Sampling Transformation Based Path Planning For Messenger UAV in Urban Environment Messenger

Benxiang Wang[1], Bin Xin[1,*], Yulong Ding[2], and Yang Li[3]

[1]School of Automation, Beijing Institute of Technology, Beijing, China
E-mail: brucebin@bit.edu.cn (Corresponding author)
[2] Peng Cheng Laboratory, Shenzhen, China
[3] Beijing Institute of Electronic Engineering, Beijing, China

Abstract. In recent years, unmanned platforms are considered to be an important part of the Internet of Things system. However, in the face of communication-limited special tasks such as large-scale logistics and multi-obstacle urban warfare, information isolation will greatly affect the efficiency of the system. In this case, indirect communication between unmanned platforms can be realize by the mobile relay of messenger UAV. This paper mainly introduces a path planning scheme of messenger UAV in urban environment, aiming at minimizing the cost time to access all mobile UGVs (unmanned ground vehicles). The paper models this problem as a special dynamic generalized traveling salesman problem (DGTSP), and proposes a sampling transformation algorithm. The algorithm discretizes the paths of UGVs through a sampling mechanism, and effectively transforms the problem into asymmetric traveling salesman problem (ATSP) by using motion relationship. Finally, the effectiveness of the algorithm is verified by computational experiments with instances of different sizes.

Keywords: Path Planning, Messenger Mechanism, Urban Environment, Traveling Salesman Problem

1. INTRODUCTION

In the past decade, Unmanned Platform such as UAV, UGV is considered as one of the most potent and important emerging industries. With the progress of related technologies, application scenarios have gradually separated from the repetitive work in traditional industrial manufacturing and continuously expanded to other fields. In most applications, Unmanned Platforms can often rely on efficient wired or wireless networks to form a multi-agent Internet of Things system [1] and efficiently complete a given task efficiently. However, in the face of special tasks such as large-scale logistics delivery, complex field search and rescue, or urban operations, the capability of a single unmanned platform is relatively limited, and it is vulnerable to environmental factors. The information isolation will reduce the real-time and effectiveness of the unmanned system [2]. How to complete the tasks safely and efficiently has become a difficult problem of wide attention. The UAV-UGV air ground coordination is a feasible and ideal solution to the above problem. Through the complementary advantages, they can complete a given task harmoniously and efficiently.

UAV and UGV agents are very different in functional positioning, scene adaptation, and task execution capabilities, and both have their own limitations. But they can complement each other in many ways and achieve synergistic advantages. The UAV is less affected by obstacles during an air flight, can quickly pass-through obstacles, rough terrain, or steep slopes, and can provide a view from a high altitude. However, compared to UGV, UAV can carry a relatively limited load, and its capacity limits the equipment load and energy load, further limiting the running time. The UGV moves on the ground and can be equipped with more mission loads to perform various specific operations on the ground targets[3]. However, the terrain and obstacles have a great impact on their traffic and communication, their actions are not as convenient and free as UAV, and the long-distance observation cannot be completed. In short, UAV has air advantage, generally plays the role of the mobile sensor, and is mainly responsible for the acquisition of task information[4]; UGV has the ground advantage and generally assumes the role of the mobile actuator, directly performs tasks.

UAV-UGV air ground cooperation is a heterogeneous system, which improves the overall action capability, perception capability, and mission capability. It can better adapt to the environment and tasks and can better adapt to special tasks such as urban operations. From the Iraq war, Russia Ukraine conflict, and other modern war experiences, urban operations must consider "Accessibility, Visibility, Communication". Complex and changeable geographical environment and spectrum situation will offset the war advantages of the traditional communication and command system and bring great challenges to information sharing and data transmission. UAV can act as a messenger function[5], taking advantage of its high-altitude work, high-speed movement, and not easily blocked by obstacles. When the traditional ground network is destroyed, or the combat platform is scattered in a large area, UAV will provide communication services for each UGV platform as auxiliary facilities. An example of such an air ground coordination heterogeneous system is given in Figure 1, which the messenger UAV can solve communication problems through large-scale motion. The In the system,

UGV ground platforms can move to the target point in the urban road network and complete the task,. The UAV air platform can regularly fly over the UGV ground platform to realize the indirect communication, acting as a messenger to collect and transmit information to the UGV. The UGV platforms exchange information through the messenger UAV to establish a special communication network, which can realize collaboration in a large range of areas that cannot be communicated directly.

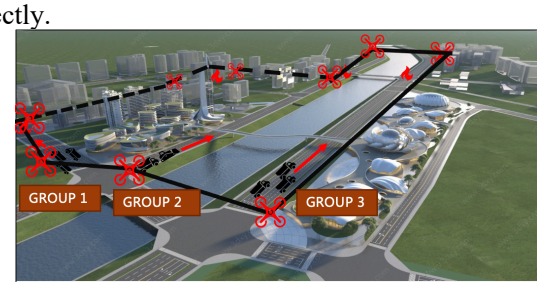


Fig. 1 Example Of An Air Ground Coordination Heterogeneous System With The Messenger Mechanism

This paper studies the path planning of messenger UAV in the above air ground coordination heterogeneous system, which can be regarded as Generalized Traveling Salesman Problem(GTSP), and the optimization goal is to find the fastest path with access to all mobile UGV platforms. For multi-UGV platform target access tasks, task planning involves not only path planning issues but also access order arrangement issues. When planning the solution, it is not only necessary to design the transfer path between adjacent targets but also to consider reasonably planning the access order of each target to obtain the overall optimal UAV flight scheme. Since the UGVs are constantly moving, the planning of access sequences is often more difficult than transfer path design. The existing algorithm [6-9] generally targets empty airspace and ignores the impact of obstacles on the flight trajectory. When using search methods such as GA and PSO, the evaluation value of each individual needs to be obtained through complex path planning, which will greatly improve the time cost of the algorithm.

This paper presents a solution algorithm based on sampling transformation(STA). Firstly, coarse-grained discrete sampling is carried out for the trajectory of UGV platforms. Then, we determine the cost matrix according to the relationship between the UAV and the target motion, which avoids multiple calling path planning compared to the search algorithm. The ATSP (asymmetric traveler problem) solver is called to get the access order, and then the discrete interval is slowly narrowed down so that the access order convergence is solved. At the same time, the UAV access position is obtained, and the flight path is planned.

The various parts of this article are contained as follows: The second part mainly gives the definition and the mathematical model of the problem. The third part mainly introduces the solution algorithm of the problem. The fourth part is the analysis of the algorithm performance. By changing the scale of the calculation examples, the algorithm running time and the UAV route time in different situations are analyzed and compared. The last part is a summary of this article.

2. PROBLEM MODELING

2.1. Urban environment model

The city has vertical and horizontal streets and tall, solid and dense buildings, which can impact the accessibility, visible ability, and communication ability. In this paper, the urban environment is simplified to the abstract road web of streets and the abstract obstacles of buildings, as shown in Figure 2.

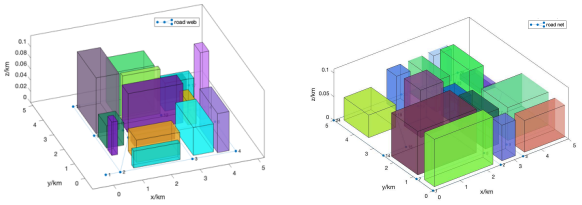


Fig. 2 Example of the Urban Environment Model

2.2. Models of the UGV and UAV

In this paper, UGV is modeled as the moving point in the urban road web, and the UAV is the moving point in the air as shown in Figure 3.Any section of the road network E is a line section, and the unmanned car can only drive on the road network, and the beginning and endpoint are all points in the road network. The UAV is not limited by the road network. Suppose that the known UGV speed is limited to a range, and the maximum speed V_G^{max} is much less than V_{UAV}.

The small rotor UAV is selected as the messenger UAV that satisfies the following motion constraints:

$$\begin{cases} \dot{x} = V_{UAV}\cos\theta\cos\varphi \\ \dot{y} = V_{UAV}\cos\theta\sin\varphi \\ \dot{z} = V_{UAV}\sin\theta \end{cases} \quad (1)$$

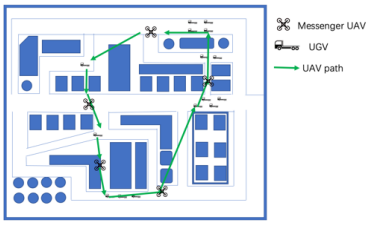


Fig. 3 Example of the Movement Path of An Unmanned Platform in Urban Environment

2.3. Models of the problem

The traveling salesman problem(TSP) is a widely studied combinatorial optimization problem, which, given a set of cities and a cost to travel from one city to another, seeks to identify the tour that will allow a salesman to visit each city only once, starting and ending in the same city, at the minimum cost. For messenger UAV, the travel cost to UGV is not fixed: the access order of UAV to UGV affects the access time and access location of each UGV, and the UAV path planning is very dependent on the access location. Based on this, we model the UAV

path planning problem as a special Dynamic Generalized Traveling Salesman Problem(DGTSP). Compared with the traditional TSP problem, this problem is more difficult to solve, and it is an NP-hard problem containing the access order and access location.

The model of the problem is as follows:

$$[S^*, P^*, T^*] = argmin\{\sum_{i=0}^{N_G} L(P_{S_i}, P_{S_{i+1}})\} \quad (2)$$

$$S_0 = 0; S_{N_G+1} = 0; P_0 = P_A \quad (3)$$

$$P_i(t) = (x_i(t), y_i(t)), i = 1,2, \dots, N_G \quad (4)$$

$$(x_i(t), y_i(t)) \in E, i = 1,2, \dots, N_G \quad (5)$$

$$L(P_{S_i}, P_{S_{i+1}}) = \overline{P_{S_i} P_{S_{i+1}}} + 2(H_{obs} + d_{obs}) \quad (6)$$

S^*, P^*, T^* is the optimized goal. S^* represents the access order. P^* represents the access location. T^* represents the tour cost time;

N_G is the number of UGVs, L is the range length of UAV;

Formula (3) indicates that the initial position and final position of the UAV are P_A;

Formula (4) and (5) represent that the position of UGV_i at time t is $(x_i(t), y_i(t))$, which must belong to the road network E ;

In formula (6), $L(P_{S_i}, P_{S_{i+1}})$ represents the range of UAV from position P_{S_i} to position $P_{S_{i+1}}$. The motion trajectory of UAV is simplified to directly cross the obstacle building, H_{obs} is the obstacle height of the building, and d_{obs} is the obstacle avoidance distance.

3. SAMPLING TRANSFORMATION ALGORITHM

In the Dynamic Generalized Traveling Salesman Problem(DGTSP), UGVs are inconstantly moving in the road network, which means the distance from one city to another city is a dynamic variable related to the access order and access time. So many classical algorithms of TSP can't be directly applied to this problem. This paper proposes a solution algorithm based on sampling transformation, which can transform the original problem into a classic problem that can be quickly solved with the existing solution tools..

In Sampling Transformation Algorithm(STA),the waypoint can be considered as the access location with some efficiency lost, and the access cost can be determined according to the motion relationship between UAV and target, so as to quickly construct the cost matrix. Then, through the transformation of the cost matrix, the original problem can be skillfully transformed into a classical ATSP problem that can be solved quickly with the existing solving tools. The discrete sampling interval of waypoints reflects the error of this method to a certain extent. When the discrete sampling interval is too large, the gap between waypoints and the actual access location is large, resulting in unnecessary waste; When the discrete sampling interval is too small, the cost of constructing the cost matrix and solving the ATSP problem is too high, and a good solution may not be found within the specified time. Therefore, at the beginning of this paper, the coarse-grained discrete sampling is used to call the ATSP solver to obtain the access order, and then the discrete sampling interval is slowly reduced in exponential form near the access point by using the bisection method. The trajectory near each access location is re discretized in smaller steps. After inserting n discrete sampling points (cities), the cost matrix cost is updated and $(n * n + 2 * n * N)$ elements are added. The algorithm can find a better solution based on the previous iteration. When solving the problem of converting to ATSP, this paper calls the best recognized TSP solver LKH, and takes no change in the access order of LKH solution as one of the convergence conditions of this algorithm. The whole process is as follows:

STA procedure

for i from 1 to N_G **do**
  discrete sampling the path i
end for
for m from 1 to N **do**
  for n from 1 to N **do**
    construct cost matrix;
    end for
end for
for i from 1 to i_{max} **do**
  if i>1 **then**
    get the meeting points by last access order;
    discrete sampling points near the meeting point with binary insertion;
    update cost matrix;
  end if
  transform cost matrix to ATSP with Noon-Bean;
  calling LKH solver program to get access order;
  for m from 1 to N_G **do**
    find the access pos;
  end for
end for
return the best solution;

Notes: i_{max} - the maximum number of iterations, similarly hereinafter.

In step initialization, the DGPST problem is transformed into a GTSP problem. It can be considered that there are $N_G + 1$ different clusters in the whole problem, of which the first cluster is the starting point of the UAV and the other N_G clusters are the trajectory points of UAV discretized according to the operation in initialization. Generally speaking, the movement cost between cities refers to the operation time of the UAV, which represents the time difference between $P_i(t)$ and $P_j(t)$, that is, the sum of movement time of UAV from $P_i(t)$ to $P_j(t)$ and the waiting time at $P_j(t)$ However, $P_i(t)$ and $P_j(t)$ are waypoints with time information, and it is only legal for UAV to complete movement within Δt. When $P_i(t)$ and $P_j(t)$ belong to the same city cluster, and the cost does not need to be calculated.

Specifically, the constructed cost matrix is an $(N) \times (N)$matrix, i.e.

$$M = \begin{bmatrix} m_{11} & \cdots & m_{1\times(N)} \\ \vdots & \ddots & \vdots \\ m_{(N)\times 1} & \cdots & m_{(N)\times(N)} \end{bmatrix} \quad (7)$$

$m_{[(start_i+m-1),(start_j+n-1)]}(i, j = 1,2,\cdots,N \text{ and } i \neq j)$ represents the generation value of the UAV from the discrete point m on the target i to the discrete point n on the target j, where $start_i$ represents that the city cluster i records from the $start_i$ element. When the UAV can reach $P_j(n)$ from $P_i(m)$ before UGV_j, the UAV can hover in position $P_j(n)$ and wait for target j to arrive. The value of this era is $t(n) - t(m)$; On the contrary, if the UAV cannot arrive at $P_j(n)$ from $P_i(n)$ in time, it will miss UGV_j and cannot access the target. At this time, the cost value is:

$$\Delta t = t(n) - t(m) \quad (8)$$

$$m_{[(start_i+m-1),(start_j+n-1)]} = \begin{cases} \Delta t, & t_{cost} < \Delta t \\ M, others \end{cases} \quad (9)$$

In particular, it can be considered that the initial position $P_A(0)$ of UAV is also an independent cluster, and the cost from $P_A(0)$ to other waypoints can also be calculated by the above method. Last, we set $m_{[(start_j+n-1)]\times 1}$ to represent the generation value of the UAV returning to the initial position $P_A(0)$ from the discrete point n on the target j. This cost value is the time when the UAV returns to the initial position $P_A(0)$ from the discrete point n:

$$m_{[(j-1)\cdot n_d+n]\times 1} = dis(P_j(t(n), P_A(0)))/v_A \quad (10)$$

In step Iteration, the road points with shorter time intervals and higher accuracy need to be obtained through the bisection operation. We can get the meeting points by access order, which is obtained in last iteration using TSP solver. Then discrete sampling points near the meeting point as (11) and update cost matrix. Taking the $(start_i + m - 1)$ and $(start_i + m)$ waypoints as an example, a new waypoint can be inserted between them. Its position is $P_i(t(x))$, and

$$t(x) = \frac{1}{2(t(m)+t(m+1))} \quad (11)$$

Then, only the $start_i$ needs to be updated, and the corresponding cost vector can be inserted at the $(start_i + x - 1)$ row and column positions in the cost matrix to update the cost matrix.

For the GTSP problem constructed above, we can make full use of the characteristics of its cluster; that is, each discrete point on the same target j trajectory can be considered to belong to the same point cluster, and the same point cluster can be considered to complete the traversal of all cities only by visiting once. Therefore, this paper uses the idea of Noon-Bean transformation [10], adds a zero-cost virtual unidirectional ring in each point cluster C_j, and connects all the points in it, that is, adds the following zero cost directed edges and modifies the cost matrix.

$$\begin{Bmatrix} e\left(P_j(t(1)), P_j(t(2))\right), \cdots, \\ e\left(P_j(t(n-1)), P_j(t(n))\right), e\left(P_j(t(n)), P_j(t(1))\right) \end{Bmatrix} \quad (12)$$

Then, we modify the starting $P_i(t(m))$ of all directed edge $e\left(P_i(t(m)), P_j(t(n))\right)$, where $i \neq j, n = 1, \cdots, n_d$, and switch it to the previous point $P_i(t(m-1))$ on the virtual unidirectional ring. Thus, the GTSP problem can be transformed into a special CTSP problem, which is equivalent to original GTSP. Last, in order to facilitate processing, this paper simply adds a large number of costs M to all inter-cluster arcs and converts the cluster TSP into a standard asymmetric TSP problem, which can be solved directly by the LKH solver[11] to generate access order.

4. COMPUTATIONAL EXPERIMENTS

To verify the effectiveness and advantages of the proposed algorithm, this paper compares it with classical genetic algorithm(GA), heuristic method(HA) and exhaustive attack method(EA). The results are averaged after each algorithm runs for 10 times. When there are a large number of obstacles in the urban environment, the GA method needs to calculate the evaluation individuals' value for collision avoidance detection, and the high cost leads to no effective comparison. Therefore, the search process of the GA will not consider the obstacles. After the best access order is obtained in each iteration, the collision avoidance path length is solved.

- Genetic algorithm(GA): the individual population is the access order, and the evaluation value is the total cost of the access order in the case of accessibility. In this search algorithm, it is unwise to calculate the obstacle avoidance distance in the environment with a large number of obstacles. Therefore, collision avoidance detection is only carried out for the optimal individual in each iteration. The population size is 50 and the number of iterations is 100.
- Heuristic construction algorithm(HA) based on the nearest distance under time-domain rolling: the access choice of UAV to UGV depends on the access cost.
- Exhaustive attack method(EA) at high sampling frequency: discretely sample all UGV tracks at high sampling frequency, and enumerate all possible access sequences to obtain the optimal path. If the sampling frequency is high enough, this method can obtain the optimal solution of the problem. This method is only used for the benchmark of the optimal solution under the barrier free example. In the experiment, the discrete sampling interval of trajectory is 0.01 min.

In experiments, to ensure the realistic significance of the problem, the example parameters are generated randomly generated by the program. Considering the influence of obstacles on the algorithm, the distinction of obstacle size in the example design. The example name *Ci-Tj-O* represents the *j* the example under the number of UGV i, and the example name ending as -O represents a large number of obstacles in the case. The example name ending as -S represents a different network with obstacles. Figure 4 shows the UAV access results in C4-T1 example. All experiments were performed on a

computer configured with Intel (R) core (TM) CPU i5-4590, 3.3 GHz, and 4 GB ram. The Gap value (%) in the table represents the gap between the optimal solution obtained by GA or heuristic method and the optimal solution obtained by sampling transformation algorithm.

$$GAP = 1 - J/\min\left(J_{GA}, J_{HE}\right)) \quad (13)$$

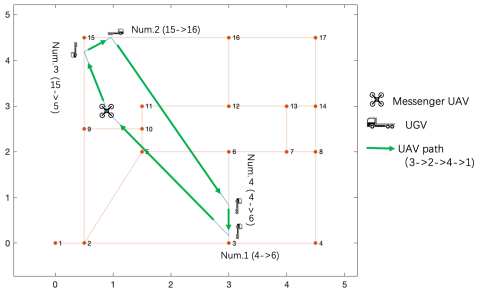


Fig. 4 Example of UAV Motion Path in the Calculation Example C4-T1

Table 1 Results of the Computational Experiments (Tour Time(min) T_{RUN} (s))

Instance	EA		GA	
	Tour Time	T_{RUN}	***Tour Time***	T_{RUN}
C3-T1	**7.899**	**0.1**	**7.899**	~
C4-T1	**10.281**	0.2	**10.281**	~
C5-T1	**12.144**	0.61	**12.144**	~
C6-T1	**12.416**	3.9	**12.416**	30.560
C7-T1	**12.800**	21.69	12.857	28.450
C8-T1	**10.269**	227	10.310	29.410
C8-T2	**11.569**	217	12.379	44.108
C9-T1	**11.091**	1792	11.180	66.340
C9-T2	**15.453**	1598	17.017	71.482
C10-T1	**11.879**	16486	12.150	82.950

Instance	HA		STA		
	Tour Time	T_{RUN}	***Tour Time***	T_{RUN}	***Gap(%)***
C3-T1	7.899	**0.136**	**7.899**	6.480	0.000
C4-T1	10.281	**0.100**	**10.281**	5.417	0.000
C5-T1	12.947	**0.234**	**12.144**	3.432	0.000
C6-T1	NAN	NAN	**12.416**	**20.932**	0.000
C7-T1	14.614	**0.546**	**12.800**	23.646	0.443
C8-T1	12.034	**0.363**	**10.269**	7.430	0.398
C8-T2	NAN	NAN	12.343	**30.797**	0.293
C9-T1	11.165	**0.663**	**11.091**	26.290	0.661
C9-T2	NAN	NAN	15.634	**15.396**	8.124
C10-T1	NAN	NAN	**11.879**	**49.580**	2.230

Notes:~ means that the optimal solution appears in the initial population, so the time is meaningless; NAN represents that the access order found is not realizable; Evaluation value UAV tour cost time (min), Running time T_{RUN} (s)

The following conclusions can be drawn from table 1. Firstly, when the sample size is relatively small ($N_G < 6$), it is easy to find the optimal solution in a small solution space, and the difference between the three algorithms is very small. Secondly, compared with the sampling method, eight of the ten examples can find the optimal solution in the STA algorithm, and only C8-T2 and C9-T2 do not find the optimal solution, but the gap is less than 7%. Third, there are certain advantages between the optimal solution of STA and the optimal solution of the comparison algorithm. With the increase of the example scale, the advantages of this algorithm are more obvious, especially in the C9-T2 example. Fourth, the time of Sampling transformation algorithm is significantly lower than GA algorithm and higher than heuristic algorithm.

Table 2 Results of the Computational Experiments (Tour Time(min) T_{RUN} (s))

Instance	GA		HA		STA		
	Tour Time	T_{RUN}	***Tour Time***	T_{RUN}	***Tour Time***	T_{RUN}	***Gap (%)***
C3-T2-O	**8.1**	~	**8.1**	0.4	**8.1**	3.5	0.0
C6-T2-O	**13.0**	21.3	NAN	NAN	**13.0**	**21.6**	0.0
C7-T2-O	13.8	42.7	15.8	**3.3**	**13.0**	63.7	5.5
C8-T3-O	11.0	59.6	12.8	**2.4**	**11.0**	11.5	0.0
C9-T3-O	12.0	42.4	11.6	**3.7**	**11.5**	35.8	0.3
C9-T4-O	17.0	98.1	NAN	NAN	**16.3**	**18.0**	4.2
C9-T5-O	14.5	67.4	NAN	NAN	**11.8**	**28.7**	19.0
C10-T2-O	16.0	61.7	NAN	NAN	**12.5**	**36.9**	22.0
C10-T3-O	14.7	70.0	15.2	**6.0**	**13.2**	36.8	9.9
C10-T4-O	16.6	62.5	NAN	NAN	**14.4**	**38.7**	13.4

Notes:~ means that the optimal solution appears in the initial population, so the time is meaningless; NAN represents that the access order found is not realizable; Evaluation value UAV tour cost time (min), Running time T_{RUN} (s)

In the example in Table 2, a large number of obstacles are added, small-scale samples are reduced and large-scale samples are added. STA algorithm performs well in the face of a large number of obstacles, because the influence of obstacles on the algorithm is only in the construction stage of cost matrix, and has little influence on the evaluation value. Compared with the STA algorithm, the evaluation value obtained by GA algorithm is relatively poor; The heuristic method constructs the solution quickly and can get the solution within 6s, but the method can't get a reasonable solution in half of the cases, and the quality of the solution is also poor. Compared with the gap value, the scene scale is larger and larger, and the relative gap value is also larger. Especially in the C10-T2-O example, the gap value reaches 21.8%. Taking C10-T2-O as an example, the convergence curve of the algorithm is shown in Figure 5. It can be seen from the figure that the initial solution obtained by GA algorithm is often an infeasible and invalid solution, which shows that ignoring the influence of obstacles will have a great impact on this problem. Compared with GA algorithm, the STA algorithm is effective and it has certain advantages in running time and evaluation value.

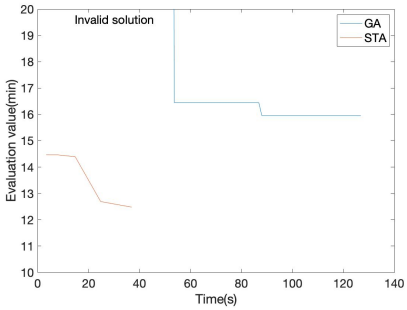


Fig. 5 Time Comparison between GA and STA in C10-T2-O

Table 3 Results of the Computational Experiments (Tour Time(min) T_{RUN} (s))

Instance	GA		HA		STA		
	Tour Time	T_{RUN}	***Tour Time***	T_{RUN}	***Tour Time***	T_{RUN}	Gap (%)
C3-T3-O	**9.6**	0.5	**9.6**	**0.3**	**9.6**	4.5	0.0
C4-T2-S	12.3	**0.5**	12.3	1.1	**11.7**	1.1	4.9
C5-T2-O	14.6	1.3	15.0	**1.9**	**13.8**	6.6	5.5
C6-T3-O	**13.4**	13.1	**13.4**	**2.1**	**13.4**	87.0	0.0
C7-T3-O	15.3	22.6	15.5	**2.8**	**13.8**	12.6	9.8
C8-T4-S	13.3	60.8	15.9	**2.9**	**13.0**	44.5	2.0
C9-T6-O	14.2	54.0	14.3	**4.2**	**13.7**	19.5	3.7
C9-T7-S	11.7	67.0	**11.3**	**3.8**	**11.3**	69.6	0.0
C10-T5-S	15.0	101.0	16.3	**5.7**	**14.8**	11.8	1.1
C10-T6-S	17.7	62.7	17.2	**5.5**	**17.2**	70.0	0.3

Notes:~ means that the optimal solution appears in the initial population, so the time is meaningless; NAN represents that the access order found is not realizable; Evaluation value UAV tour cost time (min), Running time T_{RUN} (s)

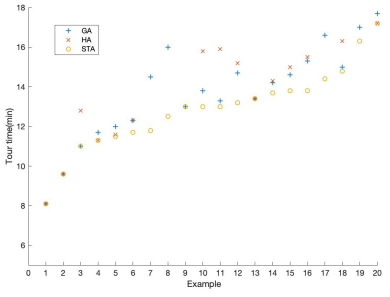


Fig. 6 Time Comparison between GA and HA and STA

In the Table 3, the speed range of the UGVs is reduced, and some tests are done in another road network structure. Experiments show that STA algorithm is still effective when the speed of UGVs is small under different road network structures. However, compared with table 2, the average GAP in Table 2 is 7.42%, and the average GAP in Table 3 is 2.73%. The GAP value of STA algorithm is quite small, which may be because the deceleration of UGVs has a certain impact on the tour time. Figure 6 shows that STA algorithm has certain applicability under different examples, and has better effect than the comparison algorithm.

5. CONCLUSION

Aiming at the problem of limited communication between urban combat environment and UGV platforms, UAV is introduced to play the role of "messenger" in the cooperative system to help solve the communication problem through large-scale motion. We model the UAV path planning problem as a special dynamic generalized traveling salesman problem (DGTSP), and design a sampling transformation algorithm. Through the discrete sampling transformation of the UGV path, we skillfully transform the problem into an ATSP problem that can be solved directly by the solver. Finally, the effectiveness of the algorithm is verified by several groups of experiments.

In the future, mission planning and path planning under the cooperation of multi messenger UAV will be studied.

Acknowledgements

This work was supported by the National Outstanding Youth Talents Support Program under Grant 61822304 and the Basic Science Center Programs of NSFC under Grant 62088101.

REFERENCES:

[1] Fantana N L, Riedel T, Schlick J, et al. Internet of Things - converging technologies for smart environments and integrated ecosystems, 2013.

[2] Hao Zhang, Bin Xin and Yulong Ding, “Online path planning of messenger UAV in air-ground collaborative system,” 2019 Chinese Control Conference (CCC), 2019, 5875-5880.

[3] M.Saska, V.Vonasek, T.Krajnik, and L.Preucil, “Coordination and navigation of heterogeneous MAV-UGV formations local-ized by a ‘hawk-eye’-like approach under a model predictive control scheme, ” International Journal of Robotics Research, 2014, 33(10): 1393–1412.

[4] Qingqing Wu, Yong Zeng, Rui Zhang. “Joint trajectory and communication design for Multi-UAV enabled wireless networks,” IEEE Transactions on Wireless Communications, 2018, 17(3):2109-2121.

[5] Yulong Ding, Bin Xin, Jie Chen. “Path planning of messenger UAV in air-ground coordination,” In Proceedings of the 20th IFAC World Congress, 2017, 8045–8051.

[6] Jie Chen, Xing Zhang, Bin Xin. “Coordination between unmanned aerial and ground vehicles: a taxonomy and optimization perspective," IEEE Transactions on Cybernetics, 2017, 46(4): 959-72.

[7] Chase C. Murray Ritwik Raj. “The multiple flying sidekicks Traveling Salesman Problem: parcel delivery with multiple drones,” Transportation Research Part C: Emerging Technologies, 2020, 110:369-398.

[8] Sung Hoon Chung,Bhawesh Sah,Jinkun Lee. “Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions,” Computers and Operations Research, 2020, 123.

[9] Yulong Ding, Bin Xin, Jie Chen. “Precedence-constrained path planning of messenger UAV for air-ground coordination,” Control Theory Tech, 2019, 17(1):13-23.

[10] Charles E. Noon, James C. Bean.“An efficient transformation of the generalized Traveling Salesman Problem,” INFOR: Information Systems and Operational Research, 1993, 31(1):139-44.

[11] Keld Helsgaun. “An effective implementation of the Lin–Kernighan traveling salesman heuristic,” European Journal of Operational Research,2000,126(1):206-130.