Multi-constrained cooperative path planning of multiple drones for persistent surveillance in urban environments

Yu Wu^{1,2} · Shaobo Wu³ · Xinting Hu⁴

Received: 10 September 2020 / Accepted: 9 February 2021 / Published online: 22 February 2021
© The Author(s) 2021

Abstract

Different from the usual surveillance task in which the goal is to achieve complete coverage of the specified area, the cooperative path planning problem of drones for persistent surveillance task is studied in this paper considering multiple constraints of the covered area. The goal is to maximize the combinational coverage area of drones while giving preference to the area that hasn't been visited beyond a certain time interval. The influence of shooting resolution and blocking of buildings are considered, and the state information of each grid is defined to record the visit information of the ground area. Considering the characteristic of the established model, the multi-constrained cooperative path planning (MCCPP) algorithm is developed. The grids which have not been visited for a long time are received special attentions, and the drone is led to reducing the flight height to cover the grid which has a special requirement on the shooting resolution. The cooperation mechanism among drones is also set to ensure that all the drones can determine the next path point synchronously. An emergency path planning algorithm with the continuous checking strategy is designed for a drone to fly to the specified area and finish a complete coverage of it.

Keywords Drones · Urban environments · Persistent surveillance · Cooperative path planning · Emergency path planning

Introduction

Drones have been increasingly popular in civil applications due to their easy operation and the open of low-altitude airspace. Typical applications contain delivery [1], target tracking [2], earth observation [3], surface inspection [4] and so on. In those tasks, drones must fly along the designed path, and the path planning problem becomes the key issue to ensure the safety and the efficiency of completing specific tasks. In general, the path planning problems for drone flying can be classified into two categories, i.e., the target-oriented problem and the area coverage problem. In the

target-oriented problem, drones are required to reach the target [5]. For example, drones can reach their perspective targets for different purposes in a dynamic environment after resolving the conflicts [6]. Different from the target-oriented problem, no specific destination is defined in the area coverage problem, and the goal is usually to realize a complete coverage of the specific area with the shortest time or path length [7]. In urban environments, the above two problems are sometimes both involved in a complicated situation, such as in the persistent surveillance task. The multi-constrained cooperative path planning problem for persistent surveillance is studied in this paper.

Considering the requirement of air traffic management for drone operations, the low-altitude airspace in urban environments is divided into cubes with the same size, and the vertexes of cubes are regarded as the path points of drones, like the operation of traffic management in civil aviation [8]. The ground area is also modeled as a gridding structure, which is the projection of 3D-cube airspace on the ground. A cooperative path planning model is established to describe the persistent surveillance task, and a multi-constrained cooperative path planning (MCCPP) algorithm is developed to generate the flight path for each drone. The goal of persistent

✉ Yu Wu
cquwuyu@cqu.edu.cn

¹ College of Aerospace Engineering, Chongqing University, Chongqing 400044, China

² School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798, Singapore

³ Automation School, Beijing University of Posts and Telecommunications, Beijing 100876, China

⁴ School of Air Traffic Management, Civil Aviation University of China, Tianjin 300300, China

surveillance task is to maximize the coverage area as well as considering the areas which have not been visited for a long time or have special requirement on the shooting resolution. Besides, an emergency is considered in the proposed method, and the corresponding countermeasure is taken to make the drone reach the specified area while performing the persistent surveillance task. The main contributions of this work can be concluded as follows:

1. A new cooperative path planning model for persistent surveillance in urban environments is established. The influences of drone's flight height on the field of vision and the shooting resolution are considered, and the effect of the blocking of buildings is also involved. The combinational coverage area of multiple drones is calculated, and the state information denoting the visit status of each grid is defined, which is essential to guide the selection of path points.
2. A MCCPP algorithm is developed. In this algorithm, the flight rules of drone and the characteristic of the coverage area are considered. The preference is given to the grids which have not been visited beyond a certain time interval, and the special requirement of some grids on the shooting resolution is also paid attention to. The cooperation mechanism among drones is set to improve the efficiency of persistent surveillance task.
3. An emergency path planning algorithm is designed for the persistent surveillance task. With this algorithm, the drone closest to the specified area is assigned to reach and have complete coverage of it, and all drones still perform the normal persistent surveillance cooperatively. By the continuous checking strategy, the drone heading for the specified area can maximize its coverage area while satisfying the constraint on the latest arrival time.

Related works

The coverage path planning (CPP) problem for an unmanned aerial vehicle (UAV) has been extensively studied in the existing literature. According to the complexity of decomposing the coverage area, three groups can be classified, i.e., no decomposition, approximate decomposition, and exact decomposition. No decomposition means the area is regarded as a whole, and the drone can fly with the back-and-forth pattern or the spiral pattern to achieve complete coverage. The areas with regular shapes are usually fit for the no decomposition mode. In the back-and-forth pattern, the path is composed of straight lines with semicircle curves at the end of each round. In the spiral pattern, the radius of the spiral is reduced with the increase of flight height to cover the vertices of the object [9]. In Ref. [10], four different flight patterns are compared when only the rectangular areas

are considered. The back-and-forth pattern performs better when the area is large, and the square flight pattern is suitable for the case when a uniform area coverage is expected. In recent studies, energy consumption is considered in the CPP problem for UAVs [11]. A new route-based optimization model with column generation is introduced in Ref. [12] to calculate the required energy of UAVs in different flight phases in view of that the traditional arc-based optimization approaches fail to estimate the energy consumption in the turning phases. For the problem of covering a number of areas with regular shapes, it is formulated into a traveling sale problem (TSP) with the goal of covering all areas completely with the shortest path. In Ref. [13], a footprints' sweep fitting method is proposed to determine the visiting sequence of footprints' sweeps, and ant colony optimization (ACO) algorithm is designed to solve the formulated TSP. A similar problem is also considered when the areas are divided into grids. The TSP and CPP problem are combined into a variant TSP-CPP problem in Ref. [14], and a grid-based approach and a dynamic programming-based approach are both utilized to search the (near) optimal path for the UAV to cover multiple regions.

The second way of dealing with the areas to be covered is the approximate decomposition. It is necessary to conduct the approximate decomposition when the area is with irregular shape and multiple distributed obstacles. The whole area is divided into grids with the same size, as in the above-mentioned Ref. [14], and the size of grids is determined by the volume of obstacle and the detection capability of sensors equipped on UAV. Considering that the back-and-forth flight pattern is inefficient when covering an area with an irregular shape, a CPP approach for image mosaicking in precision agriculture is proposed in Ref. [15]. The gridding area is converted to a regular shape and is labeled by the Wavefront algorithm to denote the relationships of adjacent grids. In Ref. [16], a constrained CPP problem considering the time limit and the feasibility of the path is addressed, and a new genetic algorithm (GA) with adaptive operator selection is proposed. With the gridding area, the number of path points is certain, which makes the new GA algorithm suitable for solving this problem. In some cases, the information of the environment is not always known to the drone, online CPP algorithms should be developed. In Ref. [17], a ϵ^* algorithm is designed based on the concept of an exploratory turning machine (ETM), and the vehicle is guided by a supervisor with adaptive navigation commands. The online coverage path is generated by using multiscale adaptive potential surfaces based on the updated information. For multiple drones, the information collected by individuals should be shared to improve the efficiency of searching the area, and a method of merging information for cooperative search is presented in Ref. [18]. Due to the restriction of sensing capabilities, each UAV can only sense the information of the adjacent

grids, and the global search map is updated by sharing the local information with each other. In this way, the sensing capabilities of UAV can be improved indirectly without a cooperative approach.

As for the exact decomposition, it is usually applied considering the size and the complexity of the area, and the irregular areas can be divided into small pieces to reduce the concavities and the difficulty of achieving complete coverage. Then the sub-areas can be searched by a single or multiple UAVs with no decomposition or approximate decomposition approaches. When the coverage task is performed by a single UAV, the connection between adjacent sub-areas must be paid special attention. In Ref. [19], after the exact decomposition, an algorithm which can address both convex and non-convex areas is proposed. Four coverage alternatives meeting all the constraints are designed to reduce the transition distance between different sub-areas, and an interrupted line sweep concept is introduced to judge whether a non-convex polygon decomposition is reasonable. For multiple UAVs carrying out the tasks of covering sub-areas, task allocation is needed considering the capabilities of each UAV. In Ref. [20], the inspection task for a 3D infrastructure with a complex shape is completed by multiple UAVs. A theoretical framework is established to provide the path for each UAV to realize the full coverage of the infrastructure, and the infrastructure is sliced by XOY plane to make it easier to identify branches and allocate certain areas to each UAV. Besides, the advantages and drawbacks are demonstrated when the method is applied in a real situation.

In the above studies, the goal of CPP problem is usually to realize a complete coverage of specific areas with the shortest path or flight time. However, for the surveillance task in urban environments, the ground area should be monitored persistently to have a global view and find the unusual situation in time. For example, drones can be employed in public security to make an inspection tour of various places and check whether there is anything abnormal. If there are people entering the forbidden area or committing an illegal action, the warning will be sent to remind them, and further action will be taken by the police. In such cases, persistent surveillance is needed to satisfy the demand, and the task goal is changed correspondingly instead of realizing a complete coverage. Besides, the field of vision and the shooting resolution for the sensors equipped on the drone are contradictory, and a higher shooting resolution can be achieved by reducing the flight height of the drone. The buildings distributed over the urban environments will also block the field of vision of sensors. All of the above factors are not given adequate considerations in the existing studies, and a new path planning algorithm should be developed when those elements are involved in the path planning model for a persistent surveillance task.

Path planning model for the persistent surveillance task

The urban space is divided into a number of cubes of the same size, and the flight rules of drones under the discrete urban environments are defined. With those general formulations, the model of drone's coverage areas at different flight heights considering the blocking of buildings is established. The formulas of judging whether two drones collide with each other are presented, and the case of the overlapping coverage area for multiple drones are considered. At last, the state value of each grid is defined to record its visit information. The goal of persistent surveillance is to maximize the coverage area while giving preference to the grids which have not been visited for a certain period.

Definition of urban environments and the flight rules of drone

The low-altitude airspace in urban environments is divided by cubes with the same size, and the space occupied by the buildings is also denoted by the cubes, as shown in Fig. 1.

In Fig. 1, the ground coordinate system OXYZ is established to describe the locations of buildings and drones. The black cubes are the spaces occupied by the buildings with different heights, and drones can only fly passing the vertices of white cubes and should keep a certain distance from the buildings. The drone can choose the vertex of a connected edge, face diagonal or body diagonal as the next path point, and it flies along the straight line between two neighboring path points. Note that the side length of a cube is decided after a comprehensive consideration of the maneuverability of the drone and the optimality of the flight path. To be specific, the side length of a cube cannot be too small because the drone may not be able to turn a large angle. As the flight path is composed of a number of line segments, in real flight, the drone can track each path point one after another by using the flight controller. On the other hand, the side length of a cube cannot be too large, which will reduce

Fig. 1 Definition of the gridding urban environments with the flight rules of drone

the times of turn and the flight efficiency. Assume that the drone flies with a constant velocity, the motion equations of the drone can be expressed as follows:

$$\begin{cases} x_i(s+1) = x_i(s) + ax_i(s) \cdot l \\ y_i(s+1) = y_i(s) + ay_i(s) \cdot l \\ z_i(s+1) = z_i(s) + az_i(s) \cdot l \\ t_i(s+1) = t_i(s) \end{cases}, i = 1, 2, \dots, N_D$$

$$+ \sqrt{\text{sign}(ax_i(s)) + \text{sign}(ay_i(s)) + \text{sign}(az_i(s))} \cdot l/v \quad (1)$$

where $(x_i(s), y_i(s), z_i(s), t_i(s))$ is the 4D position information of drone i at the s th path point, N_D is the number of drones, l is the side length of the cube and v is the flight velocity of drone. ax_i , ay_i and az_i are the control variables in direction OX, OY and OZ, and their values can be selected from the set $D = \{-1, 0, 1\}$. The item $\text{sign}(ax_i(s)) + \text{sign}(ay_i(s)) + \text{sign}(az_i(s))$ can record if the drone flies along a connected edge, face diagonal or body diagonal. The operator sign is defined as follows:

$$\text{sign}\{\text{value}\} = \begin{cases} 1, & \text{value} > 0 \\ 0, & \text{value} = 0 \\ -1, & \text{value} < 0 \end{cases} \quad (2)$$

Besides, the drone should keep a certain distance from the building to ensure flight safety, as denoted in Eq. (3).

$$\min \left\{ \sqrt{(x_i(s) - \tilde{x})^2 + (y_i(s) - \tilde{y})^2 + (z_i(s) - \tilde{z})^2} \right\} \geq l,$$

$$(\tilde{x}, \tilde{y}, \tilde{z}) \in V_B \quad (3)$$

where V_B is the space occupied by the buildings, and $(\tilde{x}, \tilde{y}, \tilde{z})$ is a point belonging to V_B . With the constraint in Eq. (3), the distance between the drone and a building should be no less than l . Note that the maximum range of drone is not taken into account directly in this study because in the persistent surveillance task, each path point of drone is generated step by step. To be specific, if there is not sufficient range-to-go for the drone in one step, it can stop working immediately, which will not have an influence on the generated path points.

Model of drone's coverage area considering the blocking of buildings

According to the flight rules of the drone, the path point is at the vertex of a cube. This is different from many previous studies that the drone is always located in the center of a grid [21], and the area of the grid is just the coverage area for the drone [22]. Assume that the current location of a

drone is (x_o, y_o, z_o) , the coverage area can be denoted as the following equation.

$$\begin{cases} X_l = \max\{x_{\min}, x_o - z_o\} \\ X_r = \min\{x_o + z_o, x_{\max}\} \\ Y_l = \max\{y_{\min}, y_o - z_o\} \\ Y_r = \min\{y_o + z_o, y_{\max}\} \end{cases} \quad (4)$$

where $[x_{\min}, x_{\max}]$ and $[y_{\min}, y_{\max}]$ are the permitted ranges of the low-altitude airspace in direction OX and OY respectively. The coverage area is a rectangle, and the intervals $[X_l, X_r]$ and $[Y_l, Y_r]$ are the boundaries of coverage area in direction OX and OY for the drone located at the point (x_o, y_o, z_o) . z_o , which can only take some discrete values, is appeared in Eq. (4) to express that the coverage area is enlarged with the increase of flight height. According to Eq. (4), it can be learned that the coverage area of a drone in 2D space increases with a greater flight height, which is consistent with the actual situation. Take $z_o = l$ and $z_o = 2l$ as examples, the coverage area corresponding to the two cases are shown in Fig. 2.

Note that, in Fig. 2a, it is impossible for the building occupying one of the four grids, or the constraint in Eq. (3) will be violated. In Fig. 2b, there is a building occupying a grid, and the grid should be excluded when calculating the coverage area. In more complicated cases, the buildings may block the field of vision of sensors, which leads to the extra invalid coverage area of the drone. An example is given in Fig. 3 to present the situation when $z_o = 3l$.

In Fig. 3, the shaded areas are blocked by the buildings from the current position of the drone, and they also should be excluded as the covered area. Those grids can be covered when the drone is located at other positions where there is no building blocking the field of vision.

Fig. 2 Examples of coverage area when the drone is located at different heights

Fig. 3 Diagram of the invalid coverage area caused by the blocking of buildings

Collision avoidance and the combinational coverage area considering multiple drones' flying

Drones should keep a certain distance from each other to avoid collision, and there are two forms of collisions between drones. The first one is that two drones pass the same path point. In this case, their time intervals of reaching the same path point must be greater than a safe threshold, as presented in Eq. (5).

$$\begin{cases} x_i(s) = x_j(s), y_i(s) = y_j(s), z_i(s) = z_j(s) \\ \left| t_i(s) - t_j(s) \right| > \Delta t \end{cases} \quad (5)$$

where $t_i(s) | (x_i(s), y_i(s), z_i(s))$ denotes the time information of path point $(x_i(s), y_i(s), z_i(s))$ for drone i , and Δt is the defined safety threshold.

The second form of collision between drones happens at the intersection of two path segments, as shown in Fig. 4.

In Fig. 4, although drone i and j are both safe at the four path points, they may collide with each other during the process of flying from $(x_i(s), y_i(s), z_i(s), t_i(s))$ to $(x_i(s+1), y_i(s+1), z_i(s+1), t_i(s+1))$ (or from $(x_j(s), y_j(s), z_j(s), t_j(s))$ to $(x_j(s+1), y_j(s+1), z_j(s+1), t_j(s+1))$) if the following equation is met.

Fig. 4 The cases of right-angle collision between two drones

$$\begin{cases} x_i(s) + x_i(s+1) = x_j(s) + x_j(s+1) \\ y_i(s) + y_i(s+1) = y_j(s) + y_j(s+1) \\ z_i(s) + z_i(s+1) = z_j(s) + z_j(s+1) \\ t_i(s) > t_j(s+1) \text{ or } t_j(s) > t_i(s+1) \end{cases} \quad (6)$$

Equation (6) denotes that drone i and j collide with each other at the middle of two adjacent path points.

Besides, as the persistent surveillance task is performed by multiple drones, there may be overlapping grids among their coverage area. Assume that the area covered by $(x_i(s), y_i(s), z_i(s), t_i(s))$ is $A_i(s)$, the total valid coverage area of multiple drones at step s can be denoted as:

$$A(s) = A_1(s) \cup A_2(s) \cup \dots \cup A_{N_D}(s), i = 1, 2, \dots, N_D \quad (7)$$

where $A(s)$ is the total valid coverage area of drones at the step s .

Model of calculating the state information of the grid

Different from the complete coverage path planning problem [23], the goal of the persistent surveillance task is to cover the grids as many as possible while giving preference to the grids which have not been visited beyond a certain period. It is not enough to use the sign '0' or '1' to denote whether the grid has ever been visited in this task [24]. A state value is introduced to describe the visit time interval of each grid and can be calculated as follows.

$$sv_k(s) = \begin{cases} -1, & \text{if it is occupied by a building} \\ 1 - \frac{t_v}{t_c(s)}, & \text{others} \end{cases} \quad (8)$$

where $sv_k(s)$ is the state value of the grid No. k , and t_v is the moment that the grid was visited last time. If the grid has not ever been visited, t_v is set to 0, and its state value is $sv_k(s) = 1$. $t_c(s)$ is the current moment, and a greater gap between t_v and $t_c(s)$ makes a larger value of $sv_k(s)$. $sv_k = -1$ is always true for the grid occupied by a building. According to Eq. (8), the index of evaluating the persistent surveillance task can be expressed as follows.

$$J(s) = \sum_{m=1}^{g(s)} sv_m(s) \quad (9)$$

where $g(s)$ is the number of covered grids at the step s , and $J(s)$ is the index. According to the form of $J(s)$, both the number of covered grids and the visit time interval of each grid are considered, which can satisfy the requirement of the persistent surveillance task. The goal is to maximize the value of $J(s)$ in each step of path planning.

Multi-constrained cooperative path planning algorithm for persistent surveillance

Based on the characteristic of the established model, the MCCPP algorithm is developed considering the preference on the grids which have not been visited beyond a certain period and the special requirement on the shooting resolution for some grids. The cooperative mechanism among drones is also designed to realize efficient persistent surveillance.

Approaches of addressing special situations

As shown in Eq. (9), the drone will choose the spare path point maximizing the value of $J(s)$. It indicates that the drone will prefer to fly at the maximum permitted height because a greater flight height means that more grids can be covered, which leads to a higher probability of getting a larger value of $J(s)$. There are two types of special situations, and the expression of index $J(s)$ must be modified to satisfy the requirements.

- 1) The grid has not been visited for a certain period.

In this circumstance, if one of the covered grids has not been visited beyond the time threshold $\Delta T_v (t_c(s) - t_v > \Delta T_v)$, $J(s)$ can be denoted as follows to pay special attention to the grid.

$$J(s) = g(s) \cdot \max(sv_1(s), sv_2(s), \dots, sv_{g(s)}(s)) \quad (10)$$

In Eq. (10), it demonstrates that if long-unvisited grids are included in the coverage area, the corresponding

spare path point will be selected with a higher probability by increasing the index value. $g(s)$ is the number of grids corresponding to the coverage area, and $J(s)$ is the product of $g(s)$ and the maximum state value of the grids.

- 2) The grid is required to be visited with a higher shooting resolution.

Coverage with higher shooting resolution can be realized by reducing the flight height of the drone. As the drone prefers to fly at the maximum permitted height, the index in Eq. (9) must be modified to guide the drone to cover the specified area with the required shooting resolution.

The spare path points which can potentially visit the specified area in the future few steps are preferred, and the following two conditions must be met.

- i) There is a path starting from the spare path point, and the destination of the path can cover the specified area with the required shooting resolution. Besides, the flight height of the drone must be always declining from the spare path point to the destination to ensure the shortest path. This condition can be described in Fig. 5.

In Fig. 5, the black grid is required to be visited when the flight height of the drone is l , and two spare path points are taken as examples. There is a path from point P1 to point P2 which the drone can always reduce its flight height, and the grid can be visited with the required shooting resolution. The index of point P1 will be calculated by a new formulation if the second condition also can be met. As for the spare point Q1, its index value is still calculated by Eq. (9).

- ii) The grids in the specified area should conform to the first special situation to avoid being visited with high frequency.

Assume that the maximum permitted flight height is $z_{\max}$, the index value of the spare path point satisfying the above two conditions are set as follows.

$$J(s) = g(s) \cdot \max(sv_1(s), sv_2(s), \dots, sv_{g(s)}(s)) + \alpha \cdot (z_{\max} - z_{sp}) \quad (11)$$

In Eq. (11), the item $\alpha \cdot (z_{\max} - z_{sp})$ is added to the index in Eq. (10), where α is a parameter determining the weight of the corresponding item, and z_{sp} is the height of the spare path point. With Eq. (11), the spare path points which can potentially visit the specified area are distinguished by the

Fig. 5 The scene described in condition i)

second item of $J(s)$ in Eq. (11), and they are further selected according to the value of the first item of $J(s)$. In this way, the spare path points are treated by different formulas, and it is possible for the drone to reduce the flight height to visit the special area.

Cooperation mechanism among the drones

In each step of path planning, the drone determines its next path point by maximizing the index value according to Eq. (9), (10) or (11). When multiple drones are assigned to perform the persistent surveillance task, their paths should be coordinated considering the flight time and the overlap among the coverage area of drones. A cooperation mechanism is introduced to deal with the above two aspects.

Note that there are three types of path segment for a drone flying to the next path point, i.e., along a connected edge, face diagonal or body diagonal (called T1, T2 and T3 respectively in the rest of this paper for convenience). If the drones choose different types of path segments in the same step to maximize the index value individually, there will be a problem that drones will reach the next path point at different moments. In this case, the cooperation among drones cannot be realized as the detection of collision avoidance is unavailable, and the coordination among drones cannot be reached at the same moment.

In view of the above situation, the drones should be restricted to fly with the same type of path segment in each step. Although it may sacrifice the optimality of the coverage area for each drone, the cooperation among drones can be realized, which is beneficial for improving the efficiency of the cooperative persistent surveillance task. The detailed procedures of the cooperation mechanism at the step s are concluded below, and a corresponding diagram also can be seen in Fig. 6.

Step 1: For each drone, classify the spare path points into three groups according to the length of path segments.

Step 2: Calculate the index value of every spare path point and sort those index values from the largest to the smallest. This operation is made in each of the three groups.

Step 3: In each of the three groups, find the best combination of spare path points for multiple drones using the greedy algorithm.

Step 4: Select the best type of path segment according to the index value, and the corresponding combination of spare path points is outputted to achieve the next path point for each drone.

In Step 2, if there is no spare path point corresponding to a certain type of path segment for a drone in step s , this type of path segment will not be considered anymore as the cooperation among drones cannot be realized. Besides, the combination of spare points is evaluated by calculating the index value, and the overlap coverage area for multiple drones should be counted only once. With the same flight velocity, the drones can reach the next path point, respectively at the same time by the above cooperation mechanism, and their next path points are determined in a centralized way to maximize the index value.

Emergency path planning algorithm for persistent surveillance

The proposed MCCPP algorithm can ensure the grids being visited with appropriate frequency. Sometimes, an emergency may occur in a certain area, and the drone must fly to the specified area right now to report the latest situation. The emergency path planning algorithm under such a case is dealt with in this section.

Fig. 6 A general description of the proposed cooperation mechanism among drones

Description of the path planning problem in emergency

Under normal circumstances, drones are flying to perform the persistent surveillance task in urban environments. When an emergency occurs in a certain area, the drone which is the closest one to the specified area (named as the special drone for convenience in the rest of this paper) will be assigned to reach there. After a complete coverage of the specified area, the special drone can return to the normal persistent surveillance task as before. In the whole process, the other drones must cooperate with the special drone to perform the surveillance task, and the special drone is required to finish a complete coverage of the specified area. To guarantee that the special drone can reach the specified area in time, the flight time must be shorter than a threshold. The constraint can be expressed in Eq. (12).

$$T_r \leq T_{\max} = 2(|x_c - x_{\text{tar}}| + |y_c - y_{\text{tar}}| + |z_c - z_{\text{tar}}|)/v \quad (12)$$

where T_r and $T_{\max}$ are the actual flight time and the maximum flight time respectively, (x_c, y_c, z_c) is the position of the special drone when it receives the command to fly to the specified area, and $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ is the position of a point which can cover some grids of the specified area and is the closest to the current position of the special drone. The setting of $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ can make the special drone reach the specified area as soon as possible, and the details of determining $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ will be elaborated when designing the path planning algorithm. In Eq. (12), the flying distance of the special drone must be no larger than twice the Manhattan distance from the starting point to the target. It demonstrates that the special drone is allowed to make a detour to cover more areas during the travel to the emergency area. Then the emergency path planning problem can be formulated as follows: Finding a path from the starting point to the target to maximize the index value $J(s)$ while satisfying the constraint in Eq. (12).

The emergency path planning algorithm with continuous checking strategy

First, the approach of determining the position of the target $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ is discussed. The target is defined as the point which can cover some grids of the specified area and is the closest to the current position of the special drone. In the flying process, the current position of the special drone is always changing, which also makes $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ change dynamically. It increases the difficulty of achieving complete coverage of the specified area because the starting point for the complete coverage path planning problem is

uncertain. In many studies, the starting point is selected at the boundary or the corner of the area to avoid the repeated path. It fails to be applied in this problem because the starting point for the complete coverage (also the target $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$) cannot be randomly selected.

To avoid the bad situation that the complete coverage is influenced by the position of the starting point, Hamilton cycle is introduced to obtain a loop path which can achieve complete coverage of the specified areas. For the special drone, when it reaches the target $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ (the target must be one of the path points in the Hamilton cycle), it just needs to fly across each path point in the Hamilton cycle one by one to achieve the complete coverage of the emergency area. There have been many algorithms calculating the Hamilton cycle, such as the linear-time algorithm [25], the polynomial algorithms [26] and parallel algorithm [27], and it will not be explored in this paper.

Another concern for the special drone is how to satisfy the constraint in Eq. (12) while maximizing the index value $J(s)$. Here the continuous checking strategy is designed, and the procedures are presented as follows.

Step 1: In the step s , list all the spare path points and calculate the index value $J(s)$. Generate the obstacle-free paths from each spare path point to the corresponding target by applying A* algorithm and record the length of these paths as d_{sp} ($\text{sp} = 1, 2, \dots, N_{\text{sp}}$), where N_{sp} is the number of spare path points. Turn to Step 2.

Step 2: Find the spare path point with the maximum value of $J(s)$ and satisfying $d_{\text{sp}} < v \cdot (T_{\max} - T_{\text{ela}})$ (T_{ela} is the elapsed time flying from (x_c, y_c, z_c) to the spare path point) and choose it as the next path point. Turn to Step 1. If none of the spare path points satisfies $d_{\text{sp}} < v \cdot (T_{\max} - T_{\text{ela}})$, turn to Step 3.

Step 3: The obstacle-free path from the current position of the special drone (denoted as $(x_{\text{now}}, y_{\text{now}}, z_{\text{now}})$) to the corresponding target $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ generated by A* algorithm is treated as the rest path, and all the path points from (x_c, y_c, z_c) to $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ are determined.

In the above procedures, A* algorithm plays two roles. The first one is to check whether the shortest path from the spare path point to the corresponding target can meet the constraint in Eq. (12). The second function is to generate the optimal path from $(x_{\text{now}}, y_{\text{now}}, z_{\text{now}})$ to $(x_{\text{tar}}, y_{\text{tar}}, z_{\text{tar}})$ when all the spare path points fail to satisfy the constraint in Eq. (12).

With the emergency path planning algorithm, the special drone can reach the target in time while maximizing its coverage area. Then the complete coverage of the specified area is achieved by flying across each path point in the Hamilton cycle one by one. Figure 7 is provided to show the scene of the drone having a complete coverage of the specified area.

Fig. 7 Drone is having a complete coverage of the specified area

Cooperation among drones in emergency

The emergency path planning algorithm for the special drone has been explained above. When there is an emergency in the cooperative persistent surveillance task, the special drone is with the highest priority to ensure that it can reach the target in time. The paths of other drones should be generated after the next path point of the special drone has been determined. As with the designed cooperation mechanism, the same is true in an emergency that every drone should fly with the same type of path segment in the same step. In other words, the other drones should apply the same type of path segment following the special drone. The comprehensive flow of the MCCPP algorithm for persistent surveillance integrating the case of emergency is presented in Fig. 8.

In normal circumstances, all the drones determine their next path points synchronously in each step. While in an emergency, the special drone has the privilege to calculate its next path point first, and other drones still obtain their

next path points in a cooperative way considering the action of the special drone. The persistent surveillance task will not be stopped until the terminal condition is reached. The terminal condition is often set as finishing a complete coverage of the area or performing the persistent surveillance task over a certain period of time considering the maximum range of drone.

Simulation studies

To investigate the rationality of the established model and the superiority of the MCCPP algorithm of multiple drones for persistent surveillance task, three groups of simulations are conducted. In the first simulation, five drones are assigned to perform the persistent surveillance task, and the effectiveness of the MCCPP algorithm is explained. Simulations with different sizes of urban space and densities of buildings are also conducted to explore their influences on the results. In the second scenario, an emergency case is considered in the process of the persistent surveillance task, and the validity of the emergency path planning algorithm is verified. Then, the comparison is made between the proposed algorithm and the partitioning-based path planning algorithm in the persistent surveillance task. In all the above simulations, the setting of the urban environments is the same if it is not otherwise specified. There are 1000 buildings randomly distributed, and some areas which have special requirement on the shooting resolution are set, as shown in Fig. 9.

Fig. 8 Flow chart of the MCCPP algorithm

Fig. 9 Setting of urban environments and persistent surveillance task

Table 1 Parameters setting in simulations

Parameter	Value	Parameter	Value
l	30 m	v	10 m/s
Δt	5 s	$[x_{\min}, x_{\max}]$	$[0, 3000]$ m
ΔT_v	600 s	$[y_{\min}, y_{\max}]$	$[0, 3000]$ m
α	3	$z_{\max}$	90 m

The settings of some common parameters in simulations are listed in Table 1, and other settings related to each scenario can be seen in the corresponding section.

Results of cooperative path planning for persistent surveillance task

Five drones are assigned to perform the persistent surveillance task, and their initial positions are (1470, 1470, 0) m, (90, 90, 0) m, (2910, 90, 0) m, (90, 2910, 0) m and (2910, 2910, 0) m respectively, which are distributed at the center and the four corners of the city map. With the above settings, the results of persistent surveillance when each grid has been visited at least once are shown in Fig. 10.

In Fig. 10, the five drones all take off from the ground and vertically ascend to the maximum flight height to cover greater number of grids. At the initial phase of the task, the drones prefer to fly along the diagonals of the cube because most grids have not ever been visited, and the diagonal path segment can maximize the index value $J(s)$. As the persistent surveillance task goes on, more grids have been visited, and their state values sv begin to decrease. The drones also change their flight directions more frequently to cover the grids with greater state values. Note that the drones are not always staying at the maximum flight height, they sometimes

Fig. 10 Paths of five drones when performing the persistent surveillance task

Fig. 11 Visit times of each grid and the average value

decrease the flight height to cover the grids with special requirements on the shooting resolution. It cost 3886.2 s to realize a complete coverage of all the grids. In the whole process, the drones can bypass the buildings and keep a certain distance with each other to ensure flight safety.

Considering that the information of the number of covered grids and the state value of the grid are both included in the index $J(s)$, the visit times and the state value of each grid at the current moment are shown in Figs. 11 and 12 respectively.

In Fig. 11, each grid has been visited 17.06 times on average, and the standard deviation is 7.41. In general, the grids distributed near the boundaries of the city map are visited by fewer times because when they are involved in the coverage area, the corresponding index value is smaller than the

Fig. 12 State value of each grid and the average value

coverage area containing more inside grids. In Fig. 12, the average state value of the grids is 0.2422 and the standard deviation is 0.21. When the grid has been visited recently, it has a lower probability to be visited again at short notice.

It can be learned from Figs. 11 and 12 that the data is not equally distributed equally for all the grids, and the reasons can be concluded as follows. First, in the established model, more than one grid can be visited when a drone is located at a path point, which is different from the mode that one location of drone just corresponds to one grid. Under this visit mode, there are some overlapping coverage areas for two neighboring path points, which leads to the fact that some grids are visited many times. Second, in the persistent

surveillance task, the goal is not to realize an equal number of visit times, and both the coverage area and the visit time interval are considered when selecting the path point. Third, there are many buildings distributed in urban environments, which makes the coverage area corresponding to a path point irregular, and the drone may make turn frequently to find the optimal path point. At last, the cooperation mechanism among drones requires the drone to fly with the same type of path segment in the same step, which impairs the optimality of path for the individual drone. To sum up, with the proposed MCCPP algorithm, the drones can complete the persistent surveillance task in urban environments considering the special requirement on shooting resolution in some areas.

To further explore the influence of the urban space size and the density of buildings, more simulations are conducted. In the first group of simulations, the size of urban space is varying, and the density of buildings is the variable in the second group. In the two groups of simulations, other settings are the same as those described at the beginning of this section. The comparative results are summarized in Tables 2 and 3.

In Table 2, the number of buildings is also different to keep the density of buildings approximately the same with the increasing of urban space size. The elapsed time of finishing a complete coverage raises with a larger urban space size, but the increment is not linear as the distribution of buildings is a random factor. The gap among the average visit times of each grid in different urban space sizes is not obvious, and a smaller urban space size can result in a larger average visit times of each grid. However, the average state

Table 2 Results under different sizes of urban space

Range along axis OX and OY	Number of buildings	Elapsed time of a complete coverage	Average visit times of each grid	Average state value of each grid
[0, 1000] m	110	1304.7 s	19.72	0.1827
[0, 1500] m	250	2051.4 s	19.18	0.1903
[0, 2000] m	450	2513.8 s	18.42	0.2183
[0, 2500] m	700	3178.5 s	17.59	0.2305
[0, 3000] m	1000	3886.2 s	17.06	0.2422

Table 3 Results under different densities of buildings

Range along axis OX and OY	Number of buildings	Elapsed time of a complete coverage	Average visit times of each grid	Average state value of each grid
[0, 3000] m	500	3372.5 s	15.21	0.2273
	750	3494.6 s	15.83	0.2361
	1000	3886.2 s	17.06	0.2422
	1250	3719.4 s	16.17	0.2387
	1500	3774.1 s	16.48	0.2396

value of the grid shows the opposite trend with the increasing urban space size, and the grids are visited more frequently in a smaller urban space size.

In Table 3, there is no significant regularity with the changing densities of buildings. This is because the influence of buildings is uncertain. Sometimes, more buildings mean that fewer grids are required to be visited when realizing a complete coverage, thus reducing the workload of drones and can lead to a shorter elapsed time. On the contrary, the blocking of buildings may make some grids more difficult to be visited, which will result in a longer elapsed time. According to the data shown in Table 3, a higher density of buildings can make a longer elapsed time of finishing a complete coverage, but when the density of buildings exceeds a certain value, the above conclusion may not be true as the number and the distribution of buildings have complex effects.

Results of emergency path planning in persistent surveillance task

In Fig. 10, all drones are under normal circumstances. When an emergency occurs in a specified area, one of them must be ready to reach the specified area and finish a complete coverage of it. With the emergency path planning algorithm, the path of the drone after receiving the command is shown in Fig. 13.

The starting point refers to the position of the drone when it receives the command to fly to the specified area. The drone does not fly to the specified area directly but makes a detour to increase the coverage area while satisfying the constraint of the latest arrival time. After the drone reaches the target, it continues to fly to bypass the path points on the Hamilton cycle one by one and finishes a complete coverage of the specified area finally. Figure 14 is given to show the

Fig. 13 Path of the special drone when performing the emergency task

Fig. 14 Paths of drones in the whole process of dealing with the emergency

paths of all drones from the moment when the emergency occurs. The ending moment is set as a moment after the special drone has finished a complete coverage of the specified area.

As marked in Fig. 14, drone No.3 is assigned to fly to the specified area, and it can return to perform the persistent surveillance task normally after finishing a complete coverage of the emergency area. The other drones keep working as usual and cooperate with the special drone to maximize the index value in each step. The above results demonstrate that the proposed emergency path planning algorithm can generate an efficient path for the special drone. The path makes the special drone reach the specified area on time and can achieve complete coverage of it. Besides, the MCCPP algorithm still works effectively in each step to ensure the normal operation of the persistent surveillance task after the emergency path planning algorithm has determined the next path point for the special drone.

To demonstrate the superiority of the emergency path planning algorithm with a continuous checking strategy, a comparison between the proposed algorithm and a control group is made. The special drone is required to reach the specified area within 42 s according to Eq. (12). In the control group, when an emergency occurs in a certain area, the special drone flies to the specified area immediately with the shortest flight path to reach there as soon as possible. In other words, the continuous checking strategy is not applied in the control group. With the above conditions, important data in the two simulations are presented in Table 4.

In Table 4, the special drone can reach the specified area less than 42 s in both cases, which can satisfy the constraint in Eq. (12). The special drone can reach the specified area

Table 4 Results of emergency path planning by the proposed algorithm and in the control group

Algorithm	Length of flight path	Number of path point	Total index	Average index in each step	Average state value of each grid
Continuous check	398.78 m	12	70.6608	5.888	0.2103
Control group	133.92 m	4	12.468	3.117	0.2078

with less flight distance and can reach the area earlier in the control group. As the special drone passes through fewer path points in the control group, the total index corresponding to those path points is smaller, which means that the flight path is inferior in the persistence surveillance task. Besides, with the continuous checking strategy, the average index in each step is larger, and its validity in visiting more grids while satisfying the constraint of reaching the area in time is verified. To sum up, although the special drone can reach the specified area with a shorter flight path, it will sacrifice the efficiency of performing the persistent surveillance task. The emergency path planning algorithm with a continuous checking strategy can make the special drone reach the specified area in time while maximizing the index value in Eq. (9).

Comparison between the MCCPP algorithm and the partitioning-based algorithm

In many cooperative coverage path planning problems, the whole area is divided into a certain number of pieces, and each drone is responsible for performing the coverage task in one piece of area. In this group of simulations, four drones join to perform the persistent surveillance task, and they are initially located at the four corners of the city map. In the first simulation, the proposed MCCPP algorithm is applied to generate the paths for four drones. Then, the square ground area is divided into four equal smaller square areas, and each drone takes charge of the persistent surveillance task in one sub-area, as the strategy shown in Ref. [28]. The MCCPP algorithm is also adopted to generate the path for each drone in this case. According to the above descriptions, the paths of drones generated by the two algorithms for a complete coverage of all grids are shown in Figs. 15 and 16 respectively.

In Figs. 15 and 16, the four drones can perform the persistent surveillance successfully with the two algorithms, and it costs 5791.5 s and 19,385.5 s for the MCCPP algorithm and the partitioning-based algorithm respectively to finish a complete coverage of all grids. Compared to the result in the first simulation, the elapsed time of finishing a complete coverage can be reduced by employing more drones. Besides, the consuming time increases a lot when the partitioning-based algorithm is applied. In the partitioning-based

Fig. 15 Paths of four drones generated by the MCCPP algorithm when a complete coverage is finished

Fig. 16 Paths of four drones generated by the partitioning-based algorithm for one complete coverage

algorithm, the drones are restricted to fly within a certain area, and they rarely fly to the points where are near the

Table 5 Comparison between the MCCPP algorithm and the partitioning-based algorithm

Case	Item	Mean value	Standard deviation
M-C	SV	0.17	0.14
	VT	21.06	9.20
M-T5000	SV	0.20	0.15
	VT	17.97	8.22
P-C	SV	0.06	0.05
	VT	71.31	25.94
P-T5000	SV	0.21	0.17
	VT	18.43	9.02

boundaries because those path points often correspond to a smaller index value. To further compare the results with the two algorithms, some important data are summarized in Table 5.

In Table 2, the cases ‘M-C’ and ‘P-C’ mean using the MCCPP algorithm or partitioning-based algorithm to finish a complete coverage. ‘M-T5000’ and ‘P-T5000’ refer to the cases when the persistent surveillance task proceeds to 5000 s with the MCCPP algorithm and partitioning-based algorithm respectively. The items ‘SV’ and ‘VT’ denote the state value and the visit times of the grid respectively. When a complete coverage is finished by the two algorithms, the average state value of grids in the MCCPP algorithm is larger than that in the partitioning-based algorithm, which demonstrates that the grids are visited with higher frequency in the MCCPP algorithm. On the other hand, the average visit times of grids in the partitioning-based algorithm is more than that in the MCCPP algorithm. This is because it costs more time to finish a complete coverage with the partitioning-based algorithm, and naturally, the grids are visited by the drones with more times.

As for the cases when the persistent surveillance task proceeds to 5000 s with the MCCPP algorithm and partitioning-based algorithm, both a smaller average state value and average visit times of grids are obtained by the MCCPP algorithm, and the reasons can be explained as follows. When $t = 5000$ s, the whole area has been covered to a large proportion with the MCCPP algorithm, while a large amount of time is still needed to finish a complete coverage for the partitioning-based algorithm. Compared to the partitioning-based algorithm, the drones spend more time covering the grids which have special requirements on the shooting resolution in the MCCPP algorithm. In the above process, the drones have to reduce their flight heights, which will also reduce the coverage area.

Conclusion

The cooperative path planning problem of multiple drones for persistent surveillance is studied in this paper. Literature investigation shows that the path planning problem for complete coverage has been widely focused on, but the persistent surveillance task is rarely considered. A comprehensive model describing the characteristic of the cooperative persistent surveillance task for multiple drones in an urban environment is lacked, and a cooperative path planning algorithm which can deal with the emergency needs to be developed.

First, the urban space is divided into a number of cubes, and flight rules of drones are defined. Then the model of drone’s coverage area for a path point is established considering the blocking of buildings. When multiple drones join the task, the formula of checking the collision between drones are designed, and the overlapping area is excluded when calculating the combinational coverage area of multiple drones. A state value denoting the visit time interval is defined in each grid, and the index of evaluating the spare path point is to maximize the sum of the state value of the covered grids. To solve the problem, a MCCPP algorithm is proposed to design the strategies of calculating the index value, and the grid which has not been visited for a certain period or has a special requirement on the shooting resolution are considered. The drones are required to fly the same type of path segment in the same step to realize the cooperation.

When an emergency occurs in a certain area, a special drone is assigned to reach there within a certain time and finish a complete coverage. An emergency path planning algorithm is designed to maximize the coverage area while satisfying the constraint of the latest arrival time to the area. A Hamilton cycle is set to realize a complete coverage of the area regardless of the different starting points. With the help of the MCCPP algorithm, the cooperation among drones also can be realized when an emergency occurs.

In the simulation studies, the validity and the superiority of the MCCPP algorithm are verified by performing the persistent surveillance task and dealing with the emergency case. Comparison is also made between the MCCPP algorithm and the partition-based algorithm. In the future, the persistent surveillance task can be applied in other environments, and other variants of the MCCPP algorithm can be explored to solve similar problems.

Funding This research work is financially supported by the Chongqing Research Program of Basic Research and Frontier Technology with the grant number of cstc2020jcyj-msxmX0602, the Fundamental Research Funds for the Central Universities with the project reference number of

2020CDJ-LHZZ-066 and China Scholarship Council with the project reference number of 201906055030.

Compliance with ethical standards

Conflict of interest The authors declare that there is no competing interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Shao J, Cheng J, Xia B, Yang K, Wei H (2020) A novel service system for long-distance drone delivery using the “Ant+ A*” Algorithm. *IEEE Syst J*.
- Liu S, Liu D, Srivastava G, Polap D, Wozniak M (2020) Overview and methods of correlation filter algorithms in object tracking. *Complex Intell Syst* <https://doi.org/10.1007/s40747-020-00161-4>.
- Deng M, Liu B, Li S, Du R, Wu G, Li H, Wang L (2020) A two-phase coordinated planning approach for heterogeneous earth-observation resources to monitor area targets. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.
- Pazooki M, Mazinan AH (2018) Hybrid fuzzy-based sliding-mode control approach, optimized by genetic algorithm for quadrotor unmanned aerial vehicles. *Complex Intell Syst* 4(2):79–93
- Liu H, Li X, Fan M, Wu G, Pedrycz W, Suganthan PN (2020) An autonomous path planning method for unmanned aerial vehicle based on a tangent intersection and target guidance strategy. *IEEE Transactions on Intelligent Transportation Systems*, Early Access
- Wu Y, Low KH (2020) An adaptive path replanning method for coordinated operations of drone in dynamic urban environments. *IEEE Systems Journal*, Early Access
- Arribas E, Mancuso V, Cholvi V (2019) Coverage optimization with a dynamic network of drone relays. *IEEE Trans Mob Comput* 19(10):2278–2298
- Netjasov F, Janic M (2008) A review of research on risk and safety modelling in civil aviation. *J Air Trans Manag* 14(4):213–220
- Yao P, Cai Y, Zhu Q (2019) Time-optimal trajectory generation for aerial coverage of urban building. *Aerosp Sci Technol* 84:387–398
- Andersen, H. L. (2014). Path planning for search and rescue mission using multicopters, *Master's thesis*, Institutt for teknisk kybernetikk.
- Cabreira TM, Di Franco C, Ferreira PR, Buttazzo GC (2018) Energy-aware spiral coverage path planning for UAV photogrammetric applications. *IEEE Robot Automation Lett* 3(4):3662–3668
- Choi Y, Choi Y, Briceno S, Mavris DN (2020) Energy-constrained multi-UAV coverage path planning for an aerial imagery mission using column generation. *J Intell Rob Syst* 97(1):125–139
- Majeed A, Lee S (2019) A new coverage flight path planning algorithm based on footprint sweep fitting for unmanned aerial vehicle navigation in urban environments. *Appl Sci* 9(7):1470
- Xie J, Carrillo LRG, Jin L (2018) An integrated traveling salesman and coverage path planning problem for unmanned aircraft systems. *IEEE Control Syst Lett* 3(1):67–72
- Valente J, Sanz D, Del Cerro J, Barrientos A, de Frutos MÁ (2013) Near-optimal coverage trajectories for image mosaicing using a mini quad-rotor over irregular-shaped fields. *Precision Agric* 14(1):115–132
- Ahmadi SM, Kebriaei H, Moradi H (2018) Constrained coverage path planning: evolutionary and classical approaches. *Robotica* 36(6):904–924
- Song J, Gupta S (2018) e*: an online coverage path planning algorithm. *IEEE Trans Rob* 34(2):526–533
- Khan A, Yanmaz E, Rinner B (2014). Information merging in multi-UAV cooperative search. In: 2014 IEEE international conference on robotics and automation (ICRA) (pp 3122–3129). IEEE.
- Torres M, Pelta DA, Verdegay JL, Torres JC (2016) Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. *Expert Syst Appl* 55:441–451
- Mansouri SS, Kanellakis C, Fresk E, Kominiak D, Nikolakopoulos G (2018) Cooperative coverage path planning for visual inspection. *Control Eng Practice* 74:118–131
- Galceran E, Carreras M (2013) A survey on coverage path planning for robotics. *Robot Autonomous Syst* 61(12):1258–1276
- Cabreira TM, Brisolara LB, Ferreira PR Jr (2019) Survey on coverage path planning with unmanned aerial vehicles. *Drones* 3(1):4
- Guastella DC, Cantelli L, Giammello G, Melita CD, Spatino G, Muscato G (2019) Complete coverage path planning for aerial vehicle flocks deployed in outdoor environments. *Comput Electr Eng* 75:189–201
- Bircher A, Kamel M, Alexis K, Burri M, Oettershagen P, Omari S, Siegwart R (2016) Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots. *Autonomous Robots* 40(6):1059–1078
- Manoussakis Y (1992) A linear-time algorithm for finding Hamiltonian cycles in tournaments. *Discrete Applied Mathematics* 36(2):199–201
- Deogun JS, Steiner G (1994) Polynomial algorithms for Hamiltonian cycle in cocomparability graphs. *SIAM J Comput* 23(3):520–552
- Sárközy GN (2009) A fast parallel algorithm for finding Hamiltonian cycles in dense graphs. *Discrete Math* 309(6):1611–1622
- Seyedi S, Yazicioglu Y, Aksaray D (2019) Persistent surveillance with energy-constrained UAVs and mobile charging stations. *IFAC-PaperOnLine* 52(20):193–198

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.