sensors

MDPI

Article

Reactive Autonomous Navigation of UAVs for Dynamic Sensing Coverage of Mobile Ground Targets

Hailong Huang, Andrey V. Savkin * and Xiaohui Li

School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney 2052, Australia; hailong.huang@unsw.edu.au (H.H.); xiaohui.li@unsw.edu.au (X.L.)
* Corresponding author: a.savkin@unsw.edu.au

Received: 4 June 2020; Accepted: 1 July 2020; Published: 3 July 2020

Abstract: This paper addresses a problem of autonomous navigation of unmanned aerial vehicles (UAVs) for the surveillance of multiple moving ground targets. The ground can be flat or uneven. A reactive real-time sliding mode control algorithm is proposed that navigates a team of communicating UAVs, equipped with ground-facing video cameras, towards moving targets to increase some measure of sensing coverage of the targets by the UAVs. Moreover, the Voronoi partitioning technique is adopted to reduce the movement range of the UAVs and decrease the revisit times of the targets. Extensive computer simulations, from the simple case with one UAV and multiple targets to the complex case with multiple UAVs and multiple targets, are conducted to demonstrate the performance of the developed autonomous navigation algorithm. The scenarios where the terrain is uneven are also considered. As shown in the simulation results, although the additional VP technique leads to some extra computation burden, the usage of the VP technique considerably reduces the target revisit time compared to the algorithm without this technique.

Keywords: unmanned aerial vehicles; navigation; networks of drones; reactive deployment; surveillance and monitoring; sensing coverage; dynamic coverage; autonomous systems

1. Introduction

Unmanned aerial vehicles (UAVs), which have been employed in various military and civilian domains [1], have become a popular tool for the surveillance of targets [2,3]. Two main reasons for the popularity of UAVs in surveillance and monitoring applications are the manufacturing cost reductions and mobility improvement due to the high-performance control of UAVs [4,5]. Considering the inefficiency of using a single large UAV, especially in large areas, it is a trend to employ multiple small UAVs to complete missions quickly [6]. For example, for the periodical area sensing mission, much time can be saved by well designing complete coverage paths for a team of UAVs that operate in a parallel manner [7,8]. Another situation is to monitor some fast changing environment such as bushfires [9], and deploying a UAV team can generally collect more temporal-spatial data than using a single UAV.

In this paper, we focus on the sensing coverage of a number of mobile targets in a region by a UAV team. This problem has several potential applications. From the sense of wireless sensor networks or Internet of Things, some sensor nodes are regarded as the targets. Consider the case where a set of sensors are deployed on the surface of the sea in the Great Barrier Reef zone to monitor the status of the coral reef system. UAVs are deployed to periodically visit the sensors to collect the sensory data [10]. From the security point, some mobile military units can be the targets of the UAVs. UAVs need to move to them to gather their dynamic status.

Different from the previous work using a sufficient number of UAVs for sensing coverage where the targets in the region of interest can be under surveillance at any time [11,12], we consider the more challenging case with a limited number of UAVs in this paper. Then, the targets cannot be covered all the time, and they need to be covered periodically. This is also more general than the case with a sufficient number of UAVs, especially when the region of interest is much larger than the sensing area of the UAVs. In this case, the revisit time among the targets should be as small as possible, as the longer time a target has not been covered, the higher the uncertainty level will be with the target. The problem of interest is how to organize the motions of the UAVs to accommodate the movements of the targets. Different from many existing approaches that construct the path first and design tracking algorithms, we jointly consider the motion control and the trajectory generation. We start from a simple scenario with a single UAV. We develop a reactive real-time sliding mode control algorithm, which navigates the UAV towards the pursuit target (PT). Here, the PT accounts for not only the distance from the UAV to the target but also the time duration this target has not been under coverage. In the meanwhile, the UAV's dynamics is taken into account. Therefore, this navigation algorithm can control the motion of the UAV and plan the path simultaneously. We then move onto the more complex case with multiple UAVs. Beyond the motion control and path planning, the resource allocation issue rises. Then, there needs a coordination scheme among the UAVs. To cooperate, the UAVs share the measured information with others via wireless communication. Then, all the UAVs follow a designed PT selection rule, so that any PT can only be selected once at any time. Whenever two UAVs are close enough, one of them stops pursuing the selected target and takes actions to avoid collisions. Moreover, we adopt the Voronoi partition (VP) technique to reduce the movement range of each UAV. To evaluate the proposed method, we conduct extensive computer simulations. We start from the simple case with one UAV and multiple targets, and then move onto the complex case with multiple UAVs and multiple targets. The scenarios where the terrain is uneven are also considered. As shown in the simulation results, having more UAVs significantly increases the frequency of surveying the targets. Furthermore, although the additional VP technique leads to some extra computation burden, the usage of the VP technique considerably reduces the target revisit time compared to the algorithm without this technique. Moreover, the UAV linear speed has an important impact on the target revisit time, while the influence of the angular speed is small.

The main contribution is the developed autonomous navigation algorithm that can guide a group of UAVs to reactively monitor a set of moving targets. This algorithm is computationally efficient and can be implemented easily in real time. Extensive computer simulations are presented to show the effectiveness of the proposed method.

The remainder of the paper is organized as follows. Section 2 discusses the related work. Section 3 first presents the dynamics of the UAVs and sensing coverage model. Then, the problem under consideration is stated. Section 4 presents the proposed UAV navigation approaches based on a sliding mode control algorithm. Illustrative examples and computer simulations are given in Section 5 to demonstrate the effectiveness of the proposed approaches. Finally, Section 6 presents a brief conclusion.

2. Related Work

The problem of target surveillance is not a new problem, and it has been considered from different levels. For the low level of sensing, many data fusion methods have been developed to integrate the observations from sensors to obtain a better estimation of the target. Approaches based on vision sensors have attracted much attention in recent years. The authors of [13,14] propose methods to detect and localize a target using images taken by a UAV. The authors of [15,16] further propose fast path planning schemes for a UAV to track a target based on estimated location and distance between the target and the UAV. Beyond using a single UAV, researchers have carried out investigations on using multiple UAVs to cover a single target [17–19]. The deployment of multiple UAVs enables the observation of the target from different directions and also reduces the probability of losing the target especially in urban areas where occlusion often exists due to the tall buildings.

In the aforementioned approaches, the focus is on the quality of the target detection. When multiple targets are to be monitored, another problem rises, i.e., how to allocate the UAV resource. The resource allocation significantly affects the quality and efficiency of sensing coverage. The stationary target surveillance has been studied in varying applications, and the targets can be military units, buildings and sensor nodes. Many existing approaches have been employed to address the path planning issue in these applications. When there is only one UAV, the problem can be formulated as the conventional travelling salesman problem (TSP) with the objective of minimizing the time to complete the survey of all the stationary targets [20]. Moreover, some variants of TSP have also been used to address the path planning problem such as TSP with neighbourhoods [21]. Specifically, each target can be surveyed from a certain distance. For example, a UAV can collect the sensory data from a sensor node from a distance. Then, the constructed path does not necessarily pass the targets. Thus, the path length can be shortened. When multiple UAVs are available and the targets spread in the field, the vehicle routing problem (VRP) is a common tool to formulate the path planning problem [22]. In the case of having enough UAVs, an interesting problem is where to deploy them so that an optimal sensing coverage of the targets can be achieved [11,12]. When the targets move in the considered region, some reactive algorithms are needed to maintain the sensing quality [2,23].

The sensing coverage of moving targets can be regarded as a generalization of TSP in the single UAV case. However, the approaches to the original TSP cannot be used to generate the path for the UAV because following a pre-computed path cannot guarantee the successful visitation of all the moving targets. Then, the path has to be recomputed once the targets move. Some general optimization algorithms, such as genetic algorithms [24], have been exploited to reconstruct the path online. It is easy to understand that executing this kind of optimization algorithms requires excellent computing ability. The work in [25] considers the problem of surveying multiple targets by multiple UAVs. The authors assume that the UAVs move much faster than the targets so that when a UAV arrives at a target's last measured location, the target can be covered by the UAV. Thus, their approach still addresses the stationary case with the movements of the targets viewed as perturbations. The work in [26] investigates the problem of covering a group of moving targets using the minimal number of UAVs. The authors focus on two problems: (1) covering all the targets for the entire duration of mission and (2) covering each target at least once during the duration of mission, and they use the generalized version of the minimum flow problem to address these problems. The authors of [27] present a strategy for a one-off mission planning for a UAV team, where n UAVs are required to move to n destinations, e.g., to cover some targets are the destinations. Moreover, A* was adopted to construct the path from a UAV to any target destination, and then the optimal combination of paths is figured out by evaluating each possible combination.

In this paper, the VP technique is adopted. This technique is widely used to partition a plane into regions close to each of a given set of objects. Different domains using VP and its variants include, but not limited to sensor coverage, area search with multiple robots and complete area coverage using multiple robots. The authors of [28] propose a VP-based method to determine the optimal deployment of a set of UAV base stations to serve ground cellular users. The authors of [29] use this technique to find the minimum number of charging stations and their locations for drone parcel delivery. Different from these references, the current paper adopts this technique to group a set of moving targets, in order to improve reduce the target revisit time, which is equivalent to improving the sensing coverage quality.

3. System Model and Problem Statement

Consider a surveillance system consisting of N UAVs la belled $i = 1, 2, \dots, N$. The UAVs fly in a plane that is parallel to the ground. The altitude of this plane is Z, and it is within the allowed deployment range. Let $p_i(t) \in \mathbf{R}^2$ denote the location of the projection of UAV i ($i = 1, \dots, N$) on the ground at time t and $h_i(t) \in \mathbf{R}^2$ ($\|h_i(t)\| = 1$) denote the heading of the UAV. The motion of UAV i is described by

$$\dot{p}_i(t) = v_i(t)h_i(t), \tag{1}$$

$$\dot{h}_i(t) = u_i(t) \tag{2}$$

where $v_i(t) \in \mathbf{R}$ is the linear speed and $u_i(t) \in \mathbf{R}^2$ is to adjust the UAV's heading. Both $v_i(t)$ and $u_i(t)$ are the control inputs. The following constraints hold,

$$\|u_i(t)\| \leq U_{max}, \quad v_i(t) \in [-V_{max}, V_{max}], \tag{3}$$

$$(h_i(t), u_i(t)) = 0 \tag{4}$$

for all t. Here, $\| \cdot \|$ denotes the standard Euclidean vector norm, $(\cdot, \cdot)$ denotes the scalar product of two vectors, and U_{max} and V_{max} are the given constants. The condition (4) guarantees that the vectors $h_i(t)$ and $u_i(t)$ are always orthogonal. The kinematics of many UAVs can be described by non-holonomic models (1)–(4), see, e.g., in [30] and the references therein (Note that in this dynamic model, the linear speed of a UAV can be negative. This differs from many other publications where the linear speed can only take positive values; see [31–33]. The advantage is that when the UAV needs to turn the moving direction for more than 90 degrees, it does not need to make a circle with the maximum angular velocity. Instead it can apply a negative speed.). The main notations used throughout the paper and their meanings are listed in Table 1.

Table 1. The main notations used throughout the paper and their meanings.

Notation	Meaning
$p_i(t)$	Position of UAV i at time t
$h_i(t)$	Heading of UAV i at time t
$v_i(t)$	Linear speed of UAV i at time t
$u_i(t)$	Angular velocity of UAV i at time t
α	Observation angle
Z	Flight altitude
R	Radius of the vision cone of UAVs
$q_i(t)$	Position of target j at time t
$\hat{q}_i(t)$	Estimated position of target j at time t
δ	Sampling time
ϵ	Maximum error of the estimated target speed
τ_j	Revisit time of target j
$\beta_j(t)$	Uncertainty level of target j at time t
$\gamma_j(t)$	Weighted uncertainty level of target j at time t
$d_{ij}(t)$	Distance between UAV i and target j at time t

The goal of this system is to survey targets moving on the ground. Thus, each UAV is equipped with a ground-facing camera that has a given observation angle $\alpha \in (0, \pi)$; see Figure 1. Each UAV, say i, can see a circular area on the ground centred at $p_i(t)$ with the radius of

$$R := Z \tan\left(\frac{\alpha}{2}\right). \tag{5}$$

It is clear that in (5), the altitude Z and the angle α determine the vision area. As mentioned above, the altitude Z must be within an allowed range which is regulated by the local government for the safety of other flying objects or some infrastructure such as power lines. The angle α is determined by the camera in use, and different types of cameras result in varying values of α.

There is a set of M targets to be surveyed moving in a given region $\mathcal{Q} \subset \mathbf{R}^2$ on the ground. Throughout this paper, the terms of survey, cover and visit are used interchangeably. Let $q_j(t) \in \mathcal{Q}$ denote the Cartesian coordinates of target j ($j = 1, \ldots, M$) on the ground at time t. Each target moves on a pre-defined trajectory with an approximately constant speed along its trajectory. This assumption holds in some practical scenarios. For example, a car usually drives with a roughly constant speed on the highway. Even this assumption does not hold, some more advanced estimation and prediction algorithms can be used to estimate and predict the targets' locations. We also assume that when a target is under coverage by a UAV, its location and speed can be measured by the UAV via some image processing techniques. When it is not under coverage by any UAVs, the measured location and speed are used to predict its current location, which is denoted by $\hat{q}_j(t)$. Let ϵ denote the maximum error of the measured speed of the targets and δ denote the sampling time. Then, the single-step prediction error of a target's location is upper bounded by $\delta\epsilon$.

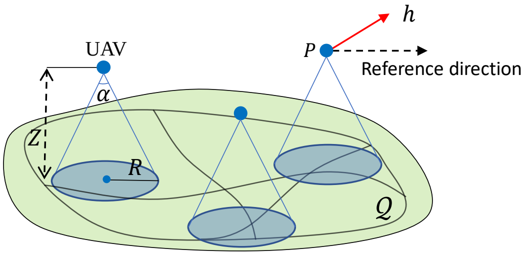


Figure 1. Illustration of the surveillance system.

When $N \geq M$, each target can be monitored by a UAV at any time. When $N < M$, there will be some time during which a target is not under coverage (Note that we consider the case where the M targets are spread in $\mathcal{Q}$. Having multiple targets under coverage by one UAV is an abnormal situation, and we assume that if there is such a situation, it will not last long.). The longer time a target has not been under coverage, the higher the uncertainty level is with this target. Thus, when the resource is limited, an important feature of the surveillance system is to maintain the uncertainty of the targets at the lowest level. This can be stated as an optimization problem that minimizes the maximum revisit time among the targets. Let τ_j denote the time gap between two consecutive visits of target j, which depends on the motion of the UAVs, and let τ denote the maximum of such time gap among the targets, i.e.,

$$\tau := \max_{j=1,\ldots,M} \tau_j. \tag{6}$$

With the single-step location prediction error bound $\delta\epsilon$, the maximum location prediction error for a time period τ is bounded as follows,

$$\max_{j=1,\ldots,M} \|q_j(t) - \hat{q}_j(t)\| \leq \delta\epsilon\frac{\tau}{\delta} = \epsilon\tau. \tag{7}$$

Moreover, to guarantee that the surveillance system can accommodate this maximum prediction error, the following condition should be satisfied,

$$\epsilon\tau \leq R. \tag{8}$$

Otherwise, when a UAV flies to the predicted location of a target, the UAV cannot have the target in view. In other words, this target is lost. To survey this target, the UAV needs to take some extra operations to search the target.

Problem Statement: for the given constants Z, α, ϵ, U_{max}, and V_{max}, we aim at navigating N UAVs to survey M moving targets in $\mathcal{Q}$, so that all the targets are under coverage periodically.

4. Methodology and Main Results

The question to be answered is how to control the motion of a UAV team so that the UAVs can periodically survey the moving targets.

4.1. Methodology

In this section, we address the considered problem following the below methodology, which is shown in Figure 2.

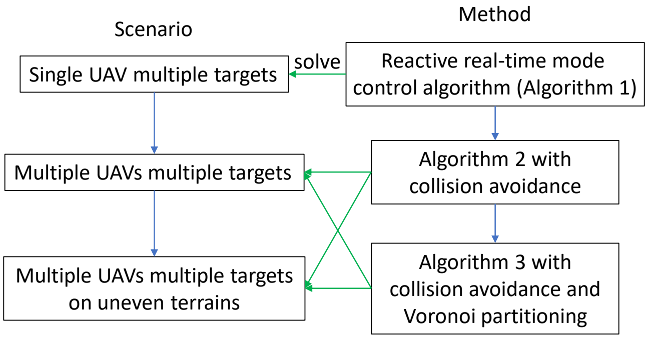

Scenario
Method
Single UAV multiple targets
solve
Reactive real-time mode control algorithm (Algorithm 1)
Multiple UAVs multiple targets
Algorithm 2 with collision avoidance
Multiple UAVs multiple targets on uneven terrains
Algorithm 3 with collision avoidance and Voronoi partitioning

Figure 2. Methodology.

We start from a simple case with one UAV to discuss how to control the motion of the UAV and analyze the number of targets it can survey (Section 4.2). A reactive real-time sliding mode control algorithm is proposed that navigates the UAV to periodically survey a group of targets. As the targets are mobile, following a predefined trajectory does not guarantee the successful visit of all the targets. Thus, we adopt propose a metric to characterize which target should be surveyed next. This metric combines the distance information between the UAV and a target and how long the target has not been surveyed. The target with the highest value of this metric is regarded as the pursuit target (PT). At any time, the UAV moves towards the PT.

With this algorithm, we furthermore address the more complex case with multiple UAVs. As the collision issue arises when multiple UAVs operate simultaneously, the UAVs survey targets with a coordinating scheme to avoid collision. In Section 4.3, the algorithm to navigate a UAV team is presented, and in Section 4.4, the algorithm is improved by adding the Voronoi partitioning technique to reduce the movement range of the UAVs, so that the target revisit time can be shortened. Finally, we present the extension of the proposed method to the cases with targets move on uneven terrains (Section 4.5).

4.2. Navigation Algorithm for a Single UAV

In this subsection, we consider the case of using a single UAV to periodically cover a set of moving targets. Different from many existing methods which determines the high-level visiting order of the targets for the UAV without accounting for the low-level control of the UAV, we consider the visiting target selection and the motion control of the UAV simultaneously. We first present a sliding mode algorithm to control the motion of the UAV.

Let $x_1, x_2 \in \mathbf{R}^2$ be two non-zero vectors, and $\|x_1\| = 1$. Introduce the function $F(\cdot, \cdot)$ mapping from $\mathbf{R}^2 \times \mathbf{R}^2$ to $\mathbf{R}^2$ as

$$F(x_1, x_2) := \begin{cases} 0, & f(x_1, x_2) = 0, \\ \frac{f(x_1,x_2)}{\|f(x_1,x_2)\|}, & f(x_1, x_2) \neq 0, \end{cases} \tag{9}$$

where $f(x_1, x_2) := x_2 - (x_1, x_2)x_1$. The rule (9) is defined in the plane of vectors x_1 and x_2, and the vector $F(x_1, x_2)$ is orthogonal to x_1 and directed "towards" x_2, see Figure 3. It is also clear that $F(x_1, x_2) = 0$ if x_1 and x_2 are co-linear, and $\|F(x_1, x_2)\| = 1$ otherwise. Moreover, introduce the function $g(x_1, x_2)$ as follows,

$$g(x_1, x_2) := \begin{cases} 1, & (x_1, x_2) > 0, \\ -1, & (x_1, x_2) \leq 0. \end{cases} \tag{10}$$

Let $a(t)$ be a vector pointing from the current location of the UAV $p(t)$ to the predicted location of a target $\hat{q}(t)$, i.e., $a(t) := \hat{q}(t) - p(t)$. We use the following sliding mode guidance law to control the motion of the UAV,

$$u(t) = U_{max}g(h(t), a(t))F(h(t), a(t)), \tag{11}$$

$$v(t) = V_{max}g(h(t), a(t)). \tag{12}$$

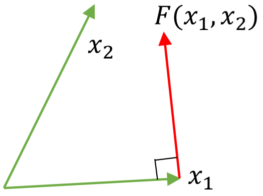


Figure 3. Illustration of the function $F(x_1, x_2)$.

Remark 1. *The navigation law (11) adjusts the heading of the UAV towards the predicted target location with the maximum angular speed, and the law (12) computes the linear speed of the UAV. Note that the linear speed obtained from (12) can be negative, as the angle between the vectors $h(t)$ representing the motion direction of the target and $a(t)$ representing the desired motion direction can be larger than $\frac{\pi}{2}$. The advantage of this algorithm is that the UAV does not need to make a circle to turn back. In other words, the UAV can quickly adjust its motion direction towards the target.*

We are now in a position to use the aforementioned motion control algorithm to deal with the task of surveying multiple moving targets. As the targets can move, following a predefined path does not ensure that all the targets can be surveyed by the UAV. Thus, it is necessary to develop an online trajectory planning method to accommodate the targets' movements. When the targets are stationary, the problem of interest can be converted to the conventional TSP for the single UAV case. TSP aims to find the trajectory for the UAV so that it takes the minimum time to visit all the targets. There are many existing solutions, including exact algorithms and heuristic algorithms. Among them, the exact algorithms take too long to solve even a medium-scale problem. There are some heuristic algorithms, such as the nearest neighbor (NN) algorithm, that can quickly solve the problem as they trade the optimality for speed. The proposed method shares the fundamental idea of the NN algorithm. Differently, the term of NN in this paper depends on not only the distance, but also on the uncertainty level of a target.

The uncertainty level of a target depends on the time the target has not been under coverage since the last visit [34] (In the case of using sensor nodes to monitor the status of the coral reef system in the Great Barrier Reef zone, the longer time a sensor node has not been visited, the more data it stores. If a node is not visited for a long time, the stored data may even overflow.). In this paper, we assume that if a target was visited at time t_1, the uncertainty level maintains a very low level for a period of time T ($T \geq 0$). After that, it increases with time. Let $\beta(t)$ be such a function for all $t \in [t_1, t_2)$. Here, t_2 is the time at which the target is visited for the next time, which is unknown in advance. The function $\beta(t)$ satisfies the following conditions,

$$\begin{aligned} &\beta(t) = 0, \quad t \in [t_1, t_1 + T], \\ &\beta(t) > 0, \dot{\beta}(t) > 0, \quad t \in (t_1 + T, t_2). \end{aligned} \tag{13}$$

When the target is visited by a UAV at time t_2, the uncertainty level of this target drops to 0, maintains 0 for a period of T, and then increases with time again; see Figure 4. An example of the function $\beta(t)$ satisfying (13) is as follows,

$$\beta(t) = \begin{cases} 0, & t \in [t_1, t_1 + T] \\ t - t_1 - T, & t \in (t_1 + T, t_2). \end{cases} \tag{14}$$

In our proposed solution, we use the notation $\beta_j(t)$ to describe the uncertainty level of target j at time t. Introduce a new metric $\gamma_j(t)$ for target j at time t:

$$\gamma_j(t) := \frac{\beta_j(t)}{d_j(t)}, \tag{15}$$

where $d_j(t)$ is the distance between the UAV and the target j on the horizontal plane at time t. This metric couples the information of distance and uncertainty level. The target with the maximum value of $\gamma(t)$ is named as the pursuit target (PT).

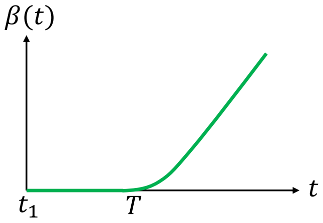


Figure 4. Illustration of the function $\beta(t)$.

The proposed method to reactively navigate a single UAV to survey a set of moving targets is summarized in Algorithm 1. It consists of three main steps: First, the UAV chooses the PT. Then, the UAV computes the control inputs $u(t)$ and $v(t)$ according to (11) and (12), and applies them to make a movement for a sampling time δ. During the movement, the UAV updates the status of the targets that have been in its view. The UAV will choose the next PT when the current PT is covered. It is clear that the UAV can reactively move towards the PT to accommodate the movement of the PT. Therefore, we call this algorithm the online reactive navigation algorithm.

Algorithm 1: Online reactive navigation algorithm for a single UAV.

```
while 1 do
    Select the PT.
    while The PT has not been surveyed do
        Compute the control input $u(t)$ and $v(t)$ according to (11) and (12), and apply $u(t)$ and
          $v(t)$ for a sampling time $\delta$.
        Update the status of all the targets that are seen by the UAV.
    end
end
```

Remark 2. *The proposed online reactive navigation algorithm, i.e., Algorithm 1, is easy to implement on a UAV in real-time. From Algorithm 1 we can see that the UAV only needs to compute two control inputs: $u(t)$ and $v(t)$ according to (11) and (12). Obviously, these simple calculations are computational efficient. Whenever the UAV sees a target, the on-board sensors take the measurements of the target.*

When there is only one UAV, the condition (8) is not guaranteed to hold. In this case, an interesting issue is how many targets in $\mathcal{Q}$ can one UAV survey subject to condition (8). Such a number depends on how the targets spread in $\mathcal{Q}$. When they are relatively close, this number is larger, while when they are relatively far from each other, it is smaller. Clearly, in the former case, the number of targets can be infinite. It is more interesting to analyze the upper bound on the number of targets in the latter case.

Let L be the length of the shortest path that connects two farthest points in $\mathcal{Q}$. Moreover, the curvature at any point on this path is no larger than $\frac{U_{max}}{V_{max}}$. Then, this path can be followed by a UAV modeled by (1)–(4). Let V_T be the maximum speed of targets. We assume that the UAV can move faster than the targets, i.e., $V_T < V_{max}$. With M targets in $\mathcal{Q}$, no matter how they move, the length of the UAV trajectory is bounded by ML. To complete the longest trajectory within the time τ, we have the following inequality:

$$ML \leq V_{max}\tau. \tag{16}$$

Together with (8), we can obtain the upper bound on M:

$$M \leq \frac{RV_{max}}{\epsilon L}. \tag{17}$$

Remark 3. *This analysis shows that when M satisfies (17), these targets can be covered by the UAV. It is also worth pointing out that (17) is a conservative bound. It considers the worst case where any pair of the targets are L away from each other. However, in practice, such a case only exists when $M = 2$, where the two targets are at the two ends of the path having the length of L. The UAV flies back and forth, and the total length of its trajectory is $2L$. When $M > 2$, the total length of the UAV's trajectory is less than ML. Then, when the UAV travels for ML, more targets can be surveyed. We will show this in Section 5.*

4.3. Navigation Algorithm for Multiple UAVs

When multiple UAVs are available, the target revisit time can be significantly reduced. The navigation of each UAV among the team is still based on Algorithm 1, but there should be some coordinating schemes so that the surveying mission can be carried out efficiently by the UAVs. The navigation algorithm for each UAV among the team is summarized in Algorithm 2, and the differences with Algorithm 1 are explained below.

Algorithm 2: Online reactive navigation algorithm for each UAV of the team.

```
while 1 do
    Share information among the UAV team.
    Select the PT.
    while The PT has not been surveyed do
        Compute the control input $u(t)$ and $v(t)$ according to (11) and (12), and apply $u(t)$ and
          $v(t)$ for a sampling time $\delta$.
        Update the status of all the targets that are seen by the UAV.
        Share information among the UAV team.
        If any other UAV is within $d_{safe}$, avoid collision.
    end
end
```

Data exchange: When a target is within the vision cone of a UAV, the target status, such as the ID, speed, and location, can be collected by the UAV. As the targets can move, it is possible for some target which was closer to one UAV to be closer to another UAV in the future. Thus, the UAVs share their own locations and the information about the targets with each other.

PT selection: Different from the single UAV case, the selection of PT for one UAV, say UAV i, needs to take into account not only the uncertainty levels and estimated locations of the targets and the location of UAV i but also the locations of other UAVs, which can be obtained by the data exchange process. The coordinating rule is stated as follows. Every UAV computes a N-by-M matrix Γ with $\gamma_{ij}(t) = \frac{\beta_j(t)}{d_{ij}(t)}$ as elements, where $d_{ij}(t)$ is the distance between UAV i and target j on the horizontal plane at time t. To select the PT, UAV i finds the element in the ith row of the matrix Γ, say $\gamma_{ij}(t)$, such that $\gamma_{ij}(t)$ is the largest element in the jth column. In the case where there is another element in the jth column having the same value as $\gamma_{ij}(t)$, target j is assigned as the PT of the UAV with the smaller index. This case occurs when two UAVs are at the same distance away from target j. Clearly, this coordinating scheme will assign each UAV a unique PT at any time, since $N < M$.

Collision avoidance: Although different UAVs have different PTs, it is still possible that one UAV collides with others during the movement, as the targets can move. There is a great amount of research results on the topic of collision avoidance of mobile robots [35,36]. In this paper, we consider a simple approach. To avoid collision, all the UAVs obey the following rule: whenever two UAVs are within a safe distance d_{safe}, the UAV with the smaller index continues to move towards its PT, while the other UAV takes actions to avoid collision. For example, it can hover at its current position or move away from the former UAV. Thus, it is required that the collision avoidance ability is embedded, which is common in the commercial UAVs. When they are at least d_{safe} away from each other, the latter re-starts to move towards its PT. Clearly, as each UAV has a unique index and a UAV knows the indices of the nearby UAVs via the data exchange, this simple approach can effectively avoid collisions between UAVs.

We now consider the number of UAVs required to survey M targets moving in $\mathcal{Q}$. When there are N UAVs, the total length of their trajectories is upper bounded by $ML + NL$. This can be obtained by breaking the trajectory for one UAV (whose length is ML as mentioned in Section 4.2) and adding N extra subpaths with the length of L. As these trajectories are followed by N UAVs, the following inequality holds,

$$ML + NL \leq NV_{max}\tau. \tag{18}$$

Together with (8), we can obtain the lower bound on N:

$$N \geq \frac{M\epsilon L}{RV_{max} - \epsilon L}. \tag{19}$$

Remark 4. *Similar to the single UAV case, (19) is also a conservative bound.*

4.4. VP-Based Navigation Algorithm for Multiple UAVs

In Section 4.3, we present an algorithm to navigate multiple UAVs. Via computer simulations, we have observed one drawback of the proposed algorithm. Specifically, each UAV may traverse the major area of the region $\mathcal{Q}$. This may result in a long revisit time of each target. To avoid this drawback and reduce the target revisit time, in this subsection, we present a Voronoi partition (VP) based navigation algorithm for multiple UAVs.

One of the most important techniques of this VP-based navigation algorithm is the partition of the region according to the locations of the targets and the UAVs. In particular, a target falls into the Voronoi cell of a UAV if their distance is no larger than the distance between this target and any other UAV. Formally, let $C_i(t)$ denote the Voronoi cell of UAV i. Target j belongs to $C_i(t)$ if the following condition holds, (Note that when the equality of (20) holds, the target j is assigned to the Voronoi cell with the smaller index.)

$$d_{ij}(t) \leq d_{hj}(t), \forall h = 1, \ldots, N, \;\; h \neq i. \tag{20}$$

As mentioned in Section 4.3, the UAVs share the information about the targets and their own locations. With this information, each UAV knows which target is closer to itself than any other UAVs, so that it can construct its Voronoi cell. Then, each target chooses its PT from its own Voronoi cell. Such a VP-based method is summarized in Algorithm 3.

It is easy to understand that the partition of the targets according to (20) assigns each target to a unique Voronoi cell. Thus, at any time, each UAV has its unique PT. Algorithm 3 has one more operation that needs to be executed by each UAV than Algorithm 2, i.e., the determination of the Voronoi cell. For each target, each UAV needs to compute the distance from this target to N UAVs. Then, for M targets, the additional computation complexity is $O(MN)$. As will be shown in the section of simulation results, such an additional computation results in a large percentage of revisit time reduction.

Algorithm 3: VP-based Online reactive navigation algorithm for each UAV of the team.

```
while 1 do
    Share information among the UAV team.
    Determine the Voronoi cell.
    Select the PT within the Voronoi cell.
    while The PT has not been surveyed do
        Compute the control input $u(t)$ and $v(t)$ according to (11) and (12), and apply $u(t)$ and
          $v(t)$ for a sampling time $\delta$.
        Update the status of all the targets that are seen by the UAV.
        Share information among the UAV team.
        If any other UAV is within $d_{safe}$, avoid collision.
    end
end
```

4.5. Extension to Uneven Terrains

So far, we have presented UAV navigation algorithms that enable a UAV team to periodically cover a set of targets that move on a flat ground. In this subsection, we point out that the proposed approaches can also be used in the cases with uneven terrains under some conditions.

Different from the flat ground situation, the notation $\mathcal{Q}$ now represents a region on an uneven terrain. Let Z_{max} be the altitude of the highest point in $\mathcal{Q}$. For the safety purpose, it is necessary deploy the UAVs at an altitude higher than Z_{max}:

$$Z \geq Z_{max} + Z_{safe}, \tag{21}$$

where Z_{safe} denotes the minimum vertical distance the UAVs should be away from the terrain.

Moreover, as the region $\mathcal{Q}$ is now uneven, any target in this region should be with a 3D coordinates rather than the 2D coordinates in the flat ground case. Let α_0 denote the largest elevation angle of the region $\mathcal{Q}$. We consider the following condition,

$$\alpha_0 < \frac{\pi - \alpha}{2}. \tag{22}$$

The motivation behind this condition is to avoid occlusion because of the terrain. An illustration is shown in Figure 5. On the left hand side of Figure 5, the above condition is not satisfied. Then, the target at the position $q_j(t)$ cannot be covered by the UAV at $p_i(t)$. When this condition holds, the part of $\mathcal{Q}$ that is inside the vision cone of a UAV can be covered by the UAV without any occlusion; see the right hand side of Figure 5. Then, when the condition (8) is satisfied, the UAV can cover the target when the UAV reaches the estimated location of the target $\hat{q}(t)$. Readers are referred to our previous work [11] on the topic of static surveillance accounting for occlusion.

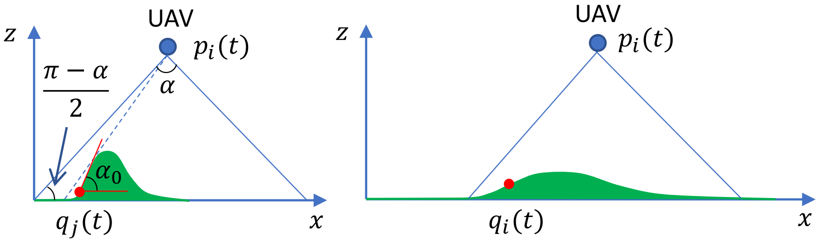


Figure 5. Vision cone with angle α. On the left side, the maximum elevation angle α_0 is larger than $\frac{\pi-\alpha}{2}$. Thus, some part of the terrain inside the vision cone of the UAV is occluded. The right hand side shows that when $\alpha_0 < \frac{\pi-\alpha}{2}$ holds, the terrain inside the vision cone can be seen by the UAV.

Therefore, when the region $\mathcal{Q}$ on the uneven terrain satisfies conditions (21) and (22), the approaches presented above can still be used. It is also worth pointing out that in practice, the condition (21) is usually easy to be satisfied, while the condition (22) may not be satisfied for some quite uneven terrain like the one on the left hand side of Figure 5. When the condition (22) cannot be satisfied, an extra searching operation can be conducted when the UAV reaches the prediction location of the target. The target can be found after the UAV searches the unseen area.

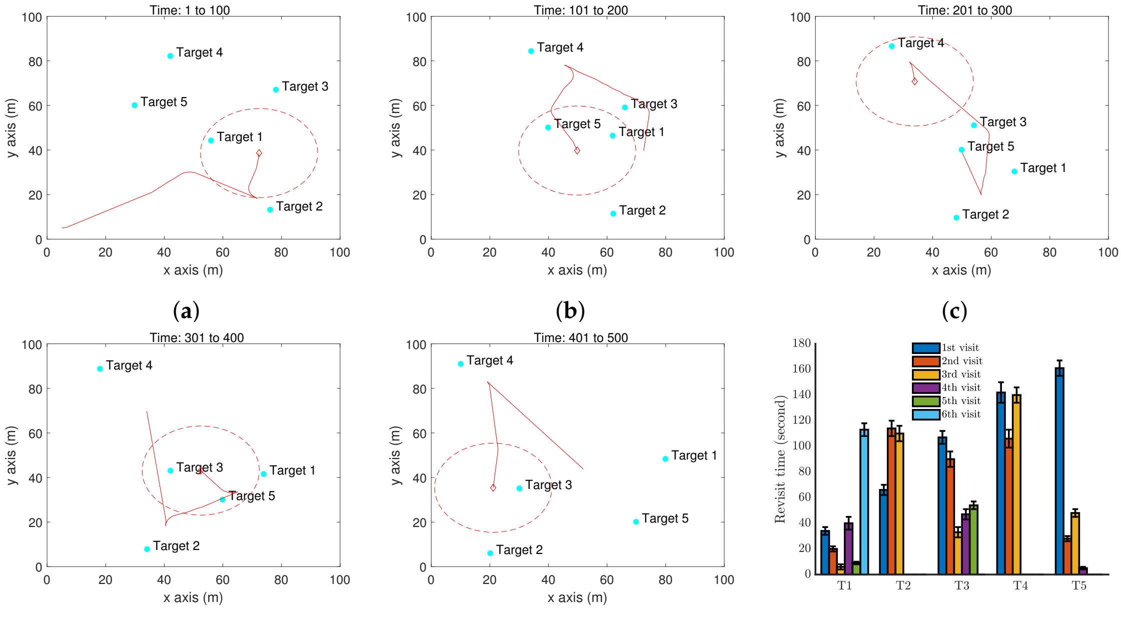
5. Simulations

The performance of the proposed approach is demonstrated by computer simulations carried out with Matlab. We first present one simulation with a single UAV periodically monitoring five moving targets in a square zone (100 m by 100 m) in Figure 6. Here, $Z = 20$ m, $\alpha = \frac{\pi}{2}$, $\epsilon = 0.1$ m/s, $V_{max} = 1$ m/s, and U_{max} = 0.2. The function $\beta(t)$ is set as the example (14) with $T = 10$ s (It is worth pointing out that any other functions of $\beta(t)$ can also be used. Also, the parameters used are only for computer simulations.). The simulate time for this case is 500 seconds. The trajectory of the UAV is shown in Figure 6a–e. A video recording the movements of the targets and the UAV in the simulation period is available at: https://youtu.be/jLyacSXMZl4. Figure 6f shows the distribution of the revisit time of each target in 10 independent simulations, in order to capture the uncertainty in the target location prediction. Target 3 has been surveyed six times, Target 1 and Target 5 have been surveyed five times, and Target 2 and Target 4 have been surveyed three times in 500 seconds. The maximum revisit time among these targets is about 160 seconds. To further demonstrate the performance of the proposed method, a comparison is made with a benchmark method. As we have not seen any existing work focusing on the currently considered problem, we regard a complete coverage path method as the benchmark. The complete coverage path method aims at generating a path that can completely cover the area of interest. This method is widely used in agriculture, search and rescue, exploration, etc. For the above considered setting, a complete coverage path is generated for the UAV; see Figure 7a. Note that this path is generated by straight lines connecting some waypoints. The UAV cannot exactly follow this path. Specifically, at the waypoints, the UAV has to make a smallest circle to turn the moving direction, and this is not shown in Figure 7a. All the targets move in the same way as above, and the target revisit time is shown in Figure 7b. Clearly, following the complete coverage path results in a much larger revisit time than the proposed method. The reason is that following the complete coverage path makes the UAV passive survey the targets.

Now, we present a case with two UAVs to test Algorithm 2. We first consider case with one more step forward from the above case. In other words, all the settings remain the same except that two UAVs are deployed. The results are shown in Figure 8. Comparing with Figure 6f, we can clearly see the advantage of introducing one more UAV: the target revisit is reduced to about 100 seconds.

Moreover, we present a case with two UAVs to test Algorithm 3. The results are shown in Figure 9. Comparing Figures 6f, 8f and 9f, we can clearly see the advantage of the VP technique: the target revisit is further reduced to about 80 seconds.

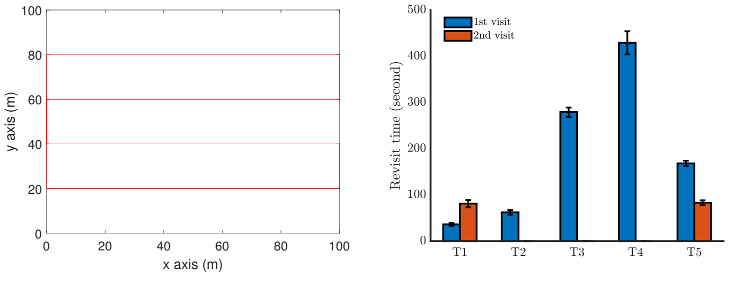
We now demonstrate some simulation results where the targets move on an uneven terrain. We first present a single UAV case with five moving targets in Figure 10. The movements of these targets are similar to those in the first case, see Figure 6. The only difference lies in the uneven terrain. All the system parameters remain the same as the first case. The trajectory of the UAV is shown in Figure 10a–e, and Figure 10f displays the revisit times of the targets. Comparing with Figure 6f, Target 5 has the largest revisit time, which is around 165 seconds. Figure 11 reports the case with two UAVs by applying Algorithm 3.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 6. Simulation results with a single UAV and five moving targets on a flat ground by Algorithm 1. (**a**–**e**) The trajectory of the UAV (red curve) and the positions of targets (cyan dots) at different time instants. The red dash circle is the boundary of the UAV vision cone (https://youtu.be/jLyacSXMZl4). (**f**) Revisit time of each target.



(**a**) (**b**)

Figure 7. Simulation results when the UAV follows a complete coverage path. (**a**) The UAV path. (**b**) Revisit time of each target.

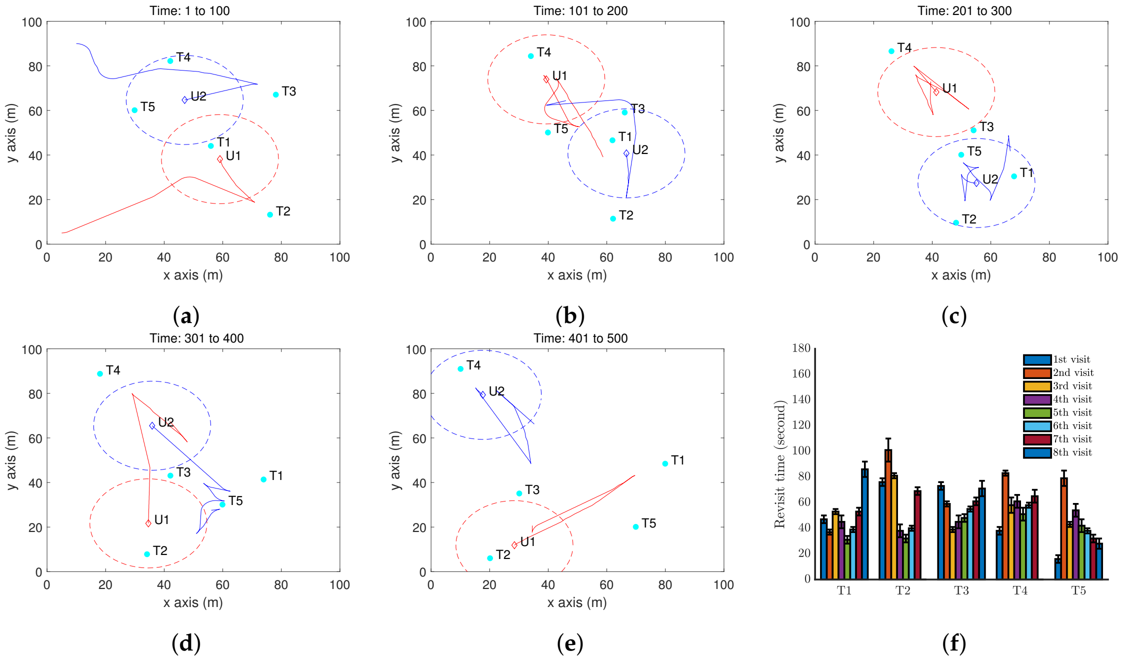


Figure 8. Simulation results with two UAV and five moving targets on a flat ground by Algorithm 2. (**a**–**e**) show the trajectory of the UAV (red curve) and the positions of targets (cyan dots) at different time instants. The red dash circle is the boundary of the UAV vision cone (https://youtu.be/h46R4QvPkQs). (**f**) Revisit time of each target.

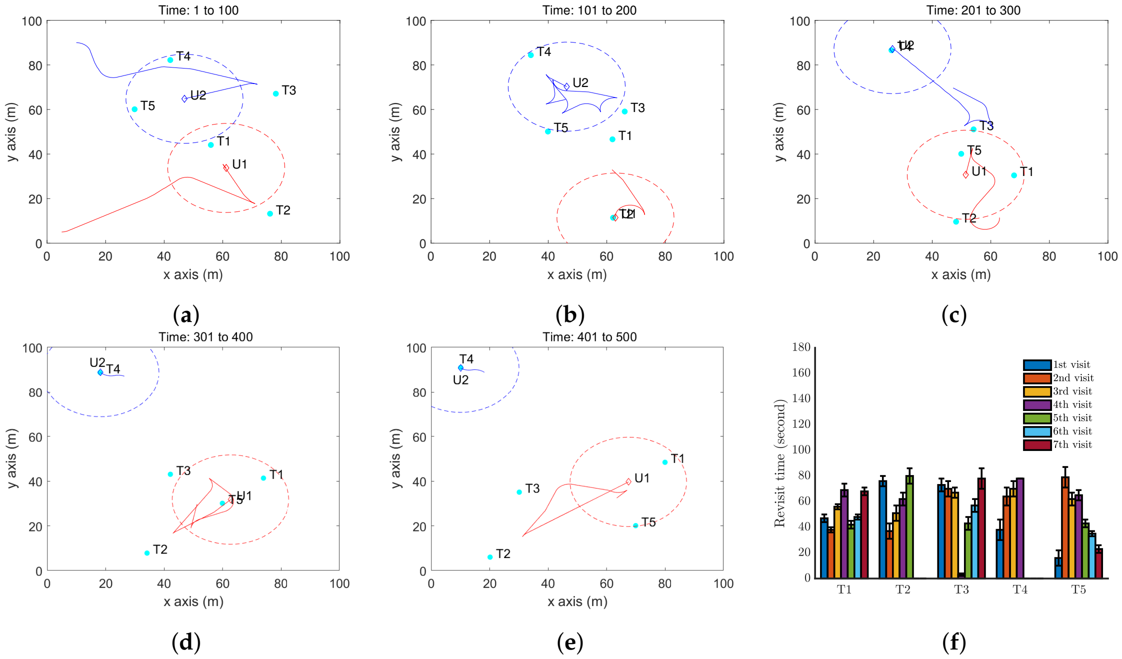
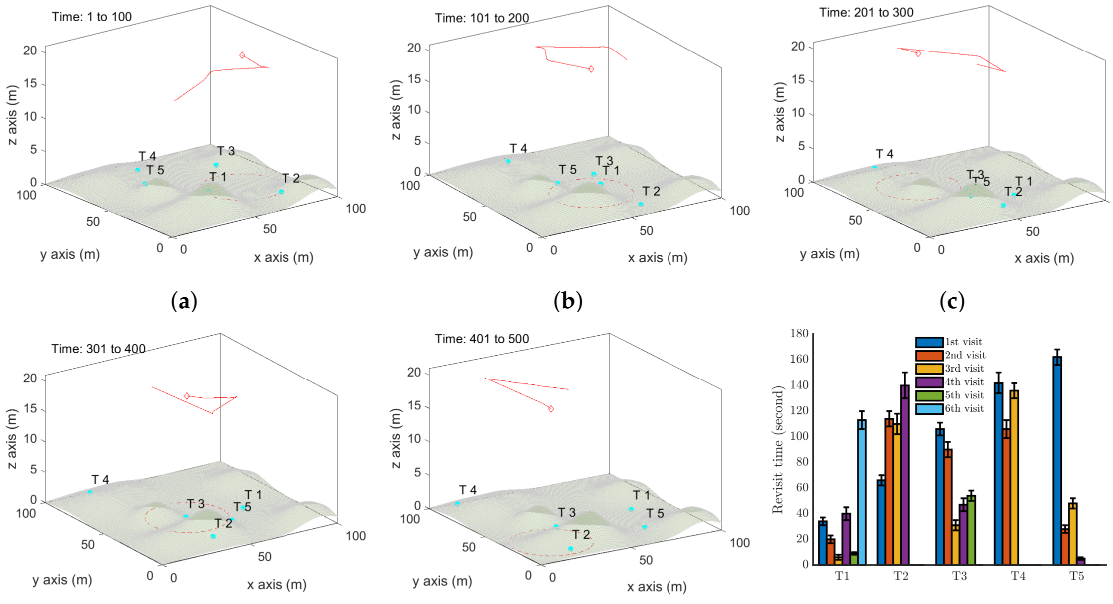


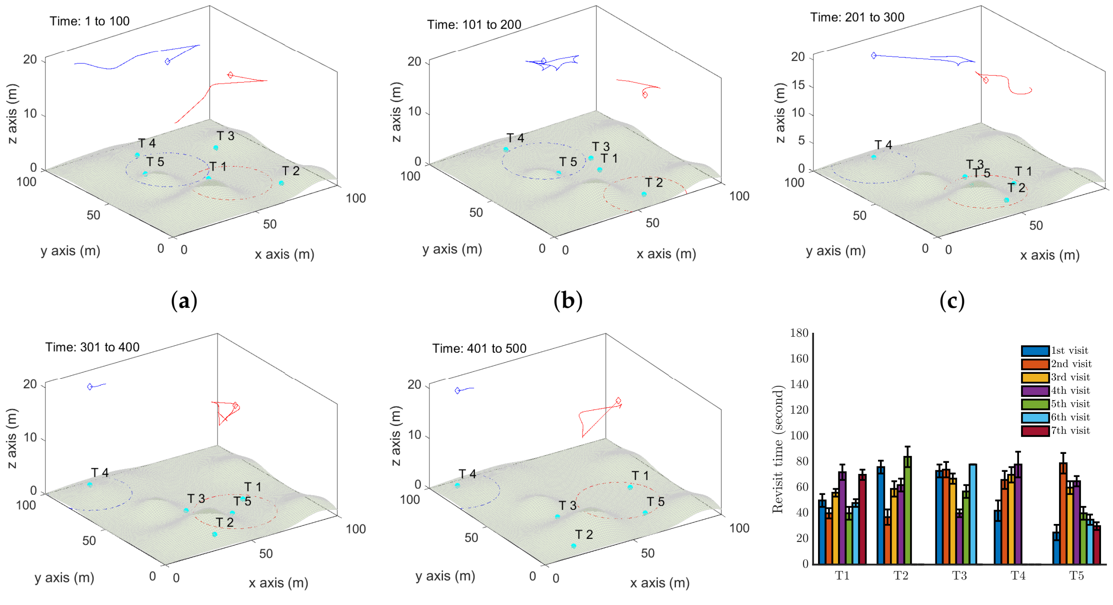
Figure 9. Simulation results with two UAV and five moving targets on a flat ground by Algorithm 3. (**a**–**e**) The trajectory of the UAV (red curve) and the positions of targets (cyan dots) at different time instants. The red dash circle is the boundary of the UAV vision cone (https://youtu.be/8wIfqLDeqHE). (**f**) Revisit time of each target.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 10. Simulation results with a single UAV and five moving targets on an uneven terrain by Algorithm 1. (**a**–**e**) The trajectory of the UAV (red curve) and the positions of targets (cyan dots) at different time instants. The red dash circle is the boundary of the UAV vision cone. (**f**) Revisit time of each target.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 11. Simulation results with two UAVs and five moving targets on an uneven terrain by Algorithm 3. (**a**–**e**) show the trajectories of the UAVs. (**f**) Revisit time of each target.

Finally, we study the impact of a key parameter on the target revisit time: V_{max}, and we consider the maximum revisit time among all the targets. We set V_{max} with various values and conduct more computer simulations on the last case. Specifically, the movements of the targets maintain the same in all the simulations. From Figure 12 we can see that increase the UAVs' maximum linear speed significantly decreases. The reason is clear: the faster the UAVs move, the less time spent on the movements between targets. The maximum angular velocity U_{max} also influences the target revisit time. But its impact is quite slight, because increasing U_{max} only influences the movement for changing the motion direction, while it does not affect the time the UAVs spend on the movements between targets.

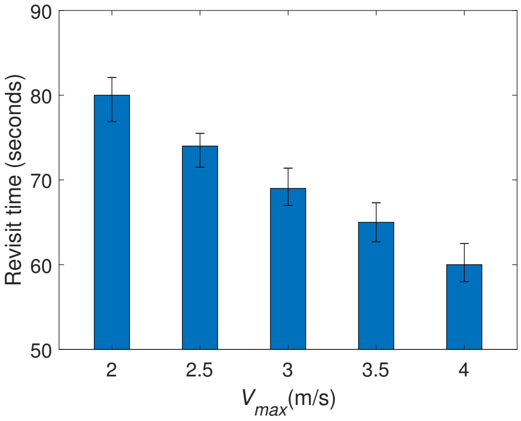


Figure 12. The impact of V_{max} on the target revisit time.

In summary, in this section, we have tested the proposed algorithms via computer simulations. The results show that the proposed online navigation algorithms can periodically survey a set of moving targets in a region. The region is not necessarily flat but can be uneven. The UAV maximum linear speed significantly affects the target revisit time while the maximum angular speed has a slight impact.

6. Conclusions

In this paper, we focused on the application of sensing coverage of moving targets by a UAV team. We presented an online path planning method for a single UAV to periodically cover a set of moving targets on the flat ground or the uneven terrain. The low-level control accounting for the dynamics of the UAV was considered in the path planning method. We further extended this method to navigate multiple UAVs for the sensing coverage of moving targets based on the shared information about targets among UAVs. Via computer simulations, we observed that the extended method may guide the UAVs to traverse the major area of the considered region, which may result in long revisit time. To avoid this, we adopted the Voronoi partition technique to narrow the movement range of each UAV. Simulation results showed that this improved method significantly reduces the revisit times of the moving targets.

One limitation of the current navigation method is based on the assumption that the targets move on some given trajectories. When these trajectories are not known to the UAVs, the target location predictions are with more uncertainty. As a result, the UAVs may lose some targets during the mission. Our future research work is to extend the proposed method to deal with this situation by including the searching operation when any target is lost.

Author Contributions: Conceptualization , A.V.S.; methodology, H.H.; software, H.H.; validation, H.H.; formal analysis, H.H.; investigation, H.H.; resources, H.H.; data curation, X.L.; writing—original draft preparation, H.H.; writing—review and editing, A.V.S., X.L.; visualization, H.H.; supervision, A.V.S.; project administration, X.L.; funding acquisition, A.V.S. All authors have read and agreed to the published version of the manuscript

Funding: This work was supported by the Australian Research Council. Moreover, this work received funding from the Australian Government, via grant AUSMURIB000001 associated with ONR MURI grant N00014-19-1-2571.

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

References

1. Huang, H.; Savkin, A.V. Towards the internet of flying robots: A survey. *Sensors* **2018**, *18*, 4038. [CrossRef]
2. Huang, H.; Savkin, A.V. An algorithm of reactive collision free 3-D deployment of networked unmanned aerial vehicles for surveillance and monitoring. *IEEE Trans. Ind. Inform.* **2020**, *16*, 132–140. [CrossRef]
3. Li, X.; Huang, H.; Savkin, A.V. A novel method for protecting swimmers and surfers from shark attacks using communicating autonomous drones. *IEEE Internet Things J.* **2020**. [CrossRef]
4. Xu, Y.; Zhao, S.; Luo, D.; You, Y. Affine formation maneuver control of high-order multi-agent systems over directed networks. *Automatica* **2020**, *118*, 109004. [CrossRef]
5. Liu, P.; Li, G.; Liu, X.; Xiao, L.; Wang, Y.; Yang, C.; Gui, W. A novel non-uniform control vector parameterization approach with time grid refinement for flight level tracking optimal control problems. *ISA Trans.* **2018**, *73*, 66–78. [CrossRef] [PubMed]
6. Ješke, P.; Klouček, Š.; Saska, M. Autonomous Compact Monitoring of Large Areas Using Micro Aerial Vehicles with Limited Sensory Information and Computational Resources. In *International Conference on Modelling and Simulation for Autonomous Systesm*; Springer: Cham, Switzerland, 2018; pp. 158–171.
7. Guastella, D.C.; Cantelli, L.; Longo, D.; Melita, C.D.; Muscato, G. Coverage path planning for a flock of aerial vehicles to support autonomous rovers through traversability analysis. *ACTA IMEKO* **2019**, *8*, 9–12. [CrossRef]
8. Guastella, D.C.; Cantelli, L.; Giammello, G.; Melita, C.D.; Spatino, G.; Muscato, G. Complete coverage path planning for aerial vehicle flocks deployed in outdoor environments. *Comput. Electr. Eng.* **2019**, *75*, 189–201. [CrossRef]
9. Savkin, A.V.; Huang, H. Navigation of a network of aerial drones for monitoring a frontier of a moving environmental disaster area. *IEEE Syst. J.* **2020**. [CrossRef]
10. Huang, H.; Savkin, A.V.; Ding, M.; Huang, C. Mobile robots in wireless sensor networks: A survey on tasks. *Comput. Netw.* **2019**, *148*, 1–19. [CrossRef]
11. Savkin, A.V.; Huang, H. Proactive deployment of aerial drones for coverage over very uneven terrains: A version of the 3D art gallery problem. *Sensors* **2019**, *19*, 1438. [CrossRef]
12. Savkin, A.V.; Huang, H. Asymptotically optimal deployment of drones for surveillance and monitoring. *Sensors* **2019**, *19*, 2068. [CrossRef]
13. Zhang, L.; Deng, F.; Chen, J.; Bi, Y.; Phang, S.K.; Chen, X.; Chen, B.M. Vision-based target three-dimensional geolocation using unmanned aerial vehicles. *IEEE Trans. Ind. Electron.* **2018**, *65*, 8052–8061. [CrossRef]
14. Wang, L.; Chen, F.; Yin, H. Detecting and tracking vehicles in traffic by unmanned aerial vehicles. *Autom. Constr.* **2016**, *72*, 294–308. [CrossRef]
15. Liu, Y.; Wang, Q.; Hu, H.; He, Y. A novel real-time moving target tracking and path planning system for a quadrotor uav in unknown unstructured outdoor scenes. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2362–2372. [CrossRef]
16. Wang, S.; Jiang, F.; Zhang, B.; Ma, R.; Hao, Q. Development of UAV-based target tracking and recognition systems. *IEEE Trans. Intell. Transp. Syst.* **2019**. [CrossRef]
17. Wise, R.; Rysdyk, R. UAV coordination for autonomous target tracking. In Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Keystone, CO, USA, 21–24 August 2006; p. 6453.
18. Oh, H.; Turchi, D.; Kim, S.; Tsourdos, A.; Pollini, L.; White, B. Coordinated Standoff Tracking Using Path Shaping for Multiple UAVs. *IEEE Trans. Aerosp. Electron. Syst.* **2014**, *50*, 348–363. [CrossRef]
19. Zhang, M.; Liu, H.H. Cooperative tracking a moving target using multiple fixed-wing UAVs. *J. Intell. Robot. Syst.* **2016**, *81*, 505–529. [CrossRef]
20. Tang, H.; Miller-Hooks, E. Solving a generalized traveling salesperson problem with stochastic customers. *Comput. Oper. Res.* **2007**, *34*, 1963–1987. [CrossRef]
21. Dumitrescu, A.; Mitchell, J.S. Approximation algorithms for TSP with neighborhoods in the plane. *J. Algorithms* **2003**, *48*, 135–159. [CrossRef]
22. Toth, P.; Vigo, D. *The Vehicle Routing Problem*; SIAM: Philadelphia, PA, USA, 2002.
23. Huang, H.; Savkin, A.V. Reactive 3D deployment of a flying robotic network for surveillance of mobile targets. *Comput. Netw.* **2019**, *161*, 172–182. [CrossRef]
24. Moraes, R.; Freitas, E. Experimental analysis of heuristic solutions for the moving target traveling salesman problem applied to a moving targets monitoring system. *Expert Syst. Appl.* **2019**, *136*, 392–409. [CrossRef]

25. Tang, Z.; Ozguner, U. Motion planning for multitarget surveillance with mobile sensor agents. *IEEE Trans. Robot.* **2005**, *21*, 898–908. [CrossRef]
26. Das, A.; Shirazipourazad, S.; Hay, D.; Sen, A. Tracking of multiple targets using optimal number of UAVs. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *55*, 1769–1784. [CrossRef]
27. Guastella, D.C.; Cavallaro, N.D.; Melita, C.D.; Savasta, M.; Muscato, G. 3D path planning for UAV swarm missions. In Proceedings of the 2018 2nd International Conference on Mechatronics Systems and Control Engineering, Amsterdam, The Netherlands, 21–23 February 2018; pp. 33–37.
28. Savkin, A.V.; Huang, H. Deployment of unmanned aerial vehicle base stations for optimal quality of coverage. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 321–324. [CrossRef]
29. Huang, H.; Savkin, A.V. A method of optimized deployment of charging stations for drone delivery. *IEEE Trans. Transp. Electrif.* **2020**, *6*, 510–518. [CrossRef]
30. Wang, C.; Savkin, A.V.; Garratt, M. A strategy for safe 3D navigation of non-holonomic robots among moving obstacles. *Robotica* **2018**, *36*, 275–297. [CrossRef]
31. Kang, Y.; Hedrick, J.K. Linear tracking for a fixed-wing UAV using nonlinear model predictive control. *IEEE Trans. Control Syst. Technol.* **2009**, *17*, 1202–1210. [CrossRef]
32. Savkin, A.V.; Huang, H. Optimal aircraft planar navigation in static threat environments. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 2413–2426. [CrossRef]
33. Savkin, A.V.; Huang, H.; Ni, W. Securing UAV communication in the presence of stationary or mobile eavesdroppers via online 3d trajectory planning. *IEEE Wirel. Commun. Lett.* **2020**. [CrossRef]
34. Nigam, N.; Bieniawski, S.; Kroo, I.; Vian, J. Control of multiple UAVs for persistent surveillance: Algorithm and flight test results. *IEEE Trans. Control Syst. Technol.* **2011**, *20*, 1236–1251. [CrossRef]
35. Savkin, A.V.; Wang, C. A simple biologically inspired algorithm for collision-free navigation of a unicycle-like robot in dynamic environments with moving obstacles. *Robotica* **2013**, *31*, 993–1001. [CrossRef]
36. Hoy, M.; Matveev, A.S.; Savkin, A.V. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: A survey. *Robotica* **2015**, *33*, 463–497. [CrossRef]