



Precedence-constrained path planning of messenger UAV for air-ground coordination

Yulong DING^{1,2}, Bin XIN^{1,2,3†}, Jie CHEN^{1,2,3}

1.School of Automation, Beijing Institute of Technology, Beijing 100081, China;

2.Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing 100081, China;

3.Beijing Advanced Innovation Center for Intelligent Robots and Systems, Beijing Institute of Technology, Beijing 100081, China

Received 22 July 2018; revised 27 October 2018; accepted 28 October 2018

Abstract

This paper addresses an unmanned aerial vehicle (UAV) path planning problem for a team of cooperating heterogeneous vehicles composed of one UAV and multiple unmanned ground vehicles (UGVs). The UGVs are used as mobile actuators and scattered in a large area. To achieve multi-UGV communication and collaboration, the UAV serves as a messenger to fly over all task points to collect the task information and then flies all UGVs to transmit the information about tasks and UGVs. The path planning of messenger UAV is formulated as a precedence-constrained dynamic Dubins traveling salesman problem with neighborhood (PDDTSPN). The goal of this problem is to find the shortest route enabling the UAV to fly over all task points and deliver information to all requested UGVs. When solving this path planning problem, a decoupling strategy is proposed to sequentially and rapidly determine the access sequence in which the UAV visits task points and UGVs as well as the access location of UAV in the communication neighborhood of each task point and each UGV. The effectiveness of the proposed approach is corroborated through computational experiments on randomly generated instances. The computational results on both small and large instances demonstrate that the proposed approach can generate high-quality solutions in a reasonable time as compared with two other heuristic algorithms.

Keywords: Air-ground coordination, curvature-constrained path planning, precedence constraints, Dubins traveling salesman problem

DOI <https://doi.org/10.1007/s11768-019-8148-z>

[†]Corresponding author.

E-mail: brucebin@bit.edu.cn. Tel.: +86-10-68912463; fax: +86-10-68918232.

This work was supported in part by the National Outstanding Youth Talents Support Program (No. 61822304), in part by the National Natural Science Foundation of China (No. 61673058), in part by NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization (No. U1609214), in part by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (No. 61621063), in part by the Projects of Major International (Regional) Joint Research Program NSFC (No. 61720106011), and in part by International Graduate Exchange Program of Beijing Institute of Technology.

© 2019 South China University of Technology, Academy of Mathematics and Systems Science, CAS and Springer-Verlag GmbH Germany, part of Springer Nature

1 Nomenclature

- (x^A, y^A) : UAV position coordinate (unit: m);
- h^A : UAV heading angle (unit: rad);
- v^A : UAV flight speed (unit: m/s);
- N_G : The number of UGVs;
- N_T : The number of task points;
- $P_i^G(t)$: The location of the UGV i ($i \in \{1, 2, 3, \dots, N_G\}$) at time t (unit: m);
- P_i^T : The location of the task point i ($i \in \{1, 2, 3, \dots, N_T\}$) (unit: m);
- \mathbb{R}^2 : The two-dimensional real coordinate space.

2 Introduction

Heterogeneous cooperation between unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) has received considerable attention lately. There are strong complementarities between UAVs and UGVs promising a bright prospect in sensing, communication, payload abilities, and so on [1]. An increasing variety of applications in the field of military [2,3] and civilian [4,5] have shown that by incorporating different types of unmanned vehicles in a mission, accomplishment of more complex tasks with higher precision will be possible. Chen et al. [6] reviewed a number of influential and successful types of unmanned aerial and ground vehicle systems (UAGVSs) and proposed a taxonomy for classification of existing UAGVSs in which different types of UAGVSs can be described in a unified way.

In UAGVSs, multiple UGVs can act as mobile actuators to collaboratively perform dynamic tasks whose states dynamically change (e.g., see [7,8]). When UGVs are required to accomplish such tasks within a large area, they may be scattered and not be able to get the information about the others and tasks due to their limited perception or environment factors. The isolation may impede their collaborations. In this case, UAVs can serve as messengers for all UGVs to support their indirect communications and report the information about task states and other UGVs, which leads to a novel and interesting type of UAGVS.

By virtue of quick speed and less space constraint [9,10], UAV can fast capture the task and UGV information, and transmit the gathered information to UGVs. UAV needs to fly the task points firstly to collect all task information. After collecting task information, the messenger UAV needs to fly over the effective communication range of UGVs to collect and transmit the infor-

mation to UGVs, and can achieve the communications relay among the UGVs. The UGVs can update their decisions according to the newest information received from the UAV to improve their task efficiency. Therefore, this messenger mechanism can broaden the operating range of UGVs effectively to achieve multi-UGV collaboration in large area. An example of this scenario is shown in Fig. 1.

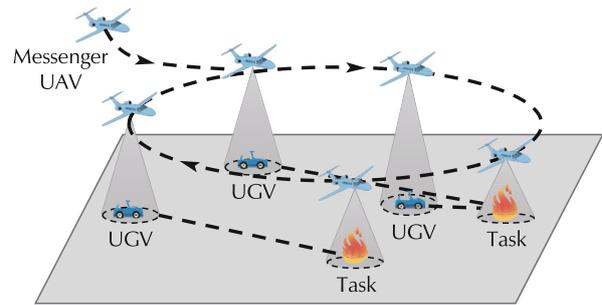


Fig. 1 An example of task scenario for a UAGVS operating in large areas.

In this paper, we will focus on the path planning for the messenger UAV in the novel UAGVS mentioned above. It is a new challenging problem which is crucial in achieving the effective air-ground coordination in the UAGVS.

The problem can be regarded as a complex special case of the *Dubins traveling salesman problem with neighborhood (DTSPN)* described in [11,12], hereinafter also being referred to as the *Precedence-constrained dynamic Dubins traveling salesman problem with neighborhood (PDDTSPN)*. This confronts us with the following challenges:

- i) To adapt the UGV’s motion, the UAV path needs to be dynamically computed and updated online by the developed algorithm. This requires the developed algorithm has relatively low computational time.
- ii) To plan a flyable UAV tour, we need to generate a smooth trajectory and guarantee that the trajectory respects the kinematic properties of the air vehicle.
- iii) This path planning is a complex optimization problem involving mixed variables. Some decision variables are constrained to be integers (e.g., the sequence of all UGVs for UAV to visit), while other decision variables (e.g., the access location and the heading of UAV) are continuous variables.
- iv) The access sequence for the UAV involves precedence relation: UAV needs to visit all task points firstly and then visit all UGVs.

This paper proposes a novel path planning problem

(PDDTSPN) which arises in a novel UAGVS with messenger mechanism. To our best knowledge, this online path planning problem is discussed for the first time. In addition, when solving this path planning problem, we apply a decoupling strategy to decompose the problem into two subproblems: one is to determine the sequence of task points and UGVs for the UAV to visit, and the other is to optimize the access locations of UAV in the neighborhood of each task point or each UGV. This strategy can achieve a desirable tradeoff between solution quality and time cost.

This work contrasts with our previous work [13] that focuses on path planning of messenger UAV which just needs to visit all UGVs and does not consider visiting task points. Our previous work assumes that all task states are static and known by UGVs in advance. However, in practice, the task states may change, and UGVs need to dynamically adjust their plans according to the state of each task. The UAV needs to collect the task information and then inform all UGVs. Therefore, compared with our previous work, this work is more suitable for air-ground coordination in dynamic tasks.

The paper is organized as follows. Section 3 will give a formal problem definition. Section 4 will present a heuristic method to determine the sequence of task points and UGVs for the UAV to visit. Section 5 will optimize the access locations of UAV in the neighborhood of each task point or each UGV. Section 6 will demonstrate the performance of our algorithm with some simulations. Finally, Section 7 will conclude the paper.

3 Problem formulation

A small fixed-wing UAV is chosen as a messenger. The motion parameters of UGVs (e.g., velocity and trajectory) are known by the UAV. Given the initial locations of moving UGVs and their motion parameters, the messenger UAV is responsible for flying over each moving UGV to acquire the information about UGVs and transmit the information about the others. In practice, UGVs and UAVs generally have limited communication range. Only when entering the communication range of each other, can they exchange information. In other words, UAV can transmit information to a UGV only if the messenger UAV is located within the communication range of the UGV. We specify the neighborhood of the UGV as a disk centered at the UGV in this paper. The valid communication radius for the UAV and the UGV i ($i \in \{1, 2, 3, \dots, N_G\}$) are denoted by r_{com}^A and

$r_{i,com}^G$, respectively. In addition, each task point can be regarded as a fixed UGV. The neighborhood of the task point can be defined as the perceivable range of UAV for the task point. The neighborhood radius of the task i ($i \in \{1, 2, 3, \dots, N_T\}$) is denoted by $r_{i,com}^T$.

When planning paths, the UAV just needs to pass neighbourhoods of all UGVs, rather than flying over their precise locations. Hence, it is demanded that UAV can plan a path to traverse the access location in each UGV's neighbourhood according to the planned sequence of all UGVs for the UAV to visit in the shortest time, where assume that information can be instantaneously transmitted from the UAV to each UGV. The planned sequence of all UGVs for the UAV to visit (briefly called *the visiting sequence of UGVs*) are denoted by $S^G = (s_1^G, s_2^G, \dots, s_{N_G}^G)$ where s_i^G ($i \in \{1, 2, 3, \dots, N_G\}$) is the index of the UGV for UAV to visit. For instance, $S^G = (3, 1, 2)$ means that UAV needs to visit UGV3, UGV1 and UGV2 in sequence. The access location, the access heading and the access time to visit the neighborhood of the UGV i ($i \in \{1, 2, 3, \dots, N_G\}$) are denoted by \tilde{P}_i , h_i and t_i , respectively.

In the same way for all task points, the UAV also just needs to access neighbourhoods of task points according to the planned visiting sequence. Let $S^T = [s_1^T, s_2^T, \dots, s_{N_T}^T]$ be the visiting sequence of task points where s_i^T ($i \in \{1, 2, 3, \dots, N_T\}$) is the index of the task point for UAV to visit. Since task point can be regarded as a fixed UGV, the access location, the access heading and the access time to visit the neighborhood of the task point i ($i \in \{1, 2, 3, \dots, N_T\}$) can be represented by \tilde{P}_i , h_i and t_i respectively.

In order to acquire a smooth and feasible path for messenger UAV, the Dubins model is used to describe the kinematic characteristics of the UAV. The Dubins model [14] is used to describe the aerodynamic of vehicles on a two-dimensional plane, and satisfies the following curvature motion constraints:

$$\begin{cases} \dot{x}^A = v^A \cos h^A, \\ \dot{y}^A = v^A \sin h^A, \\ \dot{h}^A = \frac{v^A}{r^A} u^A, \quad u^A \in [-1, 1], \end{cases} \quad (1)$$

where r^A is the minimum turning radius of the Dubins vehicle, u^A is the control input, and (x^A, y^A, h^A) represents the status of the Dubins vehicle (UAV), also called Dubins status. The definitions of v^A , x^A , y^A and h^A are given at the beginning of this paper.

It is known that there are six possible shortest path patterns, called Dubins paths, between any two configurations: RSL, LSR, RSR, LSL, RLR, and LRL [15], where L means turning left with the minimal turning radius, R means turning right with the minimal turning radius, and S means moving along a straight line. The shortest Dubins path between two configurations relies on both their positions and their heading. We use the optimal Dubins path under terminal heading relaxation [16, 17] to describe the paths of the messenger UAV. For Dubins paths with terminal heading relaxation, once the initial heading of UAV is fixed, the heading at the terminal point is determined by their relative position. The length of the shortest path of a Dubins vehicle from Dubins status (x^A, y^A, h^A) to a point $P_B \in \mathbb{R}^2$ with terminal heading relaxation is denoted as $D((P^A, h^A), P_B)$ where $P^A = (x^A, y^A)$. In the sense of terminal heading relaxation, the number of variables of Dubins paths can be decreased, which can effectively reduce the computational complexity [11].

For the messenger UAV, the travel cost between every pair of task points and UGVs can be measured by the length of the Dubins path between them. The objective of the problem J is to determine a path with minimum cost for the messenger UAV when the UAV traverses the valid neighbourhood of each task point and each UGV. We model the messenger UAV’s path planning as a PDDTSPN in which UGV’s locations dynamically change. The PDDTSPN model in the sense of terminal heading relaxation is represented as follows:

$$\begin{aligned}
 [S^*, P^*] = \arg \min \{ & J = D((P_{\text{initial}}^A, h_{\text{initial}}^A), \tilde{P}_{s_1^T}) \\
 & + \sum_{i=1}^{N_T} D((\tilde{P}_{s_i^T}, h_{s_i^T}), \tilde{P}_{s_{i+1}^T}) \\
 & + D((\tilde{P}_{s_{N_T}^T}, h_{s_{N_T}^T}), \tilde{P}_{s_1^G}) \\
 & + \sum_{i=1}^{N_G} D((\tilde{P}_{s_i^G}, h_{s_i^G}), \tilde{P}_{s_{i+1}^G}) \} \quad (2)
 \end{aligned}$$

subject to

$$\begin{cases} \dot{x}^A = v^A \cos h^A, \\ \dot{y}^A = v^A \sin h^A, \\ \dot{h}^A = \frac{v^A}{r^A} u^A, \quad u^A \in [-1, 1], \end{cases} \quad (3)$$

$$d(\tilde{P}_{s_i}, P_{s_i}^G(t_{s_i})) < \min(r_{\text{com}}^A, r_{i,\text{com}}^G), \quad i = 1, 2, \dots, N_G, \quad (4)$$

$$d(\tilde{P}_{s_i}, P_{s_i}^T) < \min(r_{\text{com}}^A, r_{i,\text{com}}^T), \quad i = 1, 2, \dots, N_T, \quad (5)$$

$$P_{s_j}^T < P_{s_i}^G, \quad \forall j = 1, 2, \dots, N_T, \quad i = 1, 2, \dots, N_G, \quad (6)$$

where $(P_{\text{initial}}^A, h_{\text{initial}}^A)$ is the UAV’s initial status, S^* and P^* are the obtained visiting sequence and access locations, respectively. The symbol $<$ implies that point i is visited before point j if $i < j$. The definitions of $N_G, N_T, P_i^G(t)$ and P_i^T are given at the beginning of this paper.

Constraint (4) ensures that the messenger UAV is located within the communication range of each UGV while Constraint (5) ensures that the messenger UAV is located within the communication range of each task point. Constraint (6) states the precedence constraint on visiting task points and UGVs.

To get a solution to the PDDTSPN, the visiting sequence of task points and UGVs (S^*) and the access locations for each neighborhood of task points and UGVs (P^*) should be determined. By relaxing dependencies between S^* and P^* , we develop an online algorithm to determine the visiting sequence at first, and then optimize the rendezvous path in each UGV’s neighborhood.

4 Heuristic for determining the visiting sequence

Generating the optimal visiting sequence through the neighborhoods of the task points and the UGVs is NP-hard. However, the complexity issue is an important part of an online algorithm, since the UAV path needs to be dynamically computed and updated, and therefore determining the visiting sequence should not become prohibitively time-consuming. An efficient approximation algorithm is used to determine the visiting sequence. The main idea of the approximation algorithm is that the optimal precedence-constrained Euclidean traveling salesman problem (PETSP) tour is computed for all task points as well as all UGVs, and then the order of visits for the PETSP is used as the visiting sequence for the UAV.

Consider a complete undirected graph $G = (V, E)$ with a set of vertices V representing the task points and the UGVs to be visited, and a set of arcs $E = \{(v_i, v_j) : i \neq j, v_i, v_j \in V\}$ connecting these vertices. The vertex set V can be partitioned into two clusters: one cluster is the set of task points $V_T = \{v_1, \dots, v_{N_T}\}$ and another is the set of UGVs $V_G = \{v_1, \dots, v_{N_G}\}$. E is associated with a cost matrix $C = [c_{ij}]_{(N_T+N_G) \times (N_T+N_G)}$ representing the Euclidean distances between vertices v_i and v_j ($v_i, v_j \in V$). The goal is to determine the shortest tour on G in which all vertices in V are visited only once, and V_T are visited before V_G are visited.

Lokin et al. [18] transformed this problem into a stan-

dard traveling salesman problem (TSP) by modification of the cost matrix. The cost matrix is modified by adding a large number M ($M \gg \max\{c_{ij}, \forall ij\}$) to the cost of each intercluster edge in matrix C , that is

$$c_{ij} = \begin{cases} c_{ij} + M, & \text{if } i \in V_T, j \in V_G \text{ or } i \in V_G, j \in V_T, \\ c_{ij}, & \text{otherwise.} \end{cases} \quad (7)$$

After the transformation, any exact algorithm for TSP can be applied to solve PETSP exactly [19]. The Lin-Kernighan heuristic algorithm proposed by Helsgaun et al. [20] is very effective to solve TSP. The visiting sequence will be obtained by means of the Lin-Kernighan heuristic algorithm with respect to the transformed matrix.

To adapt to the change of UGVs' locations, the idea of receding horizon optimization proposed in [21] is used to dynamically modify and update the visiting sequence:

- i) At any time, the UAV needs to determine the visiting sequence by using the above algorithm, taking into account the current position information of all task points and UGVs.
- ii) Apply the resulting visiting sequence to compute the path of the UAV to visit the next task point or UGV.
- iii) Repeat i) and ii) until all task points and all UGVs are visited.

5 Optimization of access location

In this section, we will optimize the access location in the neighborhood of each UGV and each task point.

5.1 UAV paths to visit a moving point

Lemma 1 (Theorem 1 in [13]) Assume that the UAV can obtain a point trajectory $p(t) = (x^p(t), y^p(t))$ in advance, and the maximum speed of the moving point v_{\max}^p is less than the UAV speed ($v_{\max}^p < v^A$). There exists t^* ($t^* \in [t_{\min}, t_{\max}]$), such that

$$D((P_{\text{initial}}^A, h_{\text{initial}}), p(t^*)) = v^A t^*,$$

where the lower bound of t^* is

$$t_{\min} = \frac{D((P_{\text{initial}}^A, h_{\text{initial}}), p(0))}{v^A + v_{\max}^p}, \quad (8)$$

and the upper bound of t^* is

$$t_{\max} = \frac{D((P_{\text{initial}}^A, h_{\text{initial}}), p(0))}{v^A - v_{\max}^p}. \quad (9)$$

Assume that the UAV can obtain the trajectory of $p(t)$ in advance and $v_{\max}^p < v^A$. According to Lemma 1, the UAV can visit a moving point at the time interval $[t_{\min}, t_{\max}]$. Accordingly, in order to obtain the UAV path to visit a moving UGV, we need to determine the time interval $[t_{\min}, t_{\max}]$ at first, and then obtain an approximate numerical solution for t^* by a *bisection approach*. Furthermore, the UAV path to visit a moving UGV and its corresponding rendezvous point $P^r(t^*)$ can be obtained. The detailed steps are shown in Algorithm 1.

Algorithm 1 (UAV path planning to visit a moving UGV)

Initialization: Given the point trajectory $p(t) = (x^p(t), y^p(t))$, the initial solution interval $[t_{\min}, t_{\max}]$ from (8) and (9), define $T(t) = D((P_{\text{initial}}^A, h_{\text{initial}}), p(t)) / v^A - t$.

Repeat

Find the midpoint $t_c = (t_{\min} + t_{\max})/2$ and then compute $T(t_{\min})$ and $T(t_c)$.

Update the solution interval $[t_{\min}, t_{\max}]$.

If $T(t_{\min}) \cdot T(t_c) > 0$ **then**

$t_{\min} = t_c$,

else

$t_{\max} = t_c$.

End if

Until The desired accuracy is reached.

$t^* = (t_{\min} + t_{\max})/2$.

The path to visit a moving UGV is a Dubins path from P_{initial}^A to the rendezvous point $P^r(t^*)$.

Apparently, the accuracy of the solution depends on the number of iterations, and can reach $(t_{\max} - t_{\min})/2^{N_b}$ after N_b iterations.

5.2 Determining access location in each neighborhood

Considering the limited communication range of UGV, a messenger UAV will achieve communication with UGVs as long as the UAV visits the UGV's neighborhood. It is clear that UAV must go through the boundary if it wants to visit any point in a neighbourhood, so we only consider the boundary points in the UGV's effective and reliable communication range. If the motion trajectory of the UGV is $P(t) = (x^G(t), y^G(t))$, the point on the boundary of the UGV's communication range can be represented precisely by its polar coordinates with respect to the centre of the neighbourhood. The trajectory of the boundary point can be described as $P(\beta, t) = (x^G(t) + r_{\text{com}}^G \cos \beta, y^G(t) + r_{\text{com}}^G \sin \beta)$, where β is the central angle from x-axis. Obviously, different β may lead to different rendezvous points with UAV.

The computational time of our approach is mainly composed of two parts: the time for determining the visiting sequence and optimizing the access locations.

The time complexity of determining the visiting sequence for each UGV or task point is $O(N_T + N_G)$. As discussed in Section 4, the process of determining the visiting sequence is based on the receding horizon optimization. After the messenger UAV visits a UGV or a task point, the visiting sequence needs to be re-determined, so the visiting sequence needs to be determined $N_T + N_G$ times. Therefore, the time complexity of determining the visiting sequence is

$$T_s = O((N_T + N_G)^2). \tag{13}$$

Assume that, when optimizing the access location in a UGV’s neighborhood, the sampling number N_{sample} is a constant. According to Section 5.2, the time complexity of optimizing the access locations is

$$T_p = O(N_{\text{sample}} \cdot (N_T + N_G)). \tag{14}$$

Therefore, the total computational complexity regarding a Dubins loop is

$$\begin{aligned} T_p + T_s &= O((N_T + N_G)^2 + N_{\text{sample}} \cdot (N_T + N_G)) \\ &= O((N_T + N_G)^2). \end{aligned} \tag{15}$$

6 Computational experiments

To evaluate the approach proposed in this paper, three computational experiments are conducted in this section. Experiment 1 shows the performance of our method to solve PDDTSPN. Experiment 2 is a comparative experiment in representative scenarios. Experiment 3 is a comparative experiment with different problem scales. All experiments were implemented in Matlab environment on a PC with Intel(R) Core (TM) CPU i5-4590 3.3 GHz, and 4 GB RAM.

6.1 Performance analysis

Fig. 3 shows a solution of the path planning of the messenger UAV constrained by the motion of UGVs. In this case, there are two task points whose locations are $P_1^T = (35, 0)$ m and $P_2^T = (-10, -30)$ m, respectively. There are two UGVs whose initial locations are $P_1^G(0) = (-35, 30)$ m and $P_2^G(0) = (10, 30)$ m, respectively. UGV 1 moves toward P_1^T with the speed vector $(1.83, -0.79)$ m/s, while UGV 2 moves toward P_2^T with the speed vector $(-0.63, -1.90)$ m/s. $r_{i,\text{com}}^T = 2.5$ m ($\forall i \in$

$\{1, 2, 3, \dots, N_T\}$), $r_{i,\text{com}}^G = 2.5$ m ($\forall i \in \{1, 2, 3, \dots, N_G\}$) and $N_{\text{loop}} = 2$.

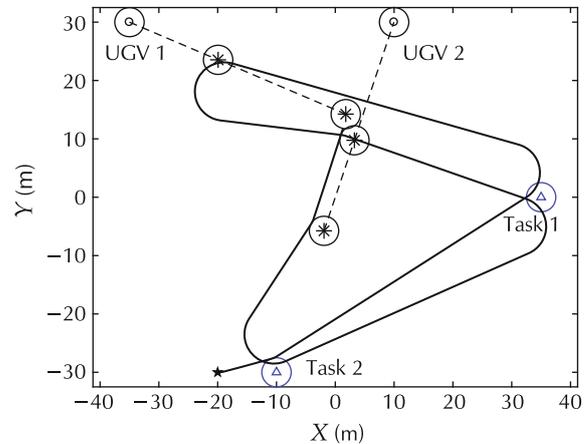


Fig. 3 Path planning of the messenger UAV. The scenario consists of two UGVs and two task points.

As shown in Fig. 3, according to the location of task points and the motion parameters of the moving UGVs, the messenger UAV can dynamically plan its path to visit the neighborhood of each task point firstly and then visit that of each moving UGV. In addition, the UAV can adjust its visiting sequence to adapt to the changing location of UGVs. In the first loop, the visiting sequence is Task1-Task2-UGV2-UGV1. However, with the movement of UGVs, the visiting sequence becomes Task2-Task1-UGV1-UGV2 in the second loop.

6.2 Comparative experiment in different representative scenarios

To investigate the performance of the algorithm proposed in this paper, we compare the results using the following methods for determining the access location:

1) *Boundary sampling* (BS, our method proposed in Section 5.3): the access locations corresponding to each task point and each UGV are the boundary points of the communication neighborhood of each task point and each UGV.

2) *Center sampling* (CS): the access location is the center of the communication neighborhood of each task point and each UGV.

The visiting sequences of the two methods are obtained by using the heuristic method proposed in Section 4.1.

Solutions of the two methods in the three representative scenarios are shown in Figs. 4–6. The presented results were obtained with the following parameters:

$$N_G = 3, \quad N_T = 3,$$

$$\begin{aligned}
 P_{\text{initial}}^A &= (-20, -30) \text{ m}, \quad h_{\text{initial}} = 0 \text{ rad}, \\
 P_1^G(0) &= (-35, 30) \text{ m}, \quad P_2^G(0) = (10, 30) \text{ m}, \\
 P_3^G(0) &= (0, 40) \text{ m}, \quad P_1^T = (30, 0) \text{ m}, \\
 P_2^T &= (-10, -30) \text{ m}, \quad P_3^T = (0, 0) \text{ m}, \\
 v_A &= 10 \text{ m/s}, \quad r_{i,\text{com}}^T = 2.5 \text{ m} \ (\forall i \in \{1, 2, 3, \dots, N_T\}), \\
 r_{i,\text{com}}^G &= 2.5 \text{ m} \ (\forall i \in \{1, 2, 3, \dots, N_G\}), \quad N_{\text{loop}} = 2.
 \end{aligned}$$

In Scenario 1 (see Fig. 4), all UGVs are fixed. In Scenario 2 (see Fig. 5), all UGVs move at different speed vectors. In Scenario 3 (see Fig. 6), UGV1 has a change of its speed during $5 \text{ s} < t < 15 \text{ s}$ while the motion parameters of the other UGVs are same with those of the corresponding UGVs in Scenario 2. From the analysis of these solutions in the three scenarios, it can be seen that the paths generated by CS and BS can make the messen-

ger UAV visit/revisit the communication neighbourhood of the moving UGVs periodically.

Table 1 shows the tour lengths and the average computing time for visiting each task point or each UGV used by the above two methods in the three representative scenarios. It can be seen that the lengths of Dubins tours obtained by CS are longer. For CS, the UAV needs to pass the center of each communication neighborhood, which can be regarded as a special case of BS. Compared with CS, sampling on the boundary in BS can provide larger search space, and therefore contribute to finding solutions with higher quality. Table 1 also indicates that CS is much faster than BS. BS has to search the neighbourhood of each task point or each UGV and consumes more time. However, the time cost of BS, approximately 0.2 s, is acceptable.

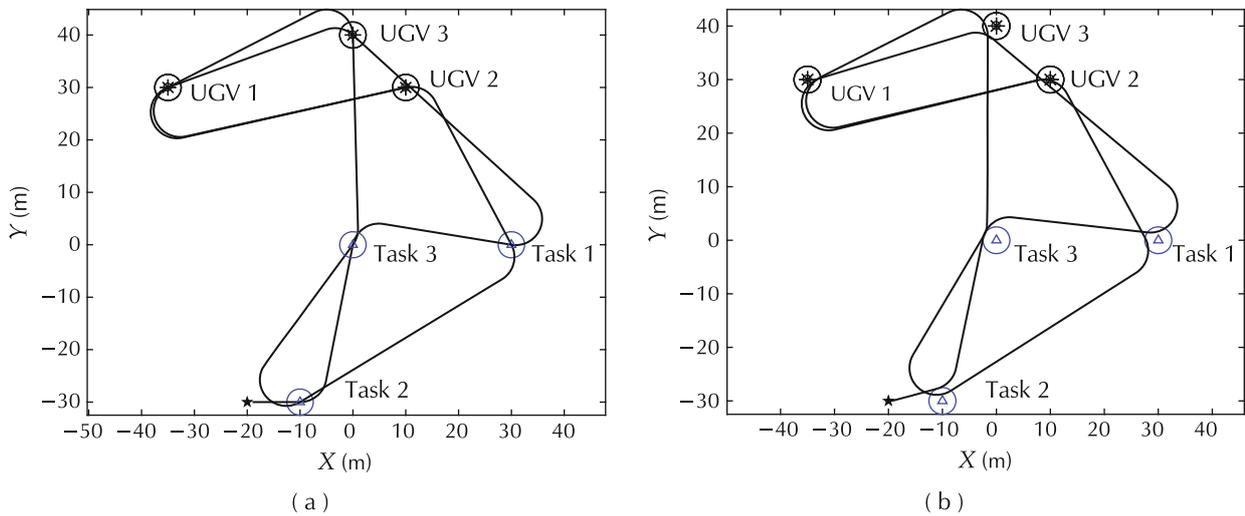


Fig. 4 Scenario 1: all UGVs are fixed. (a) Path planned by CS. (b) Path planned by BS.

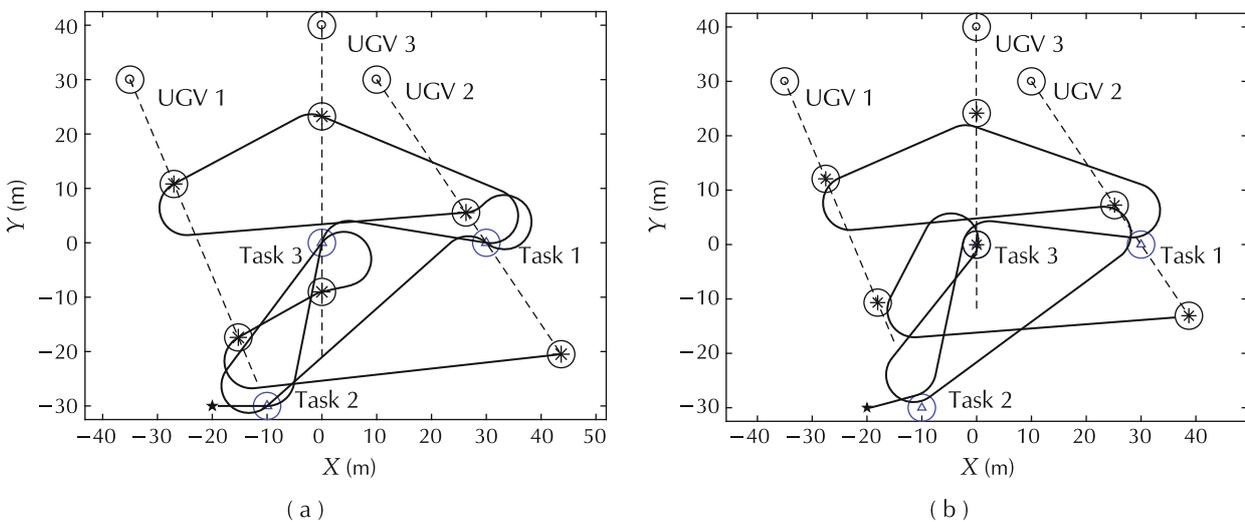


Fig. 5 Scenario 2: $\vec{v}_{UGV1} = (0.6, -1.4) \text{ m/s}$, $\vec{v}_{UGV2} = (0.8, -1.2) \text{ m/s}$, $\vec{v}_{UGV3} = (0, -1.5) \text{ m/s}$. (a) Path planned by CS. (b) Path planned by BS.

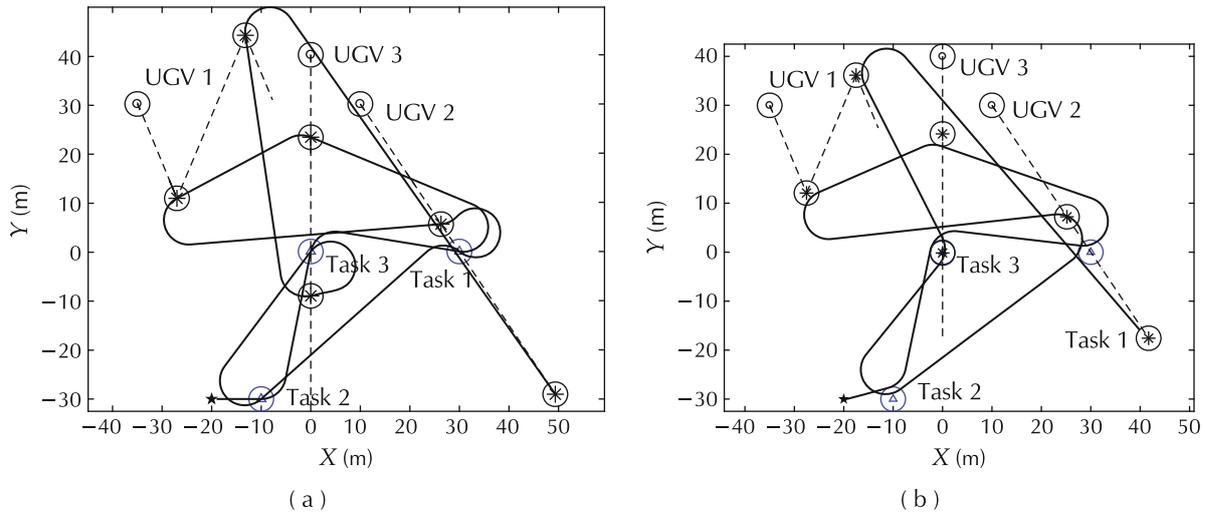


Fig. 6 Scenario 3: When $t < 5$ s, $\vec{v}_{UGV1} = (0.6, -1.4)$ m/s; when $5 < t < 15$ s, $\vec{v}_{UGV1} = (0.6, 1.4)$ m/s; when $t > 15$ s, $\vec{v}_{UGV1} = (0.6, -1.4)$ m/s. The speed of the other UGVs are similar to the corresponding UGVs in Scenario 2. (a) Path planned by CS. (b) Path planned by BS.

Table 1 Lengths of Dubins tours and time cost corresponding to the scenarios shown in Figs. 4–6.

Scenario	Center sampling		Boundary sampling	
	Tour length	Time (s)	Tour length	Time (s)
Scenario 1	500.11	0.1196	465.19	0.232
Scenario 2	428.49	0.1142	397.70	0.224
Scenario 3	451.71	0.1174	423.93	0.221

6.3 Comparative experiments on instances with different problem scales

In this section, comparative experiments on different instances are conducted to evaluate the scalability of our approach. The tested instances are labelled as Instance- N - i , where N is the number of task points, and i is the number of UGVs. For example, Instance-6-3 indicates 6 task points and 3 UGVs. In this subsection, all initial locations of UAV, UGVs and task points are generated randomly, the speed vectors of UGVs are also random vectors and $N_{loop} = 1$.

BS is compared with two algorithms: CS and a modified version of algorithm proposed in [13]. In [13], a geometry-center-based (GC) method was used to determine the visiting sequence but did not consider precedence constraints on task points. To access all task points firstly, determine the visiting sequence for task points using the geometry-center-based method at first, and then determine the visiting sequence for all UGVs using the same method.

The results obtained by the three algorithms over a set of 14 benchmark instances are reported in Table 2. As shown in Table 2, the approximate optimal solution

J^* is recorded for BS, and the optimal solutions found by CS and GC are recorded and labeled by J^{CS} and J^{GC} in Table 2, respectively. The gap between BS and CS is calculated as $GAP_1 = (J^{CS} - J^*)/J^{CS}$ while the gap between BS and GC is calculated as $GAP_2 = J^{GC} - J^*$. GAP_1 is recorded in percentage. The elapsed time for the computation of each algorithm is also recorded (see Table 2). Note that the elapsed time for the computation is the average computing time for visiting each task point or each UGV. The shortest path lengths for every instance are shown in bold.

As shown in Table 2, BS fully outperforms CS in term of solution quality for different instances (see GAP_1 in Table 2). GAP_1 increases markedly with the increase of the problem scale, ranging from 10.10% to 53.77%. The paths generated by GC are shorter than those of CS and BS on small instances (e.g., instance-4-4 and instance-10-15). However, on the larger instances, the paths found by GC are much longer than those of CS and BS (e.g., instance-100-100 and instance 60-150). This is because the greedy nature of GC makes it difficult to accommodate the complex spatial relationships among UGVs and task points with the increase of the problem scale.

Table 2 Computational results of CS, GC and BS on 14 instances.

Instance	BS		CS			GC		
	J^*	Time (s)	J^{CS}	Time (s)	GAP ₁ (%)	J^{GC}	Time (s)	GAP ₂
Instances-4-4	206.8	0.40	269.3	0.13	23.20	173.78	0.17	−33.04
Instance-4-6	416.9	0.29	533.9	0.16	21.91	356.46	0.15	−60.48
Instance-10-7	588.8	0.41	655.0	0.15	10.10	366.45	0.18	−222.37
Instance-20-40	13504.9	0.74	19384.7	0.53	30.33	17226.22	0.15	3721.28
Instance-20-60	9781.24	0.94	32519.7	0.77	69.92	52026	0.16	42244
Instance-30-35	21173.5	0.92	25424.6	0.75	16.72	25468	0.15	4294.66
Instance-30-60	24701	1.08	33102	1.00	25.38	62674	0.17	37972
Instance-40-80	88494	2.03	152521	1.50	41.97	3327103	0.17	3238609
Instance-40-100	172502	1.68	477424	1.49	63.87	3770257	0.16	3627838
Instance-50-100	615347	2.29	1029885	6.65	40.25	4277672	0.17	3662324
Instance-80-80	358274	4.67	441098	2.20	18.77	6842542	0.17	6706818
Instance-50-150	1377275	7.66	2979656	2.99	53.77	4277672	0.17	2900396
Instance-100-100	227126	4.66	255877	2.60	11.24	8807813	0.17	8580687
Instance-60-150	1026791	6.22	1871144	3.13	45.13	150706325	0.15	149679533

7 Conclusions

We present a novel path planning problem that arises in dynamic coordination between UGV and messenger UAV in air-ground coordination. This problem can be formulated as a PDDTSPN. We propose an algorithm in which a decoupling strategy is used to decompose the problem into two subproblems: one is to determine the visiting sequence of all tasks and UGVs and the other is to optimize the access locations of UAV in the neighborhood of each UGV. In our algorithm, a fast heuristic method was used to produce the precedence-constrained visiting sequence, while a boundary-based sampling method was adopted to optimize the access location of UAV in the neighborhood of each task point and each UGV.

Experiments in representative scenarios demonstrate that the paths generated by our algorithm can make the messenger UAV reliably and rapidly fly over the effective range of the task points and UGVs to collect and transmit the information. The computational results on both small and large instances demonstrate that the proposed approach can generate high-quality solutions in a reasonable time in contrast to two other algorithms. In the future work, the proposed approach will be extended to consider that the messenger UAV has incomplete or no information about the motion of UGVs in advance.

References

- [1] M. Saska, V. Vonasek, T. Krajník, et al. Coordination and navigation of heterogeneous MAV-UGV formations localized by a ‘hawk-eye’-like approach under a model predictive control scheme. *International Journal of Robotics Research*, 2014, 33(10): 1393 – 1412.
- [2] J. Y. C. Chen. UAV-guided navigation for ground robot teleoperation in a military reconnaissance environment. *Ergonomics*, 2010, 53(8): 940 – 950.
- [3] R. Bogue. The role of robots in the battlefields of the future. *Industrial Robot*, 2016, 43(4): 354 – 359.
- [4] E. Garone, J. F. Determe, R. Naldi. Generalized traveling salesman problem for carrier-vehicle systems. *Journal of Guidance Control and Dynamics*, 2014, 37(3): 766 – 774.
- [5] N. Mathew, S. L. Smith, S. L. Waslander. Planning paths for package delivery in heterogeneous multirobot teams. *IEEE Transactions on Automation Science and Engineering*, 2015, 12(4): 1298 – 1308.
- [6] J. Chen, X. Zhang, B. Xin, et al. Coordination between unmanned aerial and ground vehicles: a taxonomy and optimization perspective. *IEEE Transactions on Cybernetics*, 2016, 46(4): 959 – 972.
- [7] C. G. Cassandras, X. Lin, X. Ding. An optimal control approach to the multi-agent persistent monitoring problem. *IEEE Transactions on Automatic Control*, 2013, 58(4): 947 – 961.
- [8] B. Xin, Y. G. Zhu, Y. L. Ding, et al. Coordinated motion planning of multiple robots in multi-point dynamic aggregation task. *Proceedings of the IEEE International Conference on Control and Automation*, Nepal: IEEE, 2016: 933 – 938.
- [9] S. Lai, H. Qin, J. Q. Cui, et al. A robust online path planning approach in cluttered environments for micro rotorcraft drones.

Control Theory and Technology, 2016, 14(1): 83 – 96.

- [10] P. Kolaric, C. Chen, A. Dalal, et al. Consensus controller for multi-UAV navigation. *Control Theory and Technology*, 2018, 16(2): 110 – 121.
- [11] B. Xin, J. Chen, D. L. Xu, et al. Hybrid encoding based differential evolution algorithms for Dubins traveling salesman problem with neighborhood. *Control Theory and Applications*, 2014, 31(7): 941 – 54.
- [12] J. T. Isaacs, D. J. Klein, J. P. Hespanha. Algorithms for the traveling salesman problem with neighborhoods involving a Dubins vehicle. *Proceedings of the American Control Conference*, New York: IEEE, 2011: 1704 – 1709.
- [13] Y. L. Ding, B. Xin, J. Chen, et al. Path planning of messenger UAV in air-ground coordination. *The 20th World Congress of the International Federation of Automatic Control*, Toulouse: IFAC Secretariat, 2017: 8045 – 8051.
- [14] L. E. Dubins. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 1957, 79(3): 497 – 516.
- [15] P. Isaiah, T. Shima. Motion planning algorithms for the Dubin travelling salesperson problem. *Automatica*, 2015, 53(1): 247 – 255.
- [16] X. N. Bui, J. D. Boissonnat. *Accessibility Region for A Car that Only Moves Forwards along Optimal Paths*. Technical report. France: Institut national de recherche en informatique et en automatique, 1994: Research Report INRIA 2181.
- [17] X. Zhang, J. Chen, B. Xin, et al. A memetic algorithm for path planning of curvature-constrained UAVs performing surveillance of multiple ground targets. *Chinese Journal of Aeronautics*, 2014, 27(3): 622 – 633.
- [18] F. Lokin. Procedures for travelling salesman problems with additional constraints. *European Journal of Operational Research*, 1979, 3(2): 135 – 141.
- [19] Z. H. Ahmed. The ordered clustered travelling salesman problem: a hybrid genetic algorithm. *Scientific World Journal*, 2014: DOI 10.1155/2014/258207.
- [20] K. Helsgaun. An effective implementation of the lin-kernighan traveling salesman heuristic. *European Journal of Operational Research*, 2000, 126(1): 106 – 130.
- [21] X. Zhang, Y. Bai, B. Xin, et al. Differential evolution based receding horizon control for UAV motion planning in dynamic environments. *Jiqiren/Robot*, 2013, 35(1): 107 – 114.



Yulong DING received the B.Sc. degree from Qilu University of Technology, Jinan, China, in 2012, and the M.Sc. degree from Xiamen University, Xiamen, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Automation, Beijing Institute of Technology, Beijing. His current research interests include UAV-UGV collaborative systems, heterogeneous multi-agent systems, and task planning of multi-robot systems. E-mail: dingyulong@163.com.



Bin XIN received the B.Sc. degree in Information Engineering and the Ph.D. degree in Control Science and Engineering, both from the Beijing Institute of Technology, Beijing, China, in 2004 and 2012, respectively. He was an academic visitor at the Decision and Cognitive Sciences Research Centre, the University of Manchester, from 2011 to 2012. He is currently an associate professor with the School of Automation, Beijing Institute of Technology. His current research interests include search and optimization, evolutionary computation, combinatorial optimization, and multi-agent systems. He is an Associate Editor of the Journal of Advanced Computational Intelligence and Intelligent Informatics and the journal Unmanned Systems. E-mail: brucebin@bit.edu.cn.



Jie CHEN received the B.Sc., M.Sc., and Ph.D. degrees in Control Theory and Control Engineering from the Beijing Institute of Technology, in 1986, 1996, and 2001, respectively. From 1989 to 1990, he was a visiting scholar in the California State University, U.S.A. From 1996 to 1997, he was a research fellow in the School of E&E, University of Birmingham, U.K. He is currently a professor of control science and engineering, the Beijing Institute of Technology, China. He is also an academician of the Chinese Academy of Engineering. He serves as a managing editor for the Journal of Systems Science and Complexity (2014-2017) and an associate editor for the IEEE Transactions on Cybernetics (2016-2018) and many other international journals. His main research interests include intelligent control and decision in complex systems, multi-agent systems, and optimization methods. He has co-authored 4 books and more than 200 research papers. E-mail: chenjie@bit.edu.cn.