First, a 3D map of the environment is created by using the coordinates and LiDAR data from the Spanish Instituto Geográfico Nacional (IGN) [46]. An example map generated from this data is shown in Figure 9a. The data corresponds to the area around the Valencia harbor. The data obtained has a resolution of 5 m, so every cell in the 3D and 2D maps is equivalent to a 25 m² area.

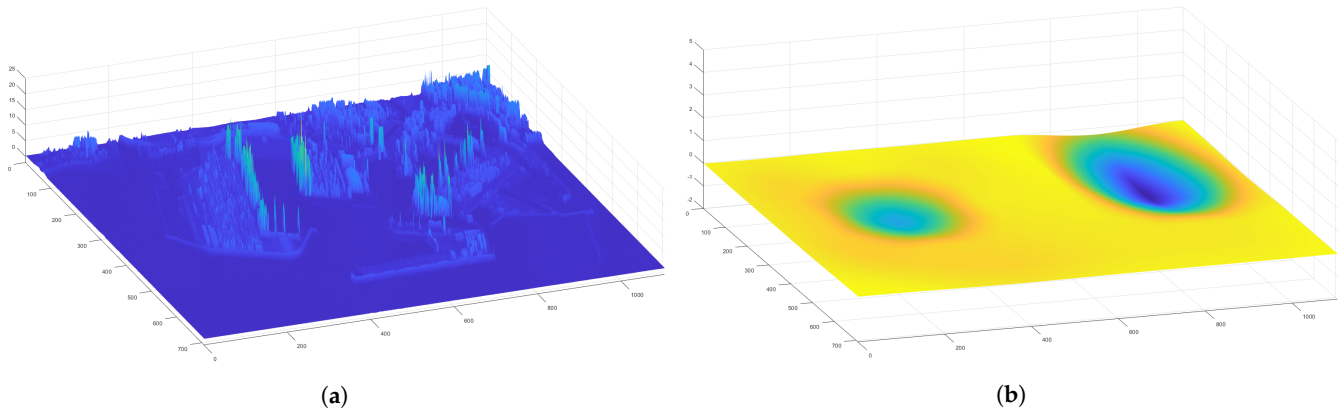


Figure 9. Valencia harbor environment used for the target area CPP mission. (a) 3D data of the Valencia harbor obtained from [46]. (b) Floor level data of the Valencia harbor.

To compute the flight level, it is necessary to calculate a map with the floor level data. For this purpose, a grid method is used. The point with minimum height of the grid is taken as a reference, and all points of the selected grid are set to that reference height. Once the grid is moved over the whole map, a median filter and an average filter are used to smooth the obtained surface. The floor level map obtained from the Valencia harbor data is shown in Figure 9b. Note that, since the raw data is taken directly from [46], and the minimum height value in the grid is taken as the floor level value, this can lead to small errors such as negative values in the floor level map. This could be fixed by doing a more thorough post-processing of the data, but the tests have determined that this does not really affect the trajectories of the UAVs at all.

After calculating the floor level, we can use this data and the desired relative flight level to obtain the absolute height interval of the drones performing the coverage mission. Using this interval, we can detect the obstacles inside the interval flight range of interest and discard the ones out of range. In addition, they are added to a binary mask that is used for the obstacle avoidance phase. The obstacle map is shown in Figure 10a.

When the obstacle map has been defined, the path for the leader drone is calculated with the Fast Marching Square method [47] and a binary mask created from the dilated original path and the obstacle map. The obstacles in the obstacle map are dilated to ensure a smooth collision-free path for the leader, which consequently allows for the followers to avoid obstacles using the leader-follower approach explained in the subsection above.

The obtained collision-free leader path is shown in Figure 10b.

Next, the paths for all followers are calculated based on the collision-free leader path and the leader-follower formation strategy. One interesting feature of the velocity map is that it can be saturated to a certain value, so all vehicles traveling at a distance greater than the one specified by the saturation can travel at full speed. Figure 11 shows the velocity map obtained from the obstacle map, saturated to a value of 15 cells, equivalent to 75 m. This value is chosen so the UAVs are aware of the position of obstacles sooner, and their movement as a formation is more smooth. This also helps to set the maximum distance allowed to obstacles, which is in a range of 5–6 cells.

After applying the leader-follower FM² based formation approach to the leader path for teams of two and three UAVs, the 2D formation paths are obtained. Figure 12a,b shows the 2D paths for the leader and the followers for teams of two and three UAVs.

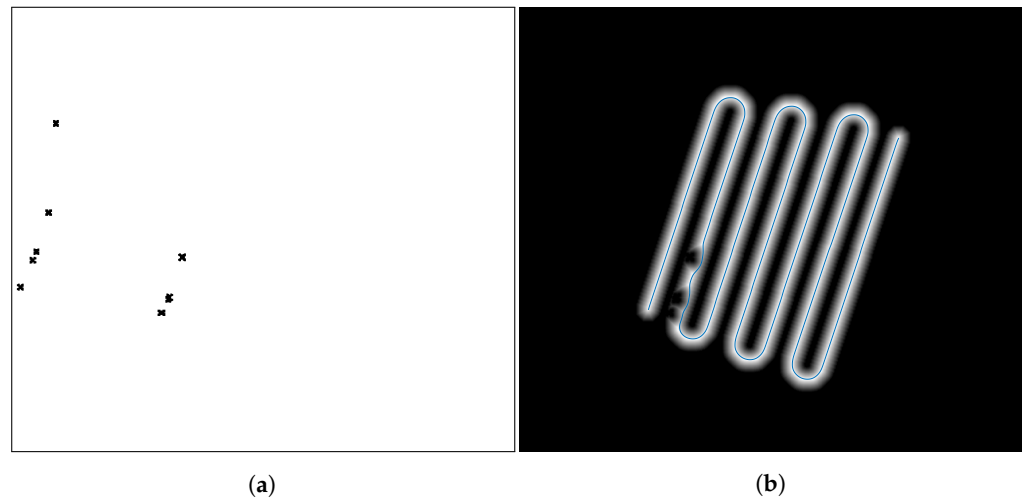


Figure 10. Calculation of the obstacle-free leader path. **(a)** Obstacle map calculated from the 3D data and the height of the drone paths. **(b)** Collision-free path calculated for the leader of the formation using the Fast Marching Square method.

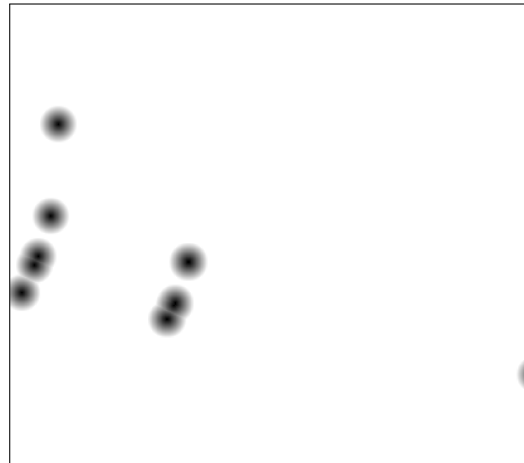


Figure 11. Saturated velocity map calculated from the obstacle map for the formation planning.

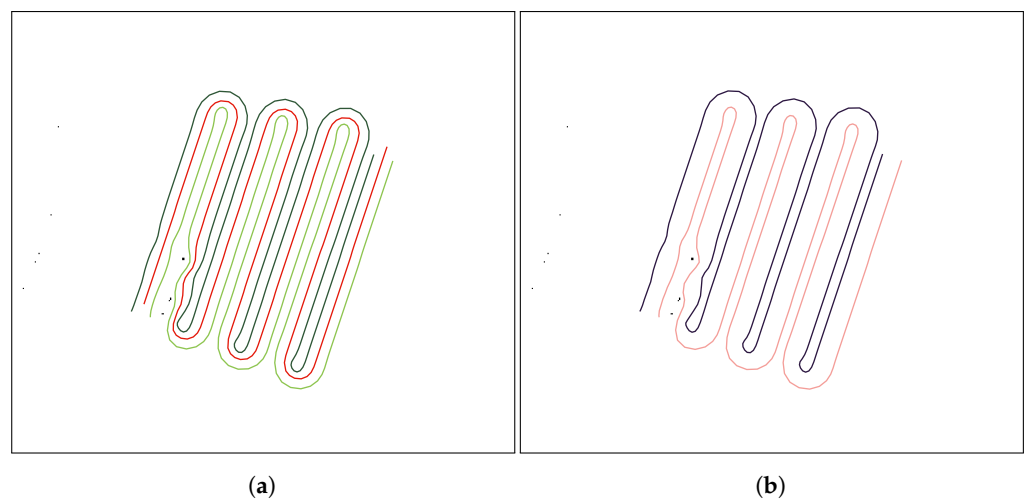


Figure 12. Calculation of the collision-free follower paths for teams of three and two UAVs, respectively. **(a)** Example of collision-free 2D paths using a team of three UAVs. **(b)** Example of collision-free 2D paths using a team of two UAVs.

Finally, the floor level data and the desired flight level are used to add height to the paths and make them 3D. Since the floor level has been smoothed with a median

and average filter, the resulting 3D paths are smooth. The obtained 3D paths are shown in Figure 13b. A flowchart showing the steps of the proposed algorithm can be seen in Figure 14.

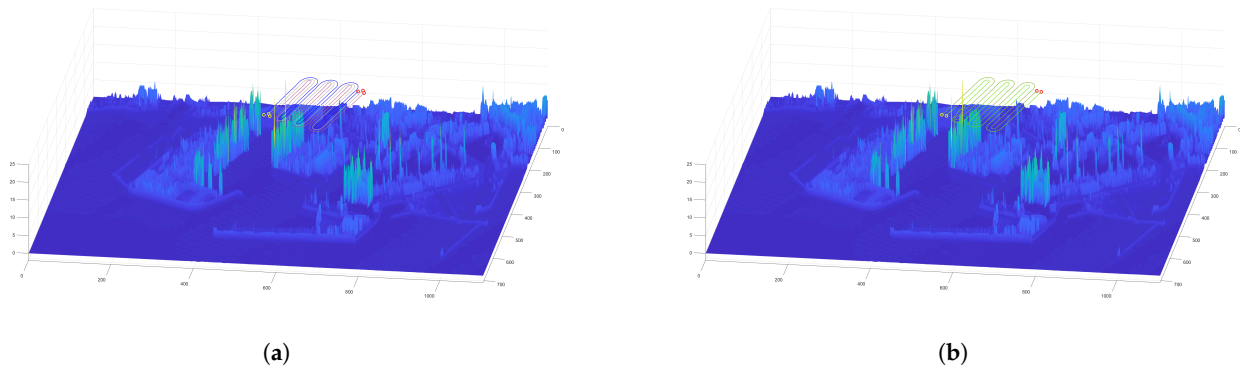


Figure 13. Calculation of the 3D collision-free follower paths for teams of three and two UAVs, respectively. (a) Example of a collision-free CPP mission given a target area using a team of three UAVs. (b) Example of a collision-free CPP mission given a target area using a team of two UAVs.

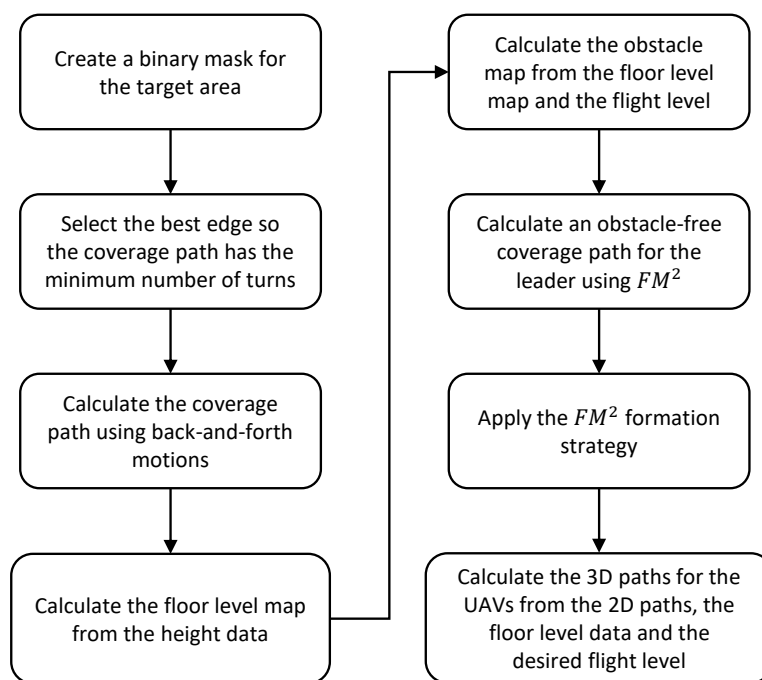


Figure 14. Flowchart for the CPP given a target area algorithm.

In the next subsection, the same procedure is going to be applied to a different coverage path planning problem given a set of waypoints to cover.

2.3. Multi UAV Coverage Path Planning Given a Set of Waypoints

Given a number of desired UAVs for the mission and a set of waypoints they have to pass through, we propose a method to perform the coverage of all defined waypoints while avoiding obstacles.

First, the waypoints must be defined. An example of waypoints that cover a street near the Bernabeu Stadium in Madrid is shown in Figure 15a.

Once the waypoints have been selected, the procedure is the same as the followed in the previous method. An obstacle map is obtained from the floor level data (Figure 15)

and the selected flight level, and an obstacle-free path is calculated for the leader of the formation. The path is calculated using the FM² method, by connecting all the waypoints in order with straight lines, and then dilating them to make room for the leader UAV to maneuver and avoid obstacles. When the velocity map is created from this binary map, the generated path is smooth by nature, so the leader UAV does not necessarily pass over the exact position of the waypoint. This makes the trajectory easier to follow by a real UAV, since it resembles the operation of a pure-pursuit controller [48]. An example of an obstacle map with the corresponding path can be seen in Figure 16.

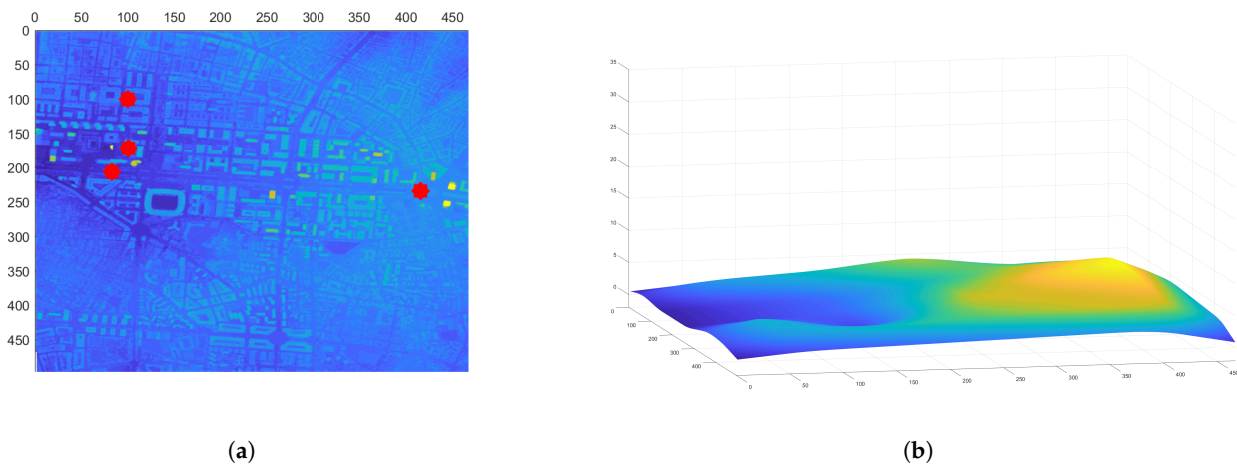


Figure 15. Bernabeu stadium environment used for the target area CPP mission. (a) Selected waypoints for the coverage mission near the Bernabeu Stadium in Madrid. (b) Floor level data of the Bernabeu Stadium area in Madrid.

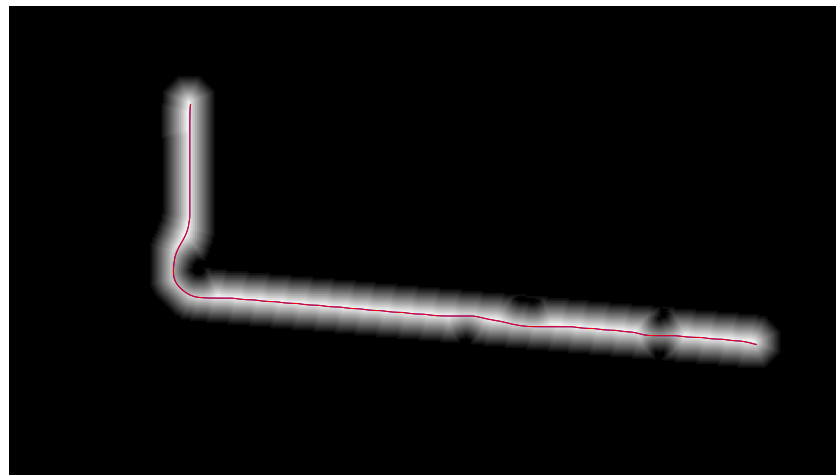


Figure 16. Collision-free path calculated for the leader of the formation using the Fast Marching Square method.

Then, the leader-follower formation approach is applied to the generated leader's path to create the collision free paths for the followers. The collision-free 2D paths using teams of three and two UAVs can be seen in Figure 17a,b, respectively.

Note that since the algorithm is intended to be used in urban environments, a feature has been added to generate a path that is set at a certain distance to either the left or the right of the selected waypoints, so it can be used to cover roads or busy streets while minimizing the possible risks. Figure 18a,b show the paths for the UAV formation when the distance is set to the left and right directions, respectively.

Finally, the floor level data and the desired flight level are used to add height to the paths and make them 3D. This greatly reduces the computation time, since the paths of the formation would need to be calculated in 3 dimensions, adding complexity to the

algorithm. The obtained 3D paths are shown in Figure 19a,b. A flowchart showing the steps of the proposed algorithm can be seen in Figure 20.

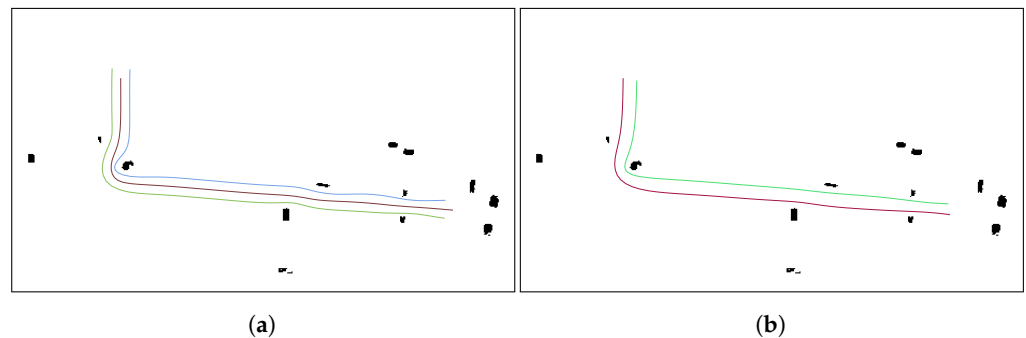


Figure 17. Collision-free 2D paths using teams of three and two UAVs, respectively. (a) Example of collision-free 2D paths using a team of three UAVs given a set of waypoints. (b) Example of collision-free 2D paths using a team of two UAVs given a set of waypoints.

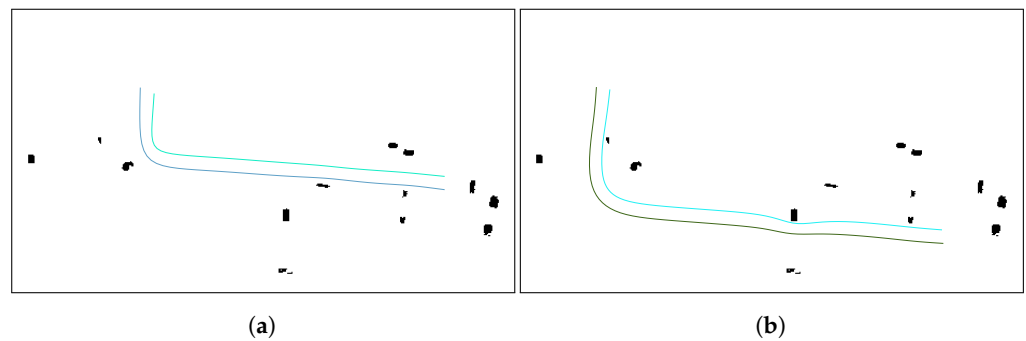


Figure 18. Collision-free 2D paths using teams of three and two UAVs, respectively. (a) Generated collision-free 2D paths using a team of two UAVs setting a certain distance to the left side of the original leader's path. (b) Generated collision-free 2D paths using a team of two UAVs setting a certain distance to the right side of the original leader's path.

In the next section, measures from these missions are going to be collected and analyzed to study the behavior of the algorithms, including the total execution time, the velocity map value of the follower positions in each iteration, and their distances to the leader.

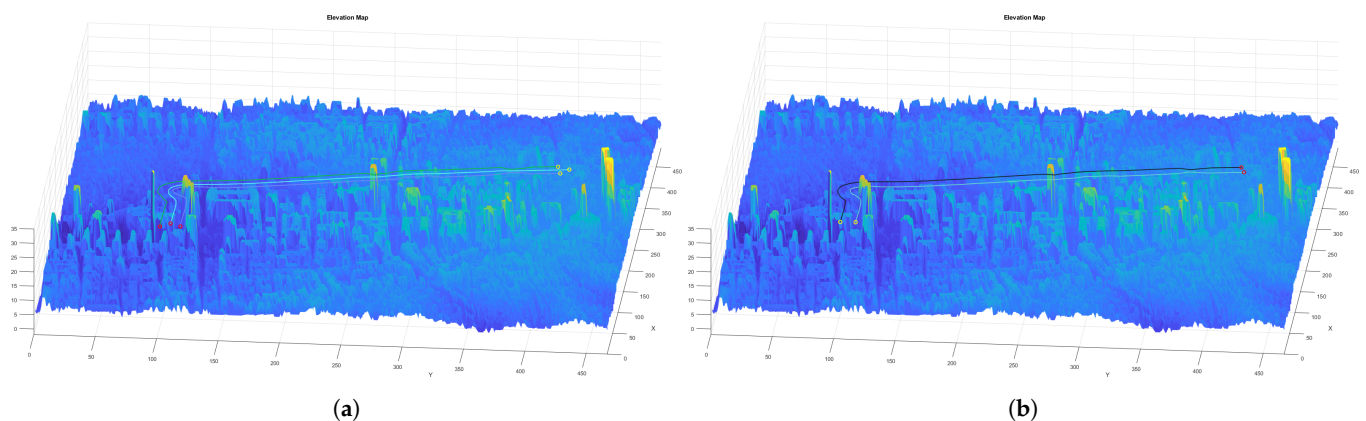


Figure 19. Collision-free 3D paths using teams of three and two UAVs, respectively. (a) Example of a collision-free CPP mission given a set of waypoints using a team of three UAVs. (b) Example of a collision-free CPP mission given a set of waypoints using a team of two UAVs.

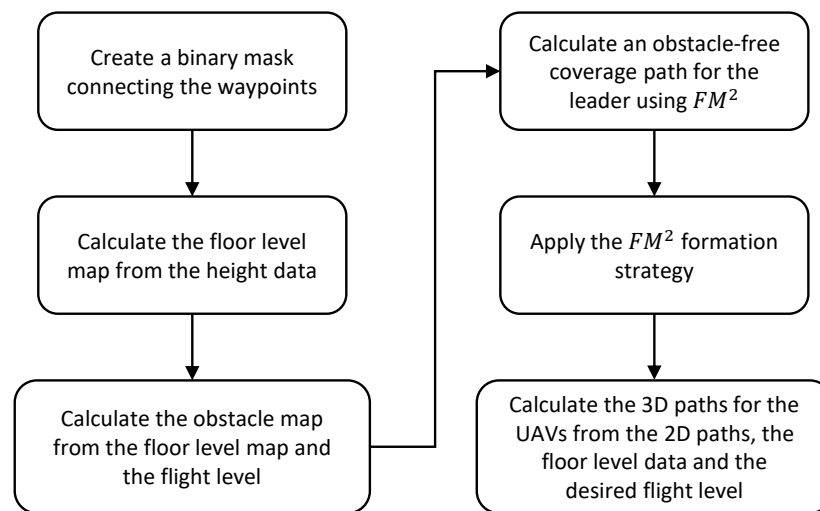


Figure 20. Flowchart for the CPP given a set of waypoints algorithm.

3. Results

Once the CPP and collision avoidance formation algorithms have been implemented, some measures are taken to study the performance of the algorithms and their applicability to real world use cases. The measures taken from the experiments are the total computation time and the distance from each follower to the leader of the formation.

The machine used to run the test is a computer with 16 GB RAM and an AMD Ryzen 5 5600X CPU@4.12 GHz.

We observed that, in general, the computation times of the algorithm were high and it was not optimal to calculate every step of the formation path with the FM^2 method if there are no obstacles around, since it only adds complexity to the computations when it is not necessary. For that, we create an hybrid approach, which only activates the FM^2 formation algorithm when the team of UAVs is closer than a certain distance from the obstacles around it, making the calculations when there are no obstacles around much faster, since we can set the value of B (see Equation (1)) to 1 and we do not need to add all the UAVs to the velocity map and compute their paths with FM^2 .

In the case of the CPP mission given a target area, the obtained distances to the leader and velocity map value data for a three drone mission is shown in Figure 21a.

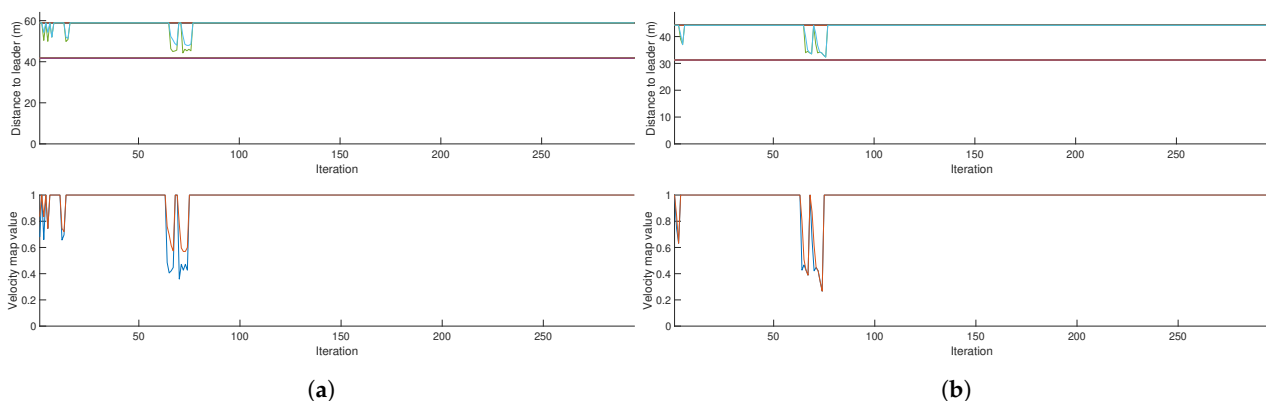


Figure 21. Velocity map values and distances from followers to the leader for teams of three and two UAVs performing a coverage path planning mission given a target area. (a) Velocity map values and distances from followers to the leader for a team of three UAVs. (b) Velocity map values and distances from followers to the leader for a team of two UAVs.

The first graph shows the distance from the followers to the leader in meters. The orange line represents the reference and therefore maximum distance from the followers to the drone and the purple line represents the minimum distance. The second graph

shows the value of the velocity map for the followers. In the case of the hybrid approach, this value has been set to 1 when obstacles have not been detected. As it can be seen in Table 1, the computation time is greatly decreased in the case of the target CPP area and slightly decreased in the case of the waypoints CPP. This is due to the fact that the target area CPP is conformed by more complex paths that do back-and-forth motions, so the step size to calculate the follower paths is smaller, allowing for a higher accuracy. That means that the number of iterations the algorithm ran is higher than in the case of the waypoints CPP algorithm, leading to higher computation times. Figures 21a and 22a show that the target area CPP mission needs around 300 iterations, while the waypoints CPP mission uses around 60 iterations to be completed. For the case of an even number of UAVs, the distances to the leader are calculated to the virtual leader.

Table 1. Total computation times in seconds.

	Target Area CPP		Waypoints CPP	
	2 UAVs	3 UAVs	2 UAVs	3 UAVs
Pure FM ²	78.94	106.26	8.13	7.40
Hybrid approach	12.30	11.91	5.90	5.88

As the velocity map value is also a measure of the distance of the UAVs from obstacles, we can determine the minimum distance from any UAV to an obstacle based on this velocity map value and the saturation of the velocity map, which in this case is 15 cells. Since each cell has a resolution of 5 m, we can see that for the case of 3 UAVs (Figure 21a), the minimum velocity value of one follower is 0.4, which multiplied by the saturation and the resolution equals 30 m, while the other follower keeps a minimum distance of 45 m. For the case of 2 UAVs (Figure 21b) the minimum distance for both followers is approximately 22 m.

This graph also indicates that the follower UAVs do not collide with each other, as that would only happen if the velocity map value were 0, which cannot happen since the leader's path is collision free. Based on the velocity values of both follower UAVs, the minimum distance between them is approximately 42 m in the case of the 3 UAV formation and approximately 37 m in the case of the 2 UAV formation, for the chosen coverage bandwidth of 25 cells.

Figure 21b shows the same metrics as the graph above for a team of two UAVs. Due to the multiplier value (see Equation (1)), the maximum and minimum distances from the followers are different for the two team sizes. See that Equation (1) also relates the distances from the followers to the leader to the velocity map value of the follower positions. Since they are directly correlated, the graphs show that when B decreases, the distances from the followers to the leader also decrease. In both cases, there is a disturbance at the beginning of the coverage mission, when the team of UAVs passes on the left side of the obstacles, and a second and greater disturbance when the team of UAVs passes on the right side of the obstacles at a closer distance.

In the case of the CPP mission given a set of waypoints, the obtained distances to leader and velocity map value data teams of two and three drones are shown in Figure 22.

The results obtained are similar to the case of the coverage mission given a target area. In this case, the disturbances in the formation are caused by the sets of buildings on both sides of the planned leader path. Note that, even though the leader path is the same in both cases, the formation of the three UAV team is wider than the one conformed by two UAVs. This difference can be perceived in Figure 22a,b, where the velocity map values for the team of three UAVs are lower than in the case of the team of two UAVs when passing near the same obstacles.

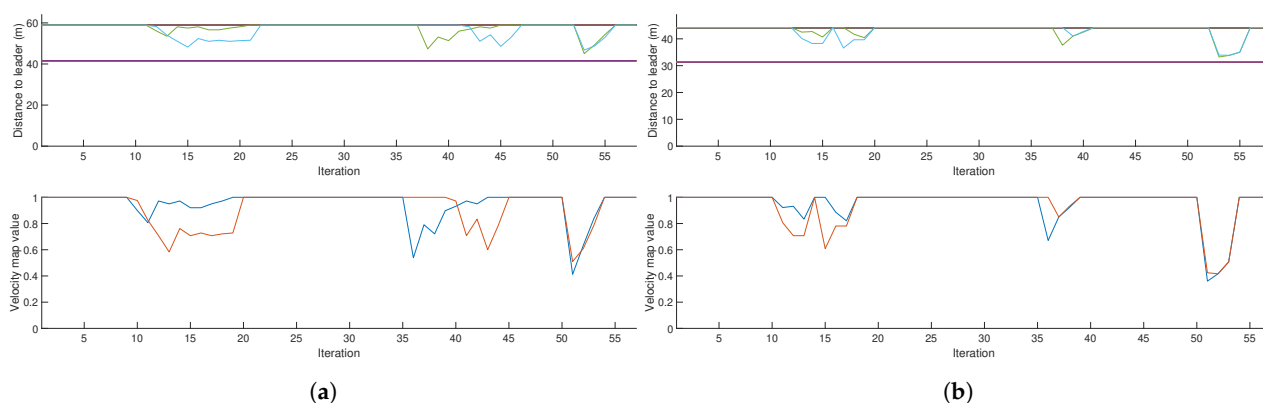


Figure 22. Velocity map values and distances from followers to the leader for teams of three and two UAVs performing a coverage path planning mission, given a set of waypoints. (a) Velocity map values and distances from followers to the leader for a team of three UAVs. (b) Velocity map values and distances from followers to the leader for a team of two UAVs.

For the waypoint coverage mission, we can see that for the case of 3 UAVs (Figure 22a), the minimum velocity value of one follower is 0.4, which multiplied by the saturation and the resolution equals 30 m, while the other follower keeps a minimum distance of 45 m. For the case of 2 UAVs (Figure 22b) the minimum distance for both followers is approximately 37 m.

Based on the velocity values of both follower UAVs, the minimum distance between them is approximately 37 m in the case of the 3 UAV formation and approximately 35 m in the case of the 2 UAV formation, for the chosen coverage bandwidth of 25 cells.

4. Discussion

When solving the coverage path planning (CPP) problem, two main issues should be addressed. One of them is how to generate a coverage path that effectively covers the designated area or waypoints, and the other one is making sure that the UAV or formation of UAVs do not collide with obstacles or enter no-fly zones (NFZ). Most approaches to CPP study empty areas and focus on efficiently covering the target area and therefore use a lower resolution to reduce the computation times and make the planning easier. Some approaches do include obstacles and NFZ in their calculations, but they still use a low resolution with low cells and, in the event of having obstacles in the target area, the whole cell is marked as a NFZ and is therefore not covered. However, some methods try to reduce the effect of obstacles in their path planning by not completely avoiding zones with obstacles, but sticking to the cell's borders.

Our proposed methods solve this issue by using the FM² method to plan the trajectories. First, the obstacle-free path for the leader is calculated, and then the formation planning algorithm is used to plan the trajectories for all follower UAVs in the formation. The use of the FM² method is essential for computing the leader's path, and it is also used to make the UAV formation deformable. That means that the UAVs get closer to the leader's path as they come closer to obstacles. The properties of the velocity map obtained with FM² such as the saturation let us define the safe distance from the UAVs to the obstacles and their maximum allowable speeds.

Since the complexity of the FM² algorithm can be reduced to $O(n)$ [49], the computation times for this algorithm are really low. This allows for a fast and easy implementation of a collision-avoidance path planning algorithm with no local minima, which can be applied to any cell resolution, making it extremely useful in cluttered urban environments.

However, there are some limitations to our work. We assume that the areas where our algorithms are implemented have obstacles in determined positions but, as the first path for the leader is calculated using either back-and-forth motions or connecting the target waypoints, and then creating a binary image and dilating it to be used as a binary mask for the FM². In the event of a big obstacle completely blocking the dilated path in this binary mask, the algorithm would fail to produce a valid path for the formation of UAVs.

This issue can be solved by setting the velocity map cells that are not obstacles to a value slightly higher than 0, so the UAVs can move through this space out of the designated path to continue their mission, but that does not guarantee that the UAVs will continue their mission right after evading the obstacle where they left off.

Finally, practical experiments were not conducted as it is illegal to fly drones in cities without the proper licenses. Although the UAVs were tested in the simulation environments, validations in real use cases should be verified. This is left as future work when the Labyrinth project reaches its final stages.

5. Conclusions

The goal of this work was to design, implement, and analyze a coverage path planning formation approach with a leader-follower formation approach and FM² based collision avoidance for multi-UAV teams in urban environments. The designed algorithm was implemented and its performance was analyzed through simulations.

Two coverage path planning missions were considered: one where the input is given as a target area and another one where the objective was to cover a set of waypoints. For both of them, the procedure is divided into different steps: firstly, the floor level and obstacle maps are calculated from the raw elevation data and then the collision-free leader path is calculated using FM². For the target area case, a back-and-forth path is generated by drawing parallel bands to the longest edge of the target area. Secondly, the leader-follower FM² formation approach is used to calculate the 2D paths for the follower UAVs. Lastly, the 2D paths get the height from the desired flight level and the obtained floor level data, generating the final 3D paths used to perform the coverage mission.

From the obtained results, we can infer that the computation times are reasonable for a mission this complex, even faster when using the hybrid approach. The distances from the followers to the leader and to obstacles have been observed to be safe thanks to the saturation levels of the velocity map and the deformable triangular formation of the team of UAVs. This approach has proven useful in cluttered urban scenarios, in line with the Labyrinth project, which financed this research and aims to integrate the safe use of UAVs in cities.

Author Contributions: Conceptualization, J.M.; methodology design, S.G. and L.E.M.; software, J.M.; validation, B.L.; formal analysis, F.Q.; and project administration, C.A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the EUROPEAN COMMISSION: Innovation and Networks Executive Agency (INEA), through the European H2020 LABYRINTH project. Grant agreement H2020-MG-2019-TwoStages-861696.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: LiDAR data for the environments used in this paper have been downloaded from IGN [46] <http://www.ign.es/web/ign/portal> (accessed on 27 April 2020).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Islam, N.; Rashid, M.M.; Pasandideh, F.; Ray, B.; Moore, S.; Kadel, R. A Review of Applications and Communication Technologies for Internet of Things (IoT) and Unmanned Aerial Vehicle (UAV) Based Sustainable Smart Farming. *Sustainability* **2021**, *13*, 1821.
2. Alladi, T.; Chamola, V.; Kumar, N. PARTH: A two-stage lightweight mutual authentication protocol for UAV surveillance networks. *Comput. Commun.* **2020**, *160*, 81–90. [CrossRef]
3. Scherer, J.; Rinner, B. Multi-UAV surveillance with minimum idleness and latency constraints. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4812–4819. [CrossRef]
4. Liu, Y.; Liu, H.; Tian, Y.; Sun, C. Reinforcement learning based two-level control framework of UAV swarm for cooperative persistent surveillance in an unknown urban area. *Aerosp. Sci. Technol.* **2020**, *98*, 105671. [CrossRef]
5. Zhao, Y.; Ma, J.; Li, X.; Zhang, J. Saliency detection and deep learning-based wildfire identification in UAV imagery. *Sensors* **2018**, *18*, 712.

6. Wu, H.; Li, H.; Shamsoshoara, A.; Razi, A.; Afghah, F. Transfer learning for wildfire identification in uav imagery. In Proceedings of the 2020 54th Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 18–20 March 2020; pp. 1–6.
7. Song, B.D.; Park, K.; Kim, J. Persistent UAV delivery logistics: MILP formulation and efficient heuristic. *Comput. Ind. Eng.* **2018**, *120*, 418–428. [[CrossRef](#)]
8. Al-Hilo, A.; Samir, M.; Assi, C.; Sharafeddine, S.; Ebrahimi, D. UAV-assisted content delivery in intelligent transportation systems-joint trajectory planning and cache management. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 5155–5167. [[CrossRef](#)]
9. Gupta, R.; Shukla, A.; Mehta, P.; Bhattacharya, P.; Tanwar, S.; Tyagi, S.; Kumar, N. Vahak: A blockchain-based outdoor delivery scheme using uav for healthcare 4.0 services. In Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs), Virtual conference, 6–9 July 2020; pp. 255–260.
10. Erdelj, M.; Natalizio, E.; Chowdhury, K.R.; Akyildiz, I.F. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervasive Comput.* **2017**, *16*, 24–32.
11. Chowdhury, T.; Rahnemoonfar, M.; Murphy, R.; Fernandes, O. Comprehensive semantic segmentation on high resolution uav imagery for natural disaster damage assessment. In Proceedings of the 2020 IEEE International Conference on Big Data, Virtual conference, 10–13 December 2020; pp. 3904–3913.
12. Cabreira, T.M.; Brisolara, L.B.; Ferreira Paulo, R. Survey on coverage path planning with unmanned aerial vehicles. *Drones* **2019**, *3*, 1–38. [[CrossRef](#)]
13. Choset, H. Coverage for robotics—A survey of recent results. *Ann. Math. Artif. Intell.* **2001**, *31*, 113–126. [[CrossRef](#)]
14. Galceran, E.; Carreras, M. A survey on coverage path planning for robotics. *Robot. Auton. Syst.* **2013**, *61*, 1258–1276. [[CrossRef](#)]
15. Jiao, Y.S.; Wang, X.M.; Chen, H.; Li, Y. Research on the coverage path planning of uavs for polygon areas. In Proceedings of the 2010 5th IEEE Conference on Industrial Electronics and Applications, Taichung, Taiwan, 15–17 June 2010; pp. 1467–1472.
16. Li, Y.; Chen, H.; Er, M.J.; Wang, X. Coverage path planning for UAVs based on enhanced exact cellular decomposition method. *Mechatronics* **2011**, *21*, 876–885. [[CrossRef](#)]
17. Torres, M.; Pelta, D.A.; Verdegay, J.L.; Torres, J.C. Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. *Expert Syst. Appl.* **2016**, *55*, 441–451. [[CrossRef](#)]
18. Coombes, M.; Chen, W.H.; Liu, C. Boustrophedon coverage path planning for UAV aerial surveys in wind. In Proceedings of the 2017 International Conference on Unmanned Aircraft Systems (ICUAS), Miami, FL USA, 13–16 June 2017; pp. 1563–1571.
19. Coombes, M.; Fletcher, T.; Chen, W.H.; Liu, C. Optimal polygon decomposition for UAV survey coverage path planning in wind. *Sensors* **2018**, *18*, 2132. [[CrossRef](#)]
20. Xu, A.; Viriyasuthee, C.; Rekleitis, I. Optimal complete terrain coverage using an unmanned aerial vehicle. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 2513–2519.
21. Cabreira, T.M.; Di Franco, C.; Ferreira, P.R.; Buttazzo, G.C. Energy-aware spiral coverage path planning for uav photogrammetric applications. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3662–3668. [[CrossRef](#)]
22. Öst, G. Search path generation with UAV applications using approximate convex decomposition, Department of Electrical, Linköpings universitet, 2012. Available online: <http://www.diva-portal.org/smash/get/diva2:526417/FULLTEXT01.pdf> (accessed on 9 September 2021).
23. Maza, I.; Ollero, A. Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms. In *Distributed Autonomous Robotic Systems 6*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 221–230.
24. Balampanis, F.; Maza, I.; Ollero, A. Coastal areas division and coverage with multiple UAVs for remote sensing. *Sensors* **2017**, *17*, 808. [[CrossRef](#)]
25. Balampanis, F.; Maza, I.; Ollero, A. Spiral-like coverage path planning for multiple heterogeneous UAS operating in coastal regions. In Proceedings of the 2017 International Conference on Unmanned Aircraft Systems, (ICUAS), Miami, FL, USA, 13–16 June 2017; pp. 617–624. [[CrossRef](#)]
26. Valente, J.; Sanz, D.; Del Cerro, J.; Barrientos, A.; de Frutos, M.Á. Near-optimal coverage trajectories for image mosaicing using a mini quad-rotor over irregular-shaped fields. *Precis. Agric.* **2013**, *14*, 115–132. [[CrossRef](#)]
27. Bouzid, Y.; Bestaoui, Y.; Siguerdidjane, H. Quadrotor-UAV optimal coverage path planning in cluttered environment with a limited onboard energy. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Vancouver, BC, Canada, 24–28 September 2017; pp. 979–984. [[CrossRef](#)]
28. Barrientos, A.; Colorado, J.; Cerro, J.d.; Martinez, A.; Rossi, C.; Sanz, D.; Valente, J. Aerial remote sensing in agriculture: A practical approach to area coverage and path planning for fleets of mini aerial robots. *J. Field Robot.* **2011**, *28*, 667–689. [[CrossRef](#)]
29. Almadhoun, R.; Taha, T.; Seneviratne, L.; Zweiri, Y. A survey on multi-robot coverage path planning for model reconstruction and mapping. *SN Appl. Sci.* **2019**, *1*, 1–24. [[CrossRef](#)]
30. Sadat, S.A.; Wawerla, J.; Vaughan, R.T. Recursive non-uniform coverage of unknown terrains for UAVs. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Chicago, IL USA, 14–18 September 2014; pp. 1742–1747. [[CrossRef](#)]
31. Sadat, S.A.; Wawerla, J.; Vaughan, R. Fractal trajectories for online non-uniform aerial coverage. In Proceedings of the IEEE International Conference on Robotics and Automation, Seattle, WA, USA, 25–30 May 2015; pp. 2971–2976. [[CrossRef](#)]
32. Santamaria, E.; Segor, F.; Tchouchenkov, I. Rapid aerial mapping with multiple heterogeneous unmanned vehicles. In Proceedings of the ISCRAM 2013 Conference–10th International Conference on Information Systems for Crisis Response and Management, Baden-Baden, Germany, 12–15 May, 2013; pp. 592–596.

33. Sauter, J.A.; Matthews, R.; Van Dyke Parunak, H.; Brueckner, S.A. Performance of digital pheromones for swarming vehicle control. In Proceedings of the International Conference on Autonomous Agents, Utrecht, Netherlands, 25–29 July 2005; pp. 1037–1044. [[CrossRef](#)]
34. Paradzik, M.; Ince, G. Multi-agent search strategy based on digital pheromones for uavs. In Proceedings of the 2016 24th Signal Processing and Communication Application Conference (SIU), Zonguldak, Turkey, 16–19 May 2016; pp. 233–236.
35. Bircher, A.; Kamel, M.; Alexis, K.; Burri, M.; Oettershagen, P.; Omari, S.; Mantel, T.; Siegwart, R. Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots. *Auton. Robot.* **2016**, *40*, 1059–1078.
36. Ghaddar, A.; Merei, A. EA OA: Energy-Aware Grid-Based 3D-Obstacle Avoidance in Coverage Path Planning for UAVs. *Future Internet* **2020**, *12*, 29. [[CrossRef](#)]
37. Cabreira, T.M.; Ferreira, P.R.; Di Franco, C.; Buttazzo, G.C. Grid-based coverage path planning with minimum energy over irregular-shaped areas with UAVs. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA USA, 11–14 June 2019; pp. 758–767.
38. Muñoz, J.; López, B.; Quevedo, F.; Monje, C.A.; Garrido, S.; Moreno, L.E. Coverage Strategy for Target Location in Marine Environments Using Fixed-Wing UAVs. *Drones* **2021**, *5*. [[CrossRef](#)]
39. Chen, J.; Du, C.; Zhang, Y.; Han, P.; Wei, W. A clustering-based coverage path planning method for autonomous heterogeneous UAVs. *IEEE Trans. Intell. Transp. Syst.* **2021**. [[CrossRef](#)]
40. Cho, S.W.; Park, H.J.; Lee, H.; Shim, D.H.; Kim, S.Y. Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations. *Comput. Ind. Eng.* **2021**, *161*, 107612. [[CrossRef](#)]
41. Melo, A.G.; Pinto, M.F.; Marcato, A.L.; Honório, L.M.; Coelho, F.O. Dynamic Optimization and Heuristics Based Online Coverage Path Planning in 3D Environment for UAVs. *Sensors* **2021**, *21*, 1108. [[CrossRef](#)]
42. Von Stackelberg, H. *Marktform und Gleichgewicht*; J. Springer: Berlin, Germany, 1934.
43. López, B.; Muñoz, J.; Quevedo, F.; Monje, C.A.; Garrido, S.; Moreno, L.E. Path Planning and Collision Risk Management Strategy for Multi-UAV Systems in 3D Environments. *Sensors* **2021**, *21*, 4414. [[CrossRef](#)]
44. Gómez, J.V.; Lumbier, A.; Garrido, S.; Moreno, L. Planning robot formations with fast marching square including uncertainty conditions. *Robot. Auton. Syst.* **2013**, *61*, 137–152.
45. Sethian, J.A. Fast marching methods. *SIAM Rev.* **1999**, *41*, 199–235. [[CrossRef](#)]
46. IGN–Instituto Geográfico Nacional. Available online: <https://www.ign.es/web/ign/portal/qsm-cnig> (accessed on 9 September 2021).
47. Garrido, S.; Moreno, L.; Blanco, D.; Martin, F. FM2: A real-time fast marching sensor-based motion planner. In Proceedings of the 2007 IEEE/ASME international conference on advanced intelligent mechatronics, Zurich, Switzerland, 4–7 September 2007; pp. 1–6.
48. Coulter, R.C. *Implementation of the Pure Pursuit Path Tracking Algorithm*; Technical report; Carnegie-Mellon UNIV Pittsburgh PA Robotics INST: Pittsburgh, PA, USA, 1992.
49. Yatziv, L.; Bartesaghi, A.; Sapiro, G. O (N) implementation of the fast marching algorithm. *J. Comput. Phys.* **2006**, *212*, 393–399. [[CrossRef](#)]