

## Article

# Cooperative Multi-UAV Conflict Avoidance Planning in a Complex Urban Environment

Kaiping Wang <sup>1</sup>, Mingzhu Song <sup>2</sup> and Meng Li <sup>1,\*</sup><sup>1</sup> Department of Civil Engineering, Tsinghua University, Beijing 100084, China; wkp15@mails.tsinghua.edu.cn<sup>2</sup> Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Shenzhen 518055, China; smz15@mails.tsinghua.edu.cn

\* Correspondence: mengli@tsinghua.edu.cn

**Abstract:** Trajectory planning is of great value and yet challenging for multirotor unmanned aerial vehicle (UAV) applications in a complex urban environment, mainly due to the complexities of handling cluttered obstacles. The problem further complicates itself in the context of autonomous multi-UAV trajectory planning considering conflict avoidance for future city applications. To tackle this problem, this paper introduces the multi-UAV cooperative trajectory planning (MCTP) problem, and proposes a bilevel model for the problem. The upper level is modeled as an extended multiple traveling salesman problem, aiming at generating trajectories based on heuristic framework for multi-UAV task allocation and scheduling and meanwhile considering UAV kinodynamic properties. The lower level is modeled as a holding time assignment problem to avoid possible spatiotemporal trajectory conflicts, where conflict time difference is analyzed based on the proposed state-time graph method. Numerical studies are conducted in both a 1 km<sup>2</sup> virtual city and 12 km<sup>2</sup> real city with a set of tasks and obstacles settings. The results show that the proposed model is capable of planning trajectories for multi-UAV from the system-level perspective based on the proposed method.



**Citation:** Wang, K.; Song, M.; Li, M. Cooperative Multi-UAV Conflict Avoidance Planning in a Complex Urban Environment. *Sustainability* **2021**, *13*, 6807. <https://doi.org/10.3390/su13126807>

Academic Editor: Tomio Miwa

Received: 26 April 2021

Accepted: 10 June 2021

Published: 16 June 2021

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



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

**Keywords:** multi-UAV; cooperative trajectory planning; conflict avoidance; holding time

## 1. Introduction

In recent years, unmanned aerial vehicles (UAVs) have been a crucial component of future delivery, service patrol, and emergency response applications. Trajectory planning is of great importance in multi-UAV applications. As an urban aerial application, an automated multi-UAV system is able to provide fast delivery services regardless of the extensive workload. In many scenarios, the multi-UAV system needs to cooperate to complete multiple tasks [1,2]. Tasks can occur anywhere in urban environments, such as landing on a charging station on top of the building or hovering for traffic monitoring. UAV charging needs to be considered in planning. By integrating charging stations, the service radius for multi-UAV systems can be considerably expanded with proper optimization [3]. Setting automatic charging stations on building roofs has been discussed in many cutting-edge studies; for this reason, the organization and management of multi-UAV systems are more complicated system-level optimization problems that require efficient trajectory planning and optimization tactics for practical applications. This problem involves trajectory planning and conflict avoidance.

Single-UAV trajectory planning is of great value for time-sensitive urban aerial applications [4–9] due to the unreliability of road traffic conditions. Given the origin and destination in 3D space, trajectory planning aims to find a feasible path with the motion plan while avoiding obstacles. Compared to 2D cases, trajectory planning in 3D urban space is much more complicated. First, there are infinite feasible trajectories in a high-dimensional continuous configuration space for UAV motion [10]. Second, the complex urban environment with cluttered buildings further complicates the configuration space.

Existing studies on UAV trajectory-planning problems can be classified into three categories. The sampling-based method, represented by RRT and PRM [11], obtains a feasible trajectory by probing random points in the space. Heuristic-based methods, such as the genetic algorithm (GA) and particle swarm optimization (PSO) algorithm [12], are capable of incorporating UAV dynamics in the modeling, but the computational cost is expensive. The two-stage-based method [13], under a complex 3D urban environment, has difficulties guaranteeing efficiency due to its heuristic framework. Due to this reason, a control network approach is proposed in our prior work [14] for efficient UAV trajectory planning in a complex urban environment. The key idea of the control network is to reduce the complexity of the trajectory-planning problem by decomposing to a discrete network routing problem, a path smoothing problem, and a time-optimal trajectory optimization problem. The numerical studies show that the proposed approach is able to identify high-quality trajectories within negligible computational time.

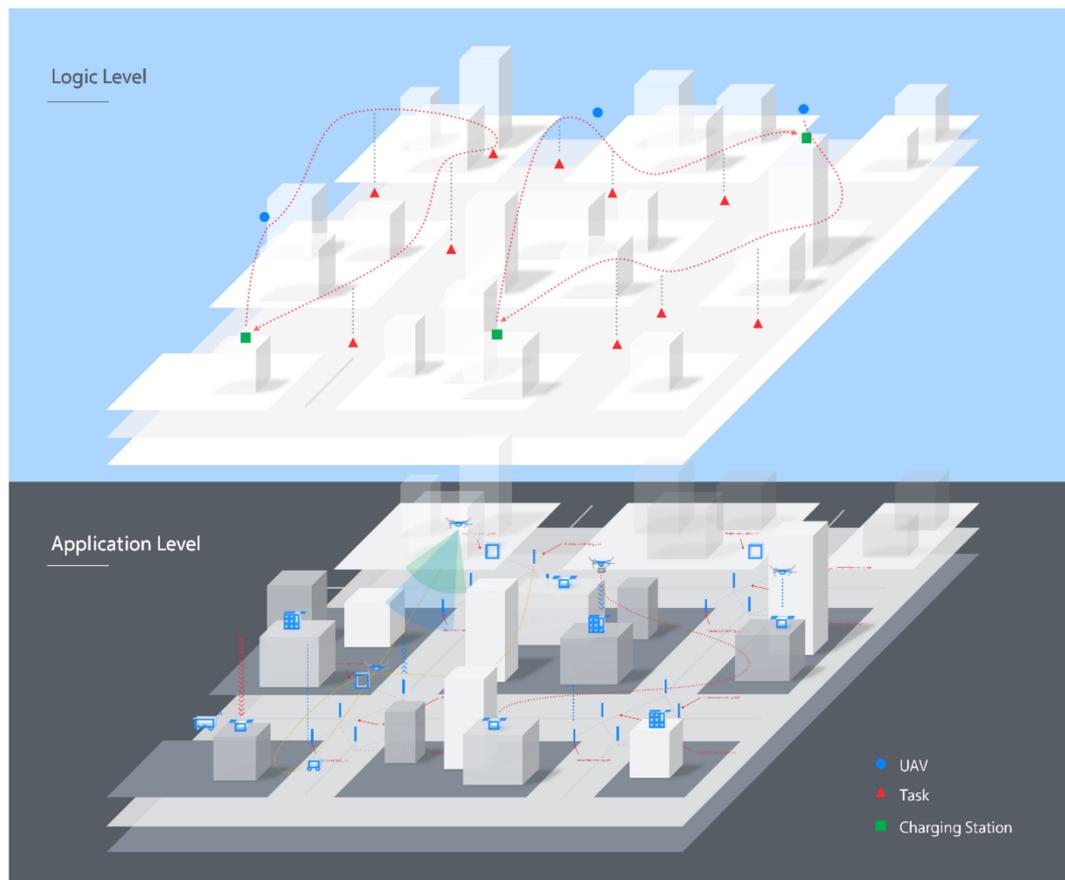
For multitask applications, the classic traveling salesman problem (TSP) solves the single-UAV task scheduling and trajectory-planning problem. Multi-UAV trajectory planning (MTP) extends single-UAV trajectory planning by optimizing task allocation in a complex urban environment. Under the premise of limited time of endurance, MTP has raised many research interests in recent years. Rasche et al. adopted the potential field theory for MTP in a 3D environment to coordinate the UAV according to the target area and obstacles [15]. Ergezer et al. [16] proposed a genetic algorithm to solve the mTSP problem of multi-UAV systems to avoid flying over restricted areas. Moon et al. [17] implemented a layered framework of task assignment and path planning for multi-UAV systems in a dynamic environment. The framework is based on the shortest path principle and the potential field method to achieve obstacle avoidance. Ergezer et al. [18] introduced a new heuristic operator in a genetic algorithm to generate paths based on the UAV dynamic model. However, there are few studies on avoiding flight route conflicts of multiple UAVs.

The trajectory is regarded as a polyline passing through a series of nodes and links in the control network. Multi-UAV trajectory conflict can be regarded as node collision and edge collision at the same time. In conflict avoidance studies, to maintain a minimum safe distance between airplanes and UAVs, Persiani et al. [19] and Furini et al. [20] modeled the routing and holding time of UAVs as a time-dependent traveling salesman planning problem (TDTSP) and proposed a heuristic algorithm to minimize the operation cost. Multiple trajectories with obstacle avoidance can be found in discrete time and space by using sparse roadmaps with annotations of possible collisions between robots [21]. Conflict-based search (CBS) using the structure of graph search and conflict trees is an effective trajectory-planning method [22,23]. It first finds the optimal paths ignoring conflicts. If the paths conflict, then constraints on the conflicting robots are imposed through a splitting operation to avoid these conflicts. Araki et al. [24] proposed a priority planning method integrating safe interval path planning (SIPP) and integer linear programming (ILP) to solve multimode collision-free path planning. The idea of robot priority planning is introduced in multirobot trajectory planning to find a nonoptimal but fast centralized solution [25,26]. The graph-based method [27,28], TDTSP [19,20], CBS [22,23], and SIPP [24] are all able to avoid conflicts; however, when the graph size or the number of robots exceeds certain limitations, the complexities grow exponentially, which makes it impossible to solve. In addition, these studies are based on designated paths of polylines with an average speed, without considering the motion plan of UAV flying states, which is crucial in UAV spatiotemporal trajectory planning.

To fill this research gap, this paper investigated conflict-free multi-UAV cooperative trajectory planning (MCTP) problems (shown in Figure 1). It provides the following contributions:

1. This paper proposed a bilevel model for the MCTP problem. The upper level optimizes the task allocation and sequencing. The lower level solves the holding time assignment problem, thereby constructing a conflict-free multi-UAV system operation strategy.

2. By extending prior work on the efficient control network approach [14], practically applicable time-optimal trajectories are generated based on an optimized task visiting route plan in an upper-level model.
3. A state-time graph method for conflict detection is proposed in the lower level for holding time assignment.
4. Numerical experiment are discussed in both a 1 km<sup>2</sup> virtual city and 12 km<sup>2</sup> real city. An optimized system operational strategy is discussed for conflict-free automated multi-UAV services in a complex urban environment.



**Figure 1.** Multi-UAV cooperative trajectory planning (MCTP).

The rest of the paper is organized as follows. The problem definition is described in Section 2. Section 3 presents the proposed bilevel model in two parts: the upper level optimizes multi-UAV trajectory planning based on a hybrid heuristic framework, and the lower level solves conflict avoidance holding time assignment. Section 4 presents the numerical experimental results for the MCTP. Finally, Section 5 gives a summary of this paper and identifies directions for future research.

## 2. Problem Statement

The following problem description and definition are given for the NP-hard MCTP problem based on an automated multi-UAV system:

The multi-UAV system is designed for the autonomous execution of multiple tasks. Tasks can occur anywhere in urban environments, such as landing on a charging station on top of a building or hovering for traffic monitoring. Each task is associated with a predefined work time for task execution. A multi-UAV system needs  $m$  UAVs with associated  $m$  charging stations. Suppose each UAV is able to recharge or replace the battery at any charging station. That is, UAVs can take off stations or tasks, and there is no need to

return to the first take-off station if there is a vacant charging station nearby. One charging station is able to preserve one UAV at a time, but the UAV can choose to land on any charging station. In other words, there is no one-to-one restriction between UAVs and charging stations in the problem setting.

The flight of multiple UAVs among the geolocations of tasks and charging stations can be defined by the UAV state (Section 2.1), and spatiotemporal trajectory conflicts may occur in the flight state (Section 2.2).

### 2.1. UAV State Definition

Different tasks can be defined in the form of  $(i, x, y, z, t_k^{i,w})$ , where  $i$  indicates the task index and  $(x, y, z)$  denotes the coordinates of the task waypoint that correspond to the node in the control network.  $t_k^{i,w}$  in Table 1 indicates the work time cost for UAV  $i$  at task  $k$ , which corresponds to real-world task situations such as monitoring tasks, video recordings, battery replacement at charging stations, cargo loading time, etc.

In other words, based on the establishment of a control network, the optimal flight time between any tasks and the predefined work time can be considered known and fixed. In addition to the flying state and working state, the holding state is defined as the hovering state of waiting after task execution. A state-time graph is then established for multi-UAV operations that are performed in accordance with the time constraints, as shown in Figure 2. Each flight starts from a task waypoint. Each flight ends at a task waypoint. The path generation in the form of polylines consists of multiple waypoints. To ensure speed continuity, the holding time is only assigned to the task-associated waypoint.

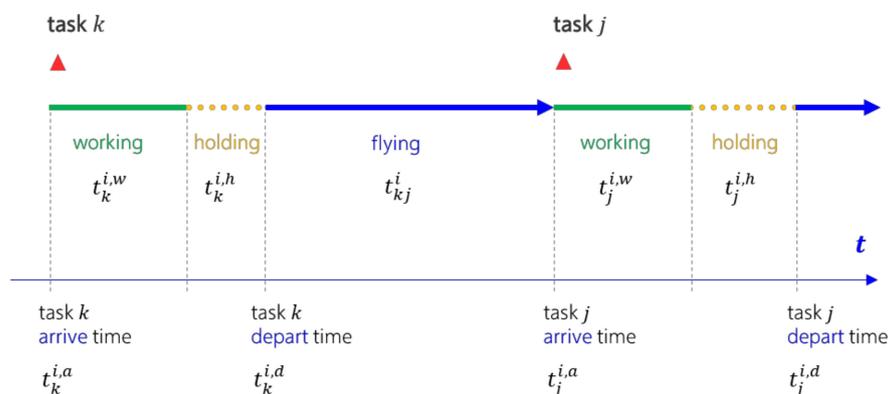


Figure 2. UAV state-time graph.

Table 1. UAV state-time graph denotation.

Symbol	Explanation
$t_k^{i,a}$	arrive time of UAV $i$ at task $k$
$t_k^{i,w}$	work time of UAV $i$ at task $k$
$t_k^{i,h}$	after work state of UAV $i$ at task $k$ , holding time is assigned at current task $k$ , to avoid possible trajectory conflict of future flight
$t_k^{i,d}$	depart time of UAV $i$ at task $k$
$t_{kj}^i$	flight time of UAV $i$ from task $k$ to task $j$

### 2.2. Conflict Definition

Three terminologies used in this paper are defined: route refers to a set of consecutive links in the control network; path refers a spatial curve plan that the UAVs follow; and trajectory is a spatiotemporal concept to represent the time-space curve planned for UAVs. The multi-UAV trajectory is regarded as a 3D polyline passing through a series of nodes and links. The trajectory conflict between two UAVs can be regarded as node conflict and

edge conflict at the same time. Since the flight polyline has multiple links and nodes, it further increases the complexity of conflict detection, which is a 3D problem.

Time is discretized in a time step of  $dt$  s, where  $timesteps = t_{max}/dt$ ;  $dt = 10$  s. At each timestep, the UAV is either in the working state or holding state at the waypoint or in the flying state. For multi-UAV conflict detection, the following rules are defined:

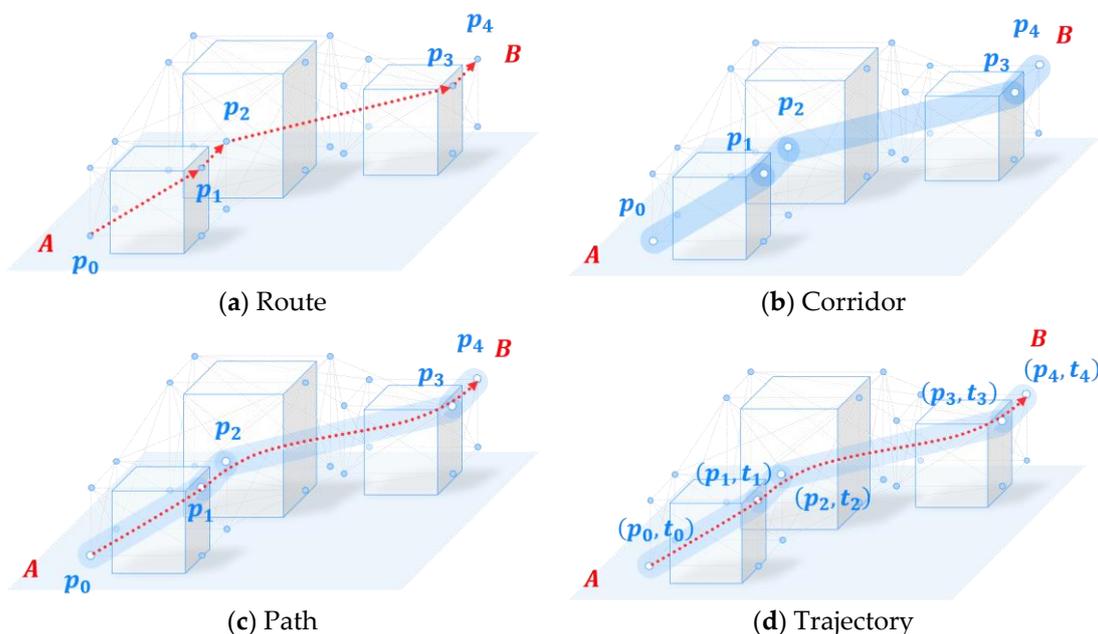
- (a) No two UAVs are on the same task waypoint at the same timestep.
- (b) No two UAVs are too close (within a Euclidean distance threshold in 3D space) while moving along the path at the same timestep.

### 3. Methodology

In this part, firstly Section 3.1 describes our prior work of trajectory-planning approach based on control network [14], which is incorporated in the MCTP problem modeling. Secondly, Section 3.2 details proposed bilevel model framework for cooperative multi-UAV conflict avoidance planning. The solution approach corresponding to Section 3.2 is presented in Section 3.3.

#### 3.1. Control-Network-Based Trajectory Planning

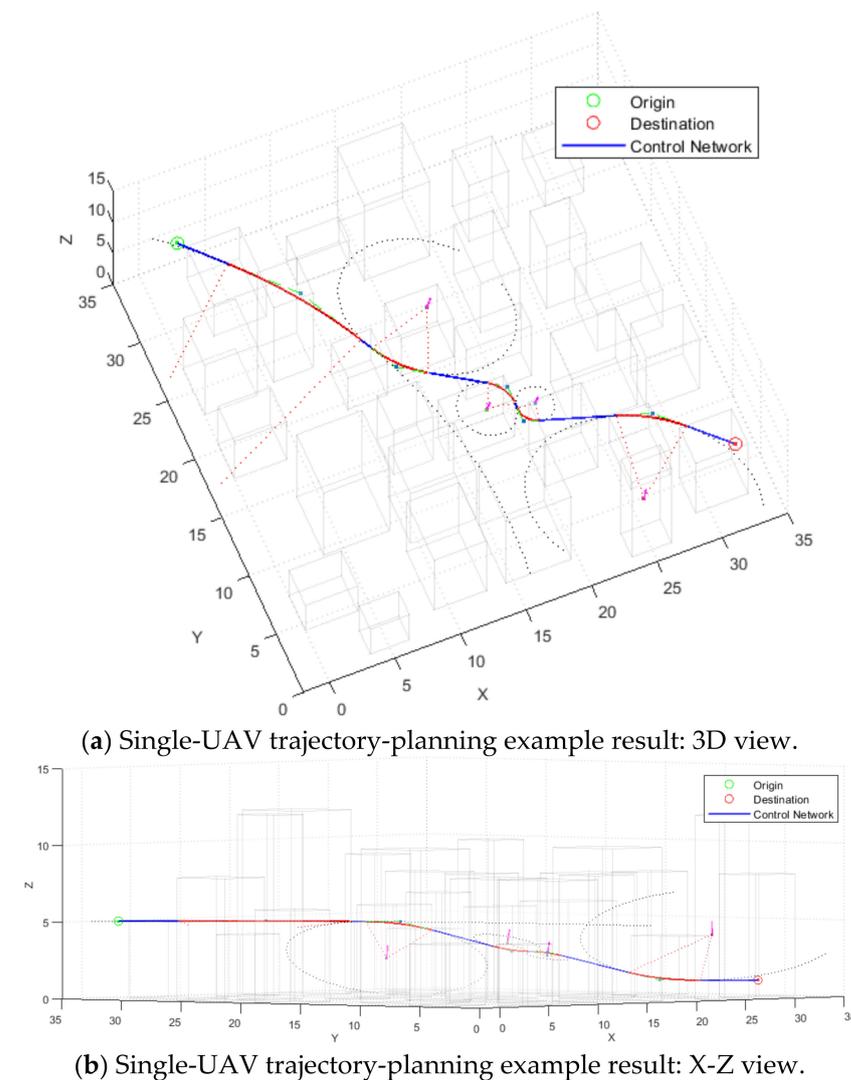
Control network approach [14] is integrated for efficient single-UAV trajectory planning in a complex urban environment. We first give definitions of route, corridor, path, and trajectory. Route in the control network is a feasible polyline that passes by a series of control points and links (Figure 3a). An unobstructed corridor can be constructed based on the valid route (Figure 3b). Path represents a 3D curve smoothed on the basis of route within the corridor (Figure 3c). A trajectory is a path with time dimension representing the motion plan of UAV (Figure 3d).



**Figure 3.** Route, corridor, path, and trajectory definition in control network approach.

The key idea of control network is to decompose a large-scale infinite-dimensional nonlinear optimization problem into a discretized network routing problem, a curve smoothing problem, and a time-optimal trajectory design problem given a fixed path, and the latter three problems were all substantially easier than the original one. The control network is constructed based on 3D building map. In order to express different fly states and strategies, the control points are then expanded while considering kinodynamic constraints. The extended 3D Dubins path-smoothing and path-following algorithms are

then performed within the safe flight corridor for time-optimal trajectory generation (see Figure 4).

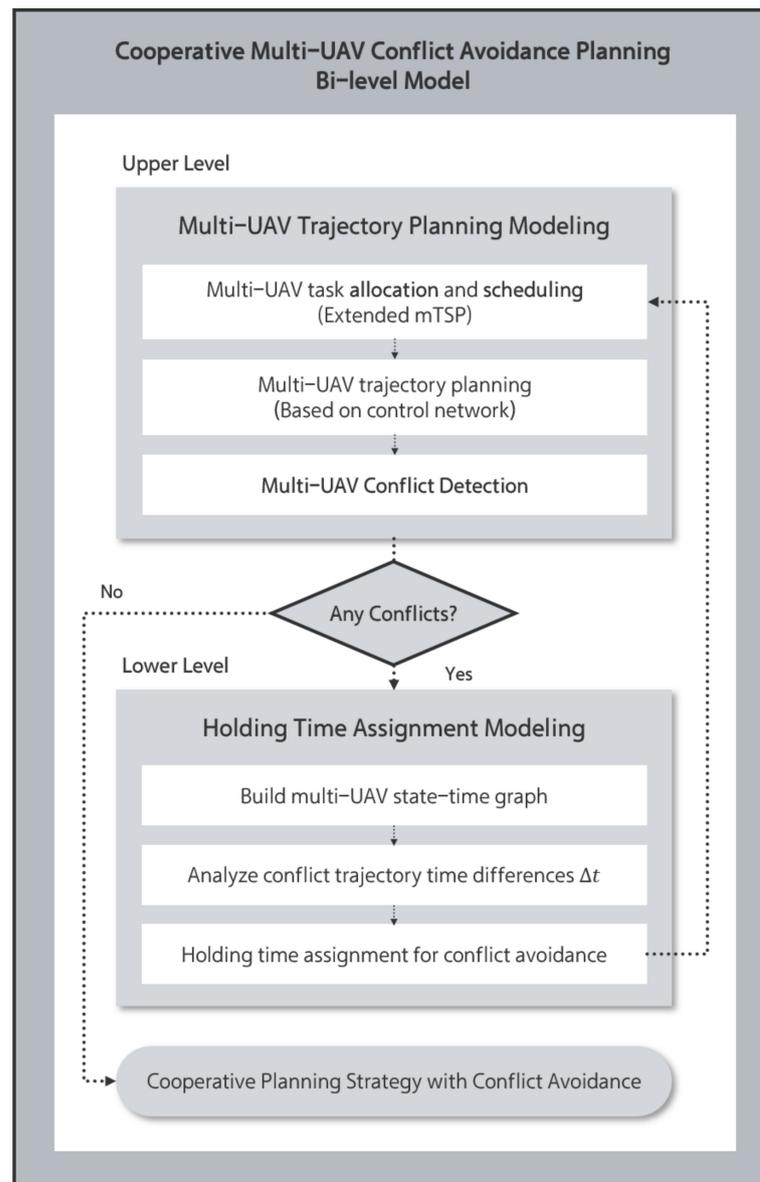


**Figure 4.** Trajectory-planning example based on control network in a complex 3D space.

The time-optimal trajectory generated by control network approach allows the UAV to fly at approximately the maximum speed. Therefore, the optimized trajectory can achieve speed-time continuity.

### 3.2. Multi-UAV Cooperative Trajectory Planning

Aiming at multi-UAV system optimization and conflict avoidance for regional multitask execution, we propose a bilevel model (as shown in Figure 5) that is defined as multi-UAV cooperative trajectory planning (MCTP) to tackle this problem. The upper level is a genetic algorithm, which optimizes multi-UAV task assignment and scheduling and generates time-optimal trajectories based on a control network approach. In the lower-level algorithm, we proposed a multi-UAV task holding time assignment strategy based on MILP for conflict avoidance.



**Figure 5.** Methodology framework.

### 3.2.1. Upper Level: Multi-UAV Trajectory Planning (MTP) Modeling

Based on the prementioned multi-UAV system constraints, the multi-UAV cooperative trajectory planning (MCTP) model is proposed based on the extended mTSP problem to tackle the spatiotemporal conflict of trajectories caused by multi-UAV trajectory planning. In the upper level, the main task is to optimize task allocation and task scheduling in the framework of a genetic algorithm. Conflict detection is conducted in each iteration to decide whether the lower-level algorithm of conflict avoidance should be performed. Systematic optimization for the MTP model has optimization objectives on operational costs, total flight distance, total flight time, and total holding time.

$$F = f_1 \times q_1 + f_2 \times q_2 + f_3 \times q_3 + f_4 \times q_4 \quad (1)$$

$$f_1 = \min \left( \sum_{i=1}^m S_i \right); i = 1, 2, 3 \dots m \quad (2)$$

$$\begin{aligned}
 S_i &= \begin{cases} 1, & \text{station } i \text{ is allocated with task} \\ 0, & \text{otherwise} \end{cases} \\
 f_2 &= \min \left( \sum_i^m \sum_k^n \sum_j^n R_{kj}^i \times d_{kj} \times S_i + \sum_i^m \sum_a^m \sum_j^n Q_{aj}^i \times d_{aj} \times S_i \right) \\
 R_{kj}^i &= \begin{cases} 1, & \text{UAV } i \text{ fly from task } k \text{ to task } j \\ 0, & \text{otherwise} \end{cases} \\
 Q_{aj}^i &= \begin{cases} 1, & \text{UAV } i \text{ fly from station } a \text{ to task } j \\ 0, & \text{otherwise} \end{cases}
 \end{aligned} \tag{3}$$

where  $q_1, q_2, q_3, q_4$  are the weights of the objectives in the multiobjective optimization problem.  $m$  is the station number, and  $n$  is the task number.  $d_{kj}$  indicates the distance between task  $k$  and task  $j$ , and  $d_{aj}$  is the distance between station  $a$  and task  $j$ .

$$f_3 = \min \left( \max \left( t^i \times S_i \right) \right); \tag{4}$$

$$\begin{aligned}
 t^i &= \sum_{j=1}^n t_j^d \times D_j^i + \sum_a^m \sum_j^n Q_{aj}^i \times t^c + \sum_k^n \sum_j^n R_{kj}^i \times t_{kj}^i \times S_i + \sum_a^m \sum_j^n Q_{aj}^i \times t_{aj}^i \times S_i; \\
 & i \in 1, 2, 3 \dots m;
 \end{aligned} \tag{5}$$

$$D_j^i = \begin{cases} 1, & \text{task } j \text{ is allocated to UAV } i \\ 0, & \text{otherwise} \end{cases}$$

where  $t^i$  is the total time for UAV  $i$  to complete all allocated tasks.  $t_j^d$  is the work time cost on task  $j$ .  $t^c$  is the time cost for UAV battery charging or replacement.  $t_{kj}^i$  is the flight time cost for UAV  $i$  from task  $k$  to task  $j$ , and  $t_{aj}^i$  is the flight time cost for UAV  $i$  from station  $a$  to task  $j$ . It should be noted that the flight times here, namely,  $t_{kj}^i$  and  $t_{aj}^i$ , are estimated minimum flight time [14].

Based on the task allocation and scheduling results of upper-level model, multi-UAV trajectories are obtained in the form of smoothed spatiotemporal 3D curves, each passing by a series of control points (nodes) and links. Therefore, the trajectory conflict between two UAVs can be regarded as node conflict and link conflict. Trajectory conflict detection is then performed in each iteration. Given same start time for multi-UAV, if no conflict is detected, then the upper-level model gives optimal conflict-free multi-UAV trajectory-planning results based on given tasks. If conflict is detected, the lower level of the model steps in to resolve conflict by assign holding time to certain UAVs at certain tasks.

### 3.2.2. Lower Level: Holding Time Assignment Modeling

The key idea of conflict-free MCTP is to assign holding time to the UAV after task execution; if properly coordinated, the potential conflict trajectory between UAVs can be postponed and resolved to achieve a systematic optimization objective on the MCTP problem. The holding time assignment problem is proposed for conflict-free minimum holding time optimization.

To avoid possible multi-UAV conflicts without replanning the optimal time trajectory, holding time is introduced for the route plan. The holding time adds time gaps of different durations to the task to stagger the planned conflicting flight trajectories. Assigning a holding time to all tasks is the key point in conflict avoidance-based MCTP problems. The holding time settings are different in different tasks. It can be set to zero if no conflict is predicted for the next flight. In other words, if there is no conflict, the UAV can fly directly to the next task after the task work is completed.

We propose mixed-integer linear programming (MILP) formulations for the conflict avoidance holding time assignment problem. The objective is to minimize the total holding time based on conflict avoidance.

$$f_4 = \min \sum_{i \in M} \sum_{k \in N} t_k^{i,h} \tag{6}$$

$$t_j^{i,d} \geq t_{kj}^i + t_k^{i,d} + t_j^{i,w} \tag{7}$$

$$\left| t_j^{u,d} - t_k^{i,d} - \Delta t_{kj} \right| \geq 1 \tag{8}$$

$$i, j \in M; M = \{1, 2, 3 \dots m\}$$

$$t \in T; T = \{1, 2 \dots, t_{max} / \Delta t\}; \Delta t = 10$$

$$k, j \in N; N = \{1, 2 \dots, n\}$$

Equation (7) is a constraint reflecting the state-time graph. The departure time of UAV  $i$  at task  $j$  should be greater than or equal to the departure time  $t_k^{i,d}$  of UAV  $i$  left previous task  $k$ , the flight time  $t_{kj}^i$  of UAV  $i$  from task  $k$  to task  $j$ , and the work time  $t_j^{i,w}$  of UAV  $i$  at task  $j$ , where holding time  $t_j^{i,h} \geq 0$ . Equation (8) is a constraint for trajectory conflict avoidance. For two tasks, calculate the departure time difference  $\Delta t$  and compare with the conflict time difference set  $T$  to meet the conflict constraint. The time difference between the departure time  $t_j^{u,d}$  of UAV  $u$  at task  $j$  and the departure time  $t_k^{i,d}$  of UAV  $i$  at task  $k$  does not belong to the conflict time difference set  $\Delta t \notin T$ .

### 3.3. Solution Approaches to MCTP

#### 3.3.1. Heuristic Framework

We extend GA for MTP problem solving by using multilayer real number coding and the corresponding genetic operator.  $n$  tasks and  $m$  UAVs are encoded in real number by using a two-layer coding method: the first layer represents task scheduling, and the second layer represents task allocation. Taking five tasks and two UAVs as an example, as shown in Figure 6, tasks allocated to UAV1 are {3, 4, 5, 6} and to UAV2 are {2, 1, 7}.

1 <sup>st</sup> layer	<b>task scheduling</b>	2	1	7	3	5	4	6
2 <sup>nd</sup> layer	<b>task allocation</b>	2	1	2	1	1	2	1

<b>uav1 task</b>	1	3	5	6	1
<b>uav2 task</b>	2	7	4	2	

Figure 6. Chromosome multilevel coding.

In MTP, a two-point crossover method is adopted by dividing the chromosome into two parts, and the crossover is performed to correct task scheduling. For example, when the crossover bit is after the third bit, each chromosome has two duplicate bits, and the duplicates are corrected according to the scheduling (as shown in Figure 7). No correction is needed. A two-point mutation is applied here, and two genes are selected for replacement mutations, i.e., In Figure 8, the 2 and 5 bits are exchanged to obtain new task scheduling. The second bit is selected for mutation to obtain new task allocation.

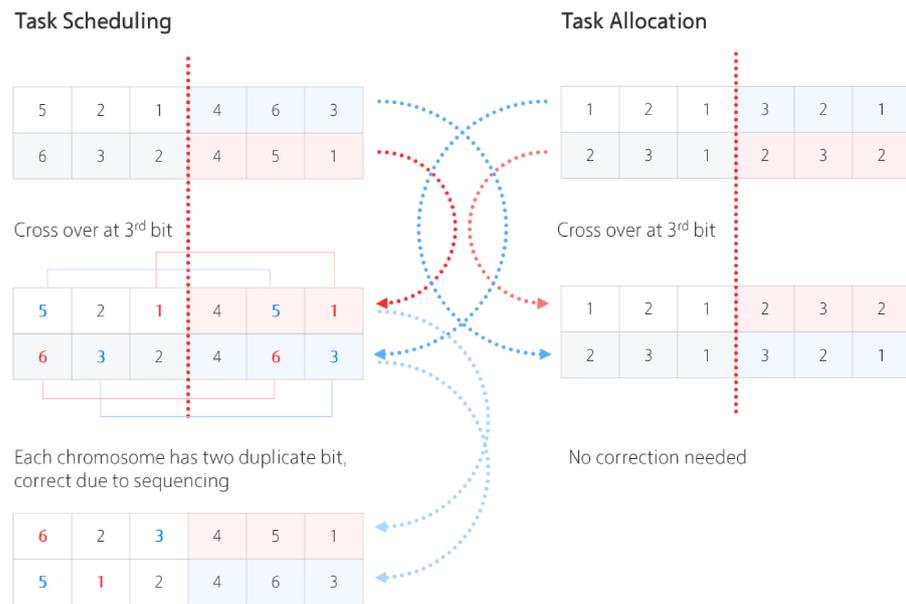


Figure 7. Crossover.

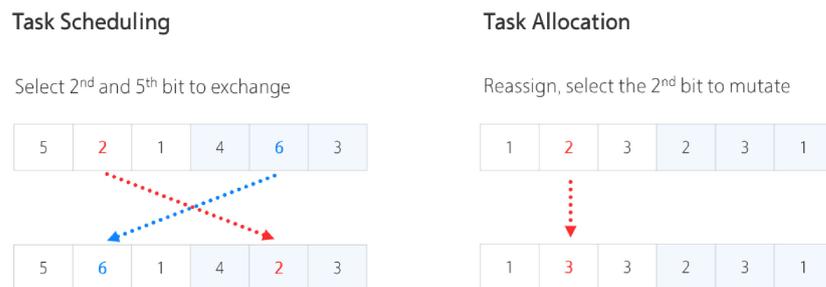


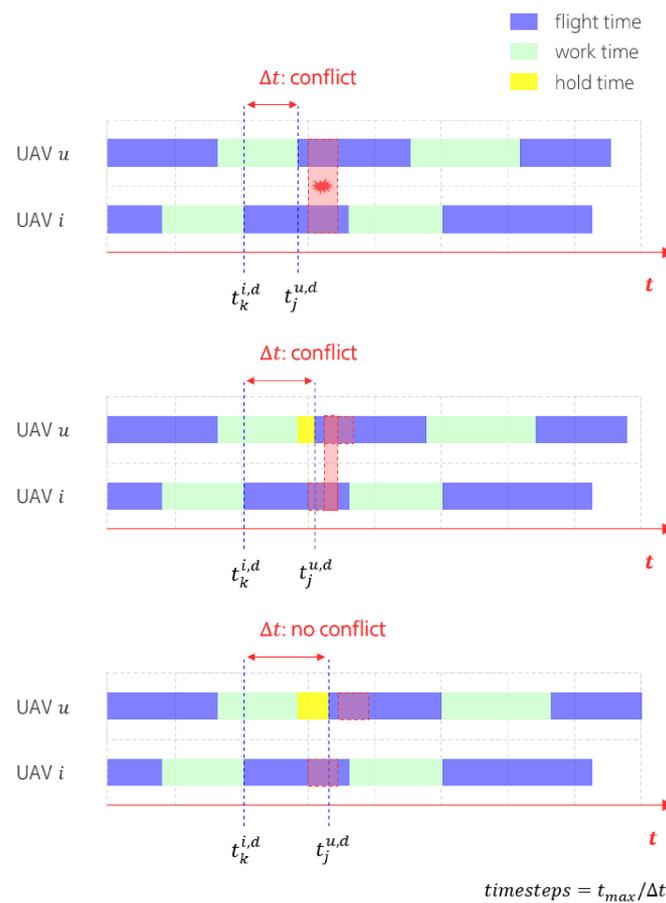
Figure 8. Mutation.

### 3.3.2. Holding Time Assignment Problem

In the second level of the model, to address the MCTP problem with spatiotemporal conflict caused by multi-UAVs that fly on the control network, a holding time assignment problem is introduced by adding a small period after each task execution. The holding time optimization problem is modeled based on MILP for all tasks holding pattern optimization and, most importantly, conflict avoidance. We propose a task trajectory comparison method based on departure time differences that lead to conflicts as references. Based on this strategy, potential task conflicts between UAV tasks can be easily identified and resolved by assigning a conflict-free time difference as the holding time to achieve a systematic optimization objective on a multi-UAV trajectory-planning problem. The lower-level algorithm procedure is shown in Table 2.

To examine conflict based on their departure time difference, we compare the assigned tasks of every two UAVs. If task  $k$  and task  $j$  conflict on the polyline of flight trajectory, then the departure time difference is recorded as  $\Delta t_{kj}$ . Define the difference  $\Delta t$  between the departure time of conflicting task  $k$  and task  $j$  of UAV  $i$  and UAV  $u$ . Define departure time differences set  $\Delta t \in T$  where conflict happens. If there is no conflict between task  $k$  and task  $j$  of UAV  $i$  and UAV  $u$ , then  $\Delta t \notin T$ .

In Figure 9,  $t_k^{i,d}$  denotes the time of departure of UAV  $i$  from task  $k$ ,  $t_j^{u,d}$  is the time of departure of UAV  $u$  from task  $j$ , and  $\Delta t$  is the time gap between  $t_k^{i,d}$  and  $t_j^{u,d}$ . The time difference between departure times can represent tasks with or without conflict.



**Figure 9.** Trajectory conflict with different departure time differences  $\Delta t$ .

**Table 2.** Holding time assignment algorithm procedure.

<b>Step 1</b>	Compare Trajectory between Tasks for Conflicts.
<b>Step 2</b>	Calculate conflict trajectories between tasks based on their departure time difference. A conflict time difference set is introduced for all flight trajectories between different tasks.
<b>Step 3</b>	Build a holding time assignment algorithm based on MILP.
<b>Step 4</b>	algorithm optimization.
<b>Step 5</b>	Return holding time assignment to upper-level algorithm iterations for fitness function evaluation.

## 4. Numerical Study

### 4.1. Experiment Setup

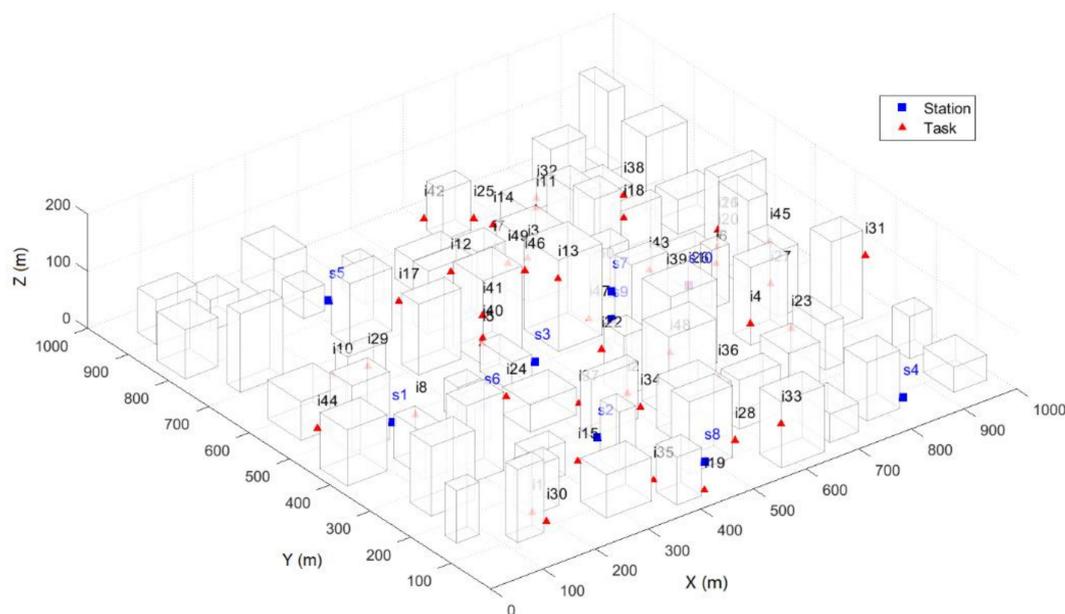
The numerical experiment is based on a computing device with 16 GB of RAM and 1.7 GHz Ryzen 4750u CPU and was programmed with MATLAB R2020a and JAVA 1.8. DJI's Phantom 2 is modeled in the experiment, the main properties are shown in Table 3.

**Table 3.** Properties of DJI's Phantom 2.

Variable	Description	Value
$v_{max}$	Maximum horizontal flight velocity	15 m/s
$v_{up}$	Maximum ascending velocity	6 m/s
$v_{down}$	Maximum descending velocity	2 m/s
$m$	Mass	1 kg
$F_{max}$	Maximum lifting force	16 N
$F_{min}$	Minimum lifting force	9 N

#### 4.1.1. Virtual City

A virtual city of 1 km<sup>2</sup> was constructed with 67 densely distributed buildings. The length and width of the building is within the range of 30–60 m, and the height is within the range of 50–60 m. The control points are set based on the rules of the control network. The corresponding expanded control network contains 33,572 nodes and 7,366,878 links. Fifty nodes are randomly selected from the control network as the locations of the 50 tasks, and then 10 nodes are selected as charging stations based on the p-median algorithm. Considering the various heights required by actual tasks, the heights of these selected points are different. For example, for traffic monitoring, a height of 50 m is required. The spatial distributions of these 55 tasks and charging stations are shown in Figure 10.

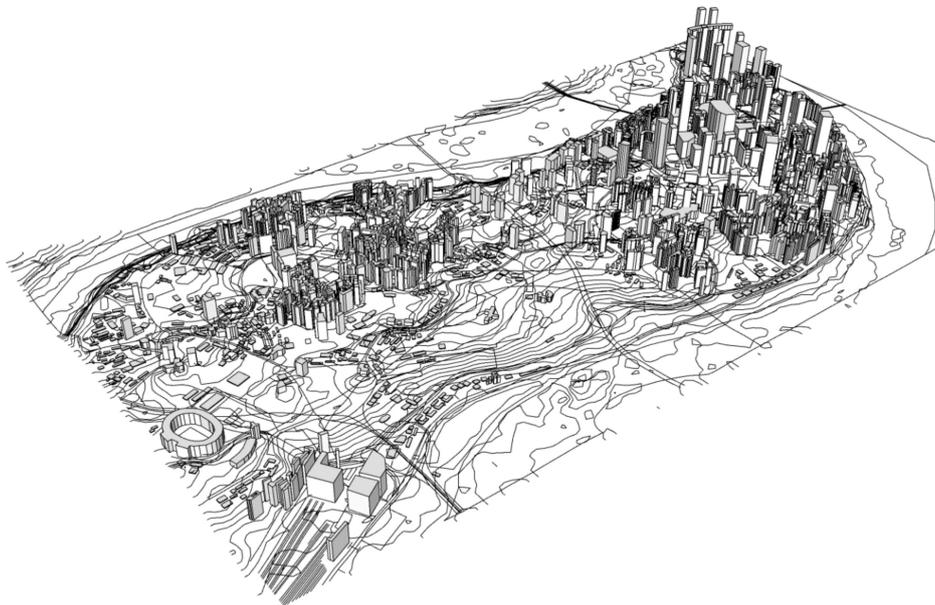
**Figure 10.** Tasks and charging stations distributed in the virtual city.

In this section, dense task distribution is adopted to create multiple possible trajectory conflicts. The cooperative multi-UAV conflict avoidance planning are verified based on the proposed bilevel model. Two cases with multiple trajectory conflicts are studied under same tasks and charging station settings. The difference is that in Case 1, the optimization weight for the objective of the number of UAVs and charging stations  $q_1$  is set to 0.05, and in Case 2,  $q_1$  is set to 0.5. This is to compare the system operation effectiveness and analyze trajectory conflicts based on different multi-UAV fleet sizes.

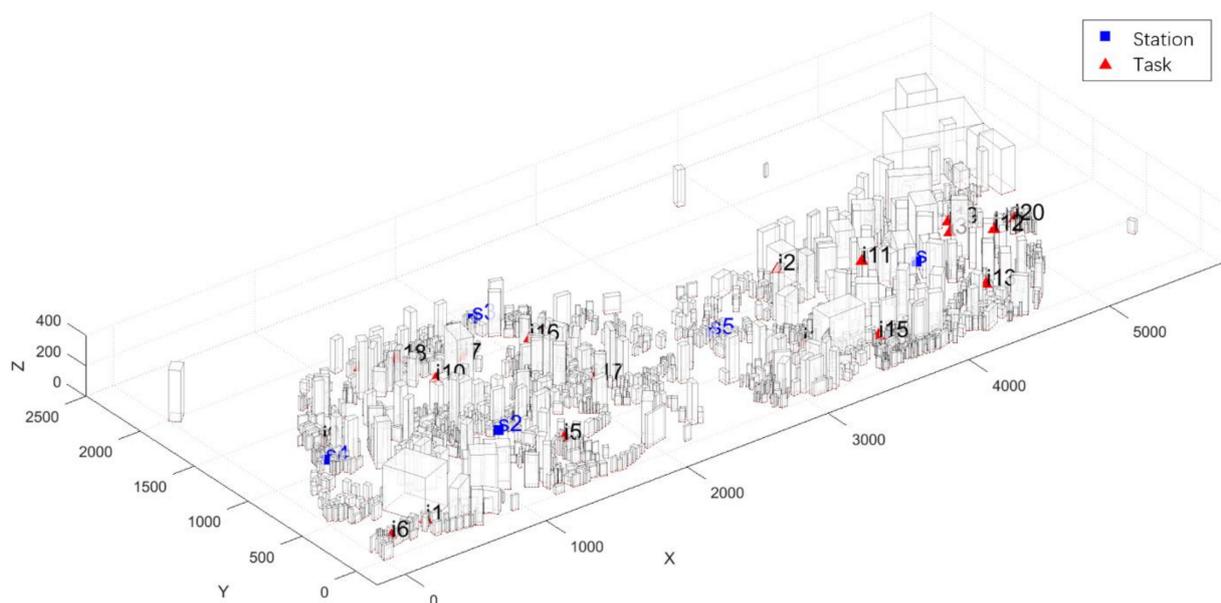
#### 4.1.2. Real City

The Yuzhong Peninsula in Chongqing (shown in Figure 11), China covers an area of 12.4 square kilometers with 1497 buildings, which generates an expanded control network

of 188,545 nodes and 17,892,562 links. The spatial distributions of 20 tasks selected and five charging stations optimized based on p-median are shown in Figure 12.



**Figure 11.** 3D map modeling of Yuzhong Peninsula in Chongqing, China.



**Figure 12.** Tasks and charging stations distributed in Yuzhong Peninsula in Chongqing, China.

In this section, the main purpose is to test the proposed multi-UAV planning model and the effect of UAV endurance in a real urban environment.

#### 4.2. Virtual City MCTP Results and Discussion

As the number of tasks or UAVs increases, conflicts of trajectories are difficult to avoid. In this experiment, two test cases are analyzed for conflict avoidance under the framework of the heuristic bilevel algorithm. In the upper level, multi-UAV trajectory planning is modeled based on the extended mTSP algorithm by a genetic algorithm. The population is set to 200, and the generation of mutations is set to 200. Task allocation and task scheduling

are optimized in the upper level. In each iteration, multi-UAV conflict analysis is conducted based on spatiotemporal trajectories. If there is a conflict between UAVs, the lower-level algorithm is utilized to assign different holding times to UAVs at tasks to avoid conflict. The total holding time at all tasks joins the iteration of GA as one of the optimization objectives.

#### 4.2.1. Case 1 with Eight UAVs

As shown in Figure 13, in this case, eight charging stations and eight UAVs are required to complete these 50 tasks. Based on the optimal results, the total flight distance  $d_{total}$  and flight time  $t_{total}$  of UAVs are shown in Table 4. Taking UAV  $u_1$  as an example, the total flight time of UAV1 is  $t_{total}^1 = 225.67$  s. It takes off from charging station  $s_{51}$  and returns to charging station  $s_{60}$  with eight tasks assigned. The task scheduling is  $\{s_{51}, i_{38}, i_{46}, i_{17}, i_{15}, i_1, i_{48}, i_{34}, i_{26}, s_{60}\}$ .

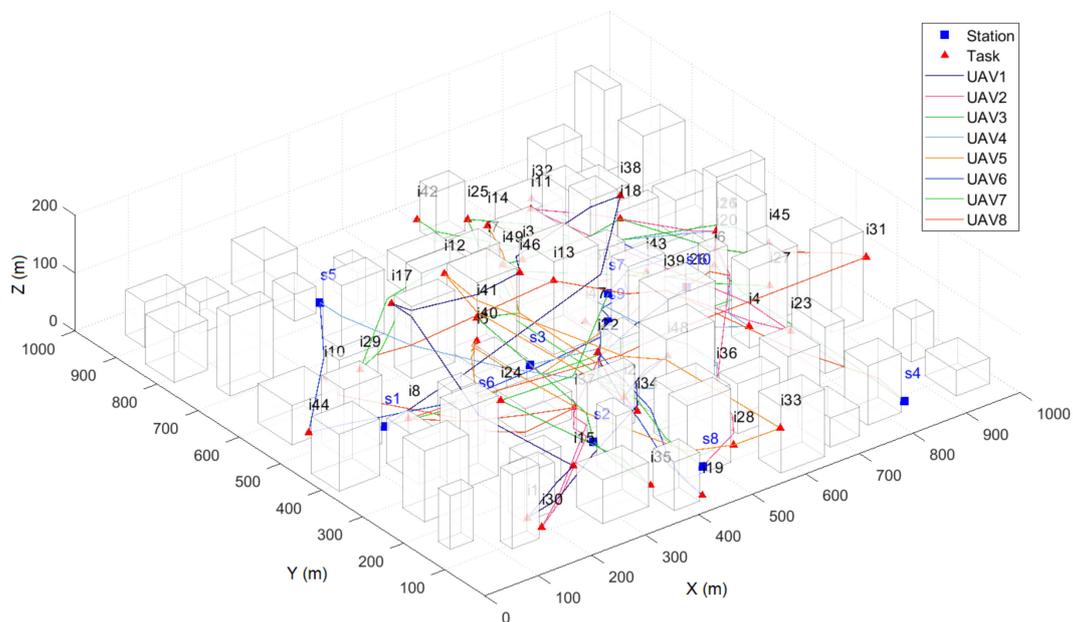
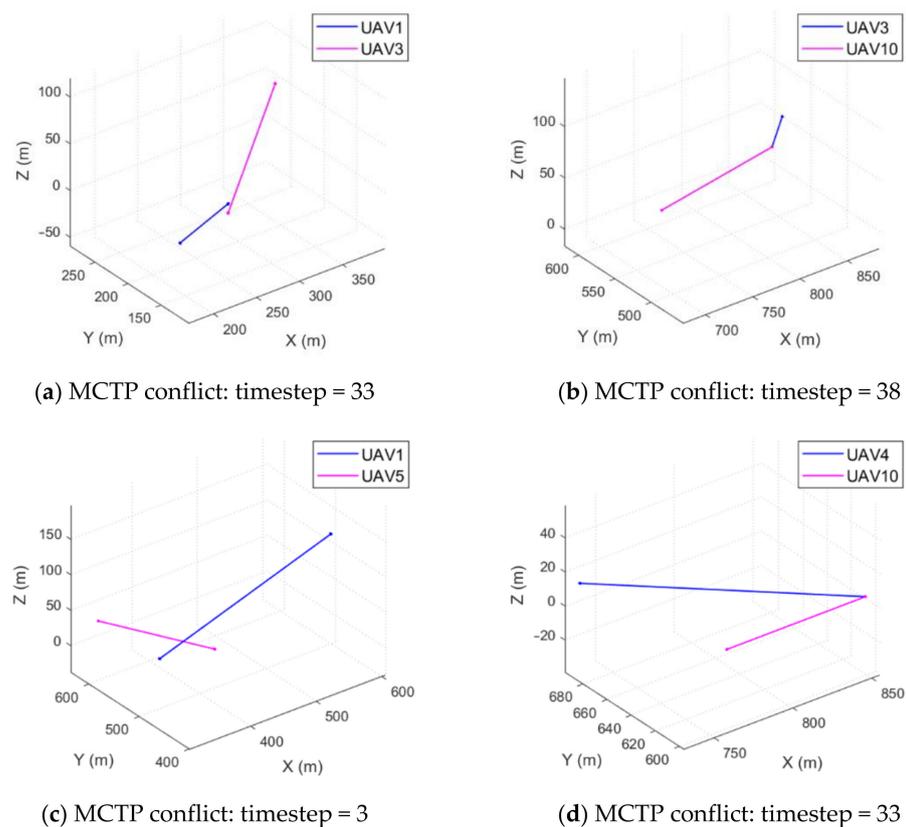


Figure 13. MCTP results of Case 1 in the virtual city.

Table 4. MCTP tasks allocation and scheduling results: Case 1.

UAV	Total Flight Time (s)	Total Task Work Time (s)	Total Flight Distance (m)	Number of Assigned Tasks	Task Scheduling
UAV1	225.67	240	3166.81	8	$\{s_{51}, i_{38}, i_{46}, i_{17}, i_{15}, i_1, i_{48}, i_{34}, i_{26}, s_{60}\}$
UAV3	225.91	150	3117.41	5	$\{s_{53}, i_{30}, i_{39}, i_{23}, i_{11}, i_{16}, s_{58}\}$
UAV4	259.14	270	3463.28	9	$\{s_{54}, i_6, i_{49}, i_3, i_{18}, i_{45}, i_{25}, i_{41}, i_{35}, i_{24}, s_{57}\}$
UAV5	162.98	120	2260.10	4	$\{s_{55}, i_{36}, i_7, i_{21}, i_4, s_{53}\}$
UAV6	256.75	270	3053.24	9	$\{s_{56}, i_{43}, i_{50}, i_{14}, i_2, i_{12}, i_{33}, i_{28}, i_{40}, i_5, s_{56}\}$
UAV7	228.46	150	3132.72	5	$\{s_{57}, i_{32}, i_{22}, i_{19}, i_{47}, i_{44}, s_{55}\}$
UAV8	218.76	120	3147.48	4	$\{s_{58}, i_9, i_{29}, i_{42}, i_{27}, s_{51}\}$
UAV10	254.32	180	3106.79	6	$\{s_{60}, i_{20}, i_{31}, i_8, i_{37}, i_{10}, i_{13}, s_{54}\}$

UAV  $u_3$  takes the longest time to complete the assigned tasks. Suppose eight UAVs take off at the same time. The maximum time for the multi-UAV system to complete all tasks and flights is  $t_{total} = 529.14$  s, and the total distance of flight is  $d_{total} = 24447.86$  m. However, four possible spatiotemporal trajectory conflicts existed in the original flight route plans. These conflicts are shown in Figure 14.



**Figure 14.** Case 1: four trajectory conflicts.

Based on the optimized multi-UAV route plans, holding time is assigned to the UAV task to prevent possible conflicts by using a mixed linear programming (MILP) approach. Each task is associated with a holding time variable. It should be noted that although the UAV has no task at the charging station and has to take off from the charging station, there is still a holding time before departure to the first task. This is because a conflict may occur during the first flight from the charging station to the first task; therefore, the holding time variable is also set here. The optimized result is shown in Figure 15.

The optimized holding time is assigned by  $x_{56} = 20$ , which corresponds to the sixth task of UAV10. UAV1 is also assigned a holding time  $x_1 = 20$ , which indicates that UAV1 should postpone the first flight to avoid conflict between UAV3 and UAV4 in the first flight trajectory, which is shown in Figure 14.

By assigning holding time to tasks, the total minimum holding time can be obtained, where  $F = \min \sum_{i \in M} \sum_{k \in N} t_k^{i,h}$ . In Case 1,  $F = t_1^{1,h} + t_{10}^{6,h} = 4$ , which corresponds to a minimum of 40 seconds of total holding time in the MCTP problem to avoid conflicts.

#### 4.2.2. Case 2 with Five UAVs

Case 2 (in Figure 16) needs five charging stations and five UAVs for 50 tasks. The flight distance  $d_{total}$  and flight time  $t_{total}$  of UAVs are shown in Table 5.

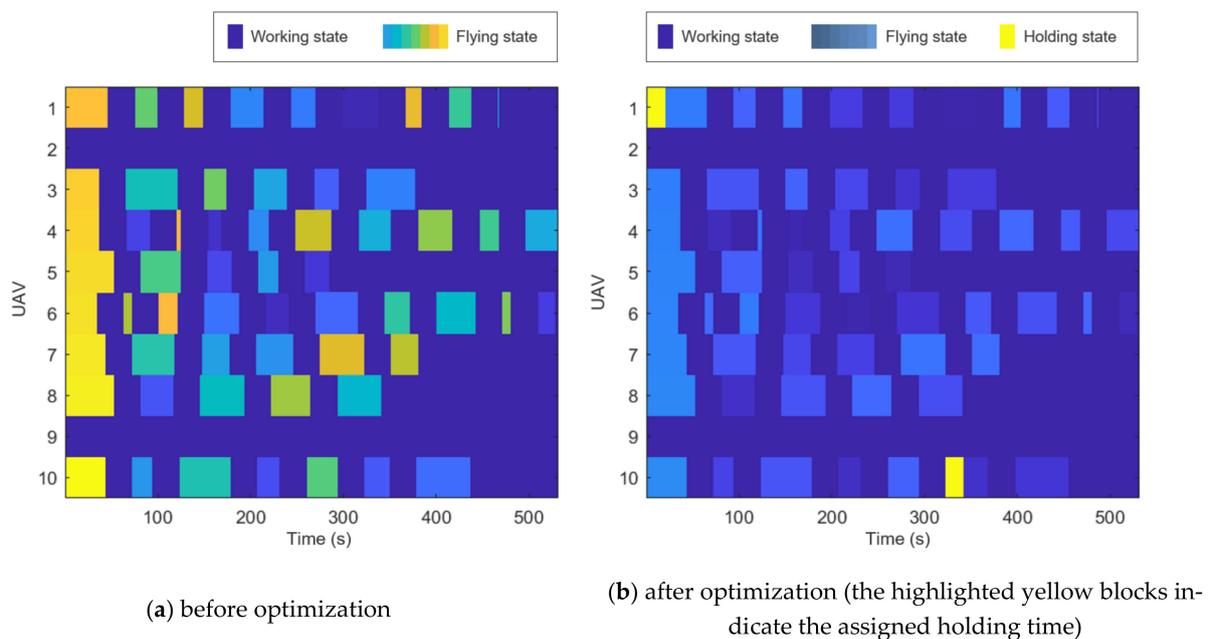


Figure 15. State-time graph of Case 1. (UAV2 and UAV9 are with no tasks).

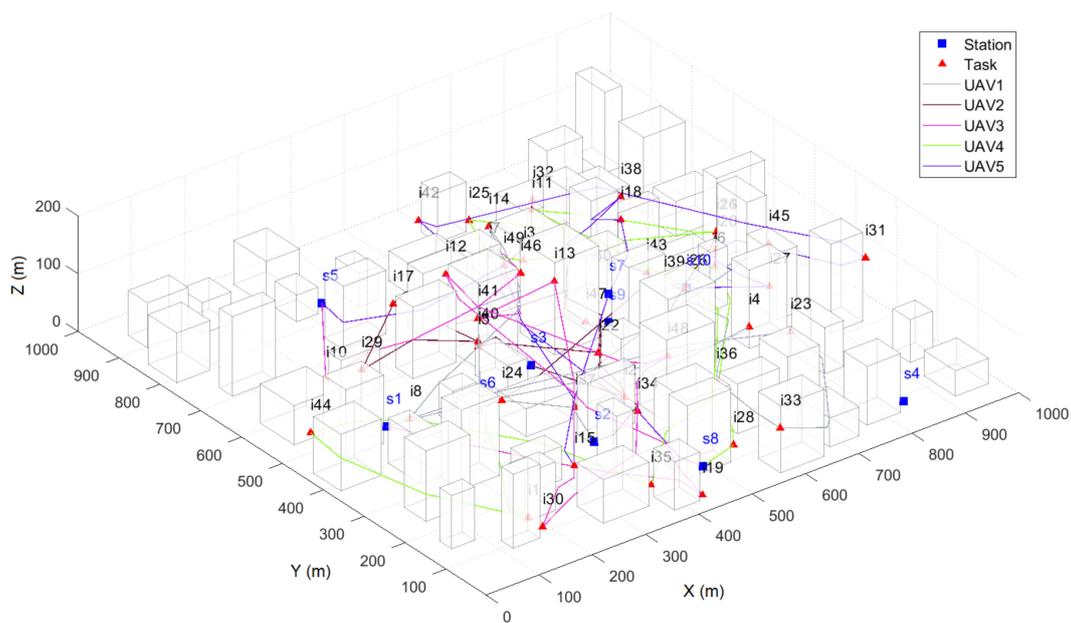


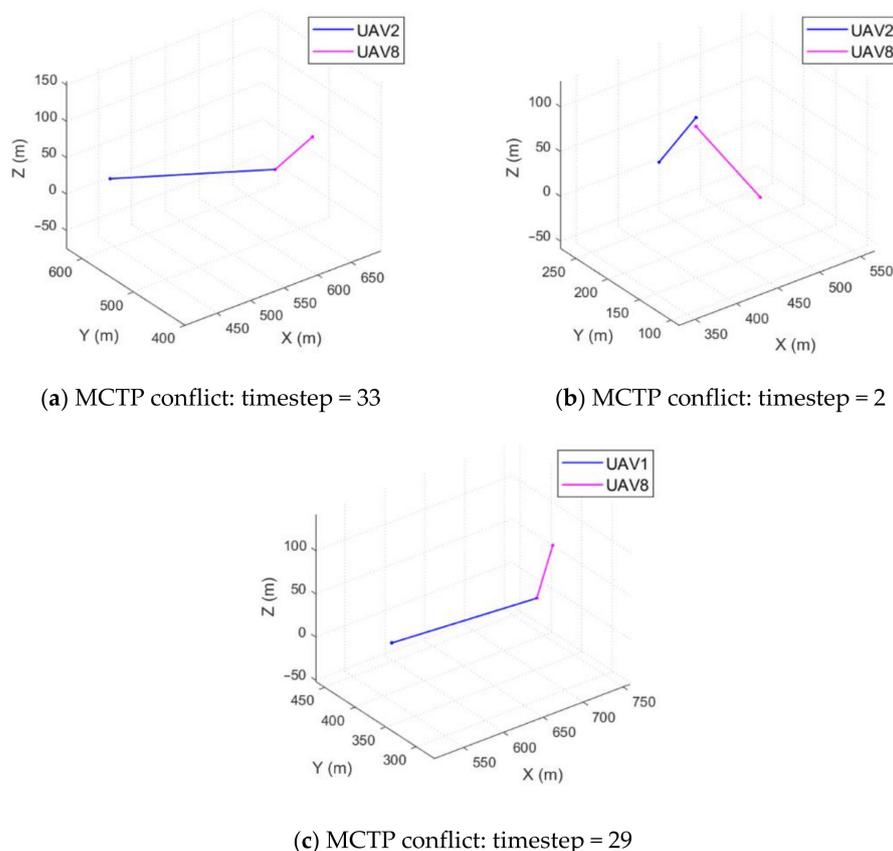
Figure 16. MCTP results of Case 2 in the virtual city.

Table 5. MCTP tasks allocation and scheduling results: Case 2.

UAV	Total Flight Time (s)	Total Task Work Time (s)	Total Flight Distance (m)	Number of Assigned Tasks	Task Scheduling
UAV1	343.70	300	4763.52	10	{s <sub>51</sub> , i <sub>4</sub> , i <sub>37</sub> , i <sub>24</sub> , i <sub>14</sub> , i <sub>8</sub> , i <sub>23</sub> , i <sub>33</sub> , i <sub>36</sub> , i <sub>39</sub> , i <sub>45</sub> , s <sub>52</sub> }
UAV2	320.80	330	4240.58	11	{s <sub>52</sub> , i <sub>47</sub> , i <sub>32</sub> , i <sub>9</sub> , i <sub>17</sub> , i <sub>29</sub> , i <sub>40</sub> , i <sub>50</sub> , i <sub>22</sub> , i <sub>5</sub> , i <sub>2</sub> , i <sub>21</sub> , s <sub>56</sub> }
UAV5	340.79	270	3837.02	9	{s <sub>55</sub> , i <sub>10</sub> , i <sub>46</sub> , i <sub>7</sub> , i <sub>41</sub> , i <sub>13</sub> , i <sub>30</sub> , i <sub>48</sub> , i <sub>12</sub> , i <sub>19</sub> , s <sub>51</sub> }
UAV6	312.35	300	4135.37	10	{s <sub>56</sub> , i <sub>1</sub> , i <sub>44</sub> , i <sub>35</sub> , i <sub>26</sub> , i <sub>11</sub> , i <sub>16</sub> , i <sub>25</sub> , i <sub>3</sub> , i <sub>6</sub> , i <sub>28</sub> , s <sub>58</sub> }
UAV8	363.21	300	4460.12	10	{s <sub>58</sub> , i <sub>34</sub> , i <sub>42</sub> , i <sub>31</sub> , i <sub>20</sub> , i <sub>27</sub> , i <sub>43</sub> , i <sub>38</sub> , i <sub>18</sub> , i <sub>15</sub> , i <sub>49</sub> , s <sub>55</sub> }

UAV8 takes the longest time for assigned tasks. Suppose all five UAVs take off at the same time; then, the maximum time for the multi-UAV system to visit all tasks is

$t_{total} = 663.21$  s, and the total distance of flight is  $d_{total} = 21,436.62$  m. In Case 2, three possible spatiotemporal trajectory conflicts exist in optimized flight route plans. These conflicts are shown in Figure 17.



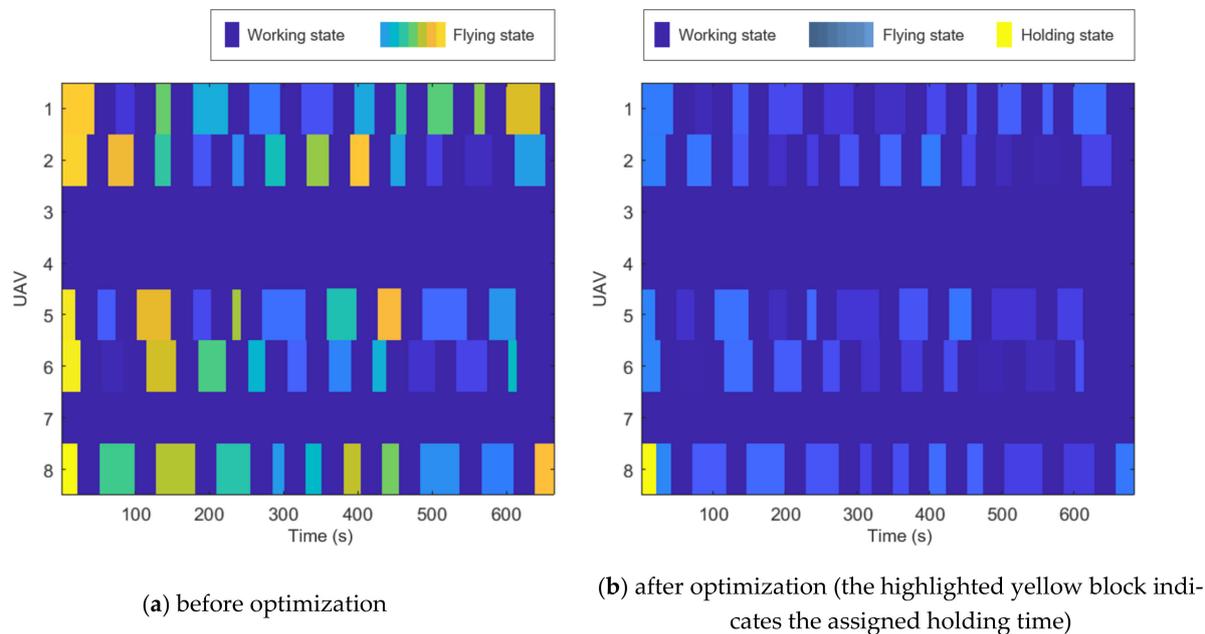
**Figure 17.** Case 2: three trajectory conflicts.

In the optimized results of MCTP, the optimized holding time is assigned by  $x_{44} = 20$ , which corresponds to the first task of UAV8, which indicates that UAV8 should postpone the take-off at the beginning to avoid conflict between UAV1 and UAV2 (see Figure 17). The minimum holding time can be obtained in  $F = \min \sum_{i \in M} \sum_{k \in N} t_k^{i,h}$ . In Case 2,  $F = t_8^{1,h} = 2$ , which corresponds to a minimum of 20 seconds of total holding time for the MCTP problem to avoid conflicts, which is shown in Figure 18.

#### 4.2.3. Comparison between Case 1 and Case 2

Compared with the results of Case 1 and Case 2, while reducing the number of system hardware deployments (from eight to five), the assignment of tasks is also relatively even, which greatly reduces the total flight distance of the multi-UAV system. The total distance  $d_{total}$  is shortened by 3011 meters. In actual applications, by adding the hardware cost of the charging station and UAV to the cost of system operation, Case 2 greatly reduces the system investment and manpower for the operator. With biased weights on UAV hardware, corresponding results can be obtained for the optimized hardware number. Therefore, the number of hardware items is also a decision variable in the MCTP herein. This kind of correspondence increases the complexity of optimization. However, from the other perspective of metrics, because of the reduced UAVs, the total time cost  $t_{total}$  is increased by 134.07 seconds compared with Case 1. For time-sensitive applications, this is certainly not good news. One UAV takes off from a charging station  $s_i$  and does not need to return to the charging station  $s_i$  after a series of tasks. In Table 4, UAV6 starts from charging station  $s_{56}$  and finally returns to  $s_{56}$ . However, in Case 2, as shown in Table 5, each UAV takes off from

a charging station and returns to a different UAV. This one-to-many UAV and charging station policy ensures system-level flexibility and robustness in real urban applications.



**Figure 18.** State-time graph of Case 2. (UAV3, UAV4, and UAV7 are with no tasks.).

#### 4.3. Real City MCTP Results and Discussion

Two test cases are tested for multi-UAV planning under the framework of bilevel algorithm. In the upper level, multi-UAV trajectory planning is modeled based on the extended mTSP algorithm by a genetic algorithm. In Case 1, the population is set to 100, and the generation of mutations is set to 100. In Case 2, the population is set to 500, and the generation of mutations is set to 500. Task allocation and task scheduling are optimized in the upper level. In each iteration, multi-UAV conflict analysis is conducted based on spatiotemporal trajectories. Other experiment configurations remain the same, including the four objective weights.

In comparison between Case 1 and Case 2 for MCTP planning results, Case 1 (shown in Figure 19) requires three UAVs, and Case 2 requires two UAVs (shown in Figure 20). Due to the wide coverage of Yuzhong Peninsula, no trajectory conflict was found in the optimization process of Case 1 and Case 2. As the actual flight distance in the urban area has greatly increased, the 30 min flight endurance setting is facing challenges. In order to access the multiple tasks assigned in this area, the multi-UAV system can make better use of the advantages of the deployed charging stations to perform multiple tasks in large-scale urban areas. In Case 2, as the population and generation number increase to 500, the optimization result of multitasking access in this area has minimized infrastructure deployment costs: only two UAVs and charging stations are needed. At the same time, the optimized task scheduling of two UAVs makes better use of the functions of the charging station, greatly increasing the flight distance and flight coverage of large cities. It can be seen that both UAV1 and UAV2 consider endurance constraints. When the total flight time and working time of the tasks are close to 30 min, they all return to the charging stations  $s_2$  and  $s_5$  for charging, and then continue with other tasks. In Case 2, the total flight distance  $d_{total} = 29,031.87$  m. The total flight time  $t_{total} = 1231.59$  s.

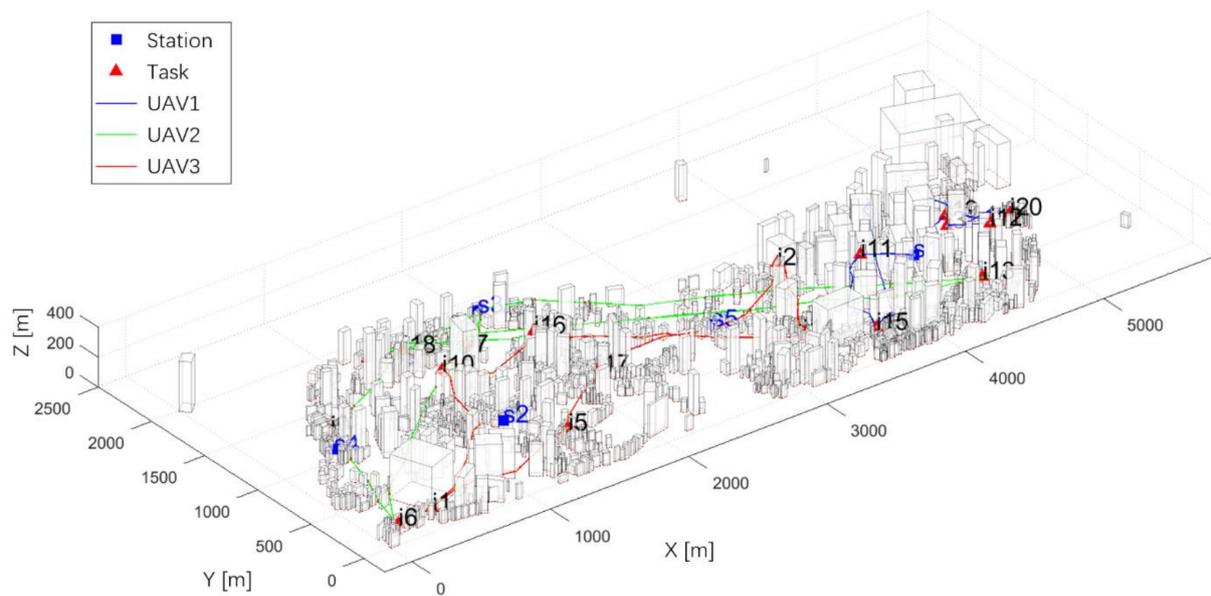


Figure 19. MCTP results of Case 1 in the real city: three UAVs.

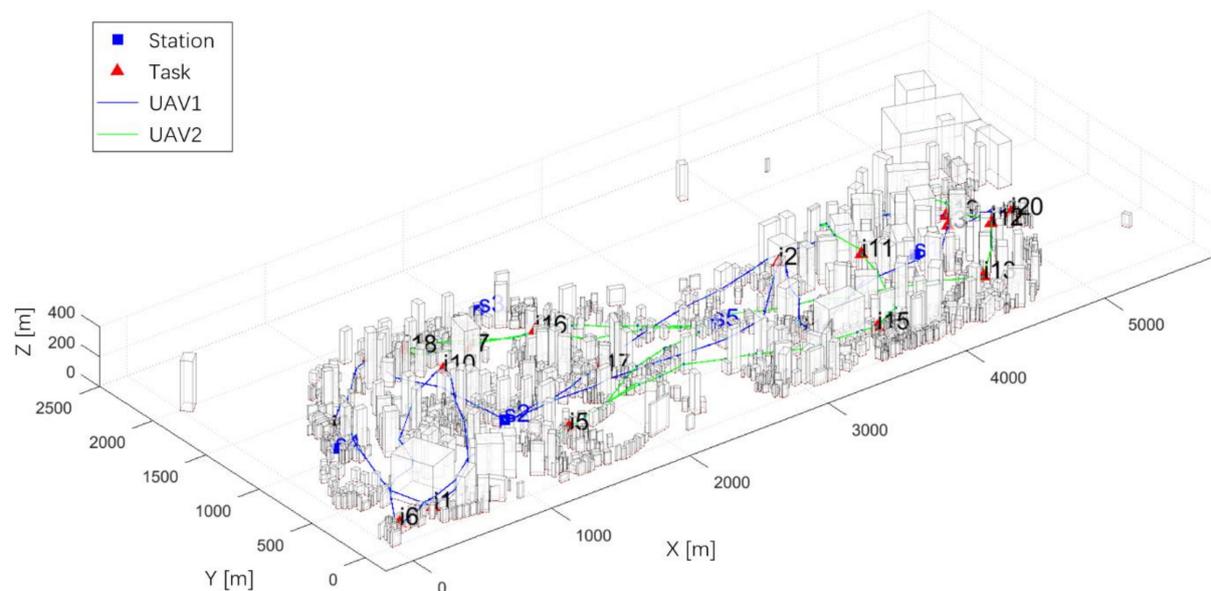


Figure 20. MCTP results of Case 2 in the real city: three UAVs.

As shown in Table 6, in Case 2, the two mTSP trajectories both cross the east and west sides of the entire peninsula. After the endurance time exceeds the maximum endurance time of the drone, UAVs choose to return to the charging station first, then continue with the tasks, and return to the respective charging station after the tasks are over. In actual multi-UAV system deployment, if considering the infrastructure cost of UAVs and associated automatic charging/battery replacement station, and system installation, operation, and maintenance cost, Case 2 greatly reduces the system investment and manpower for the operator. Case 2 also provides a cost-optimal and flight-time-balanced deployment strategy for large-scale real city.

**Table 6.** MCTP tasks allocation and scheduling results in the real city.

Total Flight Time (s)		Total Flight Distance (m)	Task Allocation and Scheduling	Computation Time (s)
<b>Case 1: Population 100, Generation 100</b>				
UAV1	510.46	5921.48	$\{s_1, i_{11}, i_{15}, i_4, i_{12}, i_3, i_{20}, i_{19}, s_1\}$	19.96
UAV2	891.6	13,108.13	$\{s_3, i_7, i_8, i_{13}, s_3, i_6, i_9, i_{18}, s_3\}$	
UAV3	643.87	7891.79	$\{s_5, i_2, i_{14}, i_{16}, i_{10}, i_1, i_5, i_{17}, s_5\}$	
<b>total</b>	891.6	26,921.41		
<b>Case 2: Population 500, Generation 500</b>				
UAV1	1231.59	16,544.28	$\{s_2, i_8, i_9, i_1, i_{10}, i_6, s_2, i_2, i_{14}, i_3, i_{20}, i_{19}, i_{17}, s_2\}$	453.09
UAV2	960.18	12,487.59	$\{s_5, i_5, i_{15}, i_{11}, i_4, i_{12}, i_{13}, s_5, i_{16}, i_7, i_{18}, s_5\}$	
<b>total</b>	1231.59	29,031.87		

## 5. Conclusions

For urban aerial applications, multi-UAV trajectory planning is of great importance to future delivery, service patrol, and emergency response services. This paper establishes a bilevel model for the multi-UAV cooperative trajectory-planning method considering conflict avoidance. By extending the control network approach, the upper level optimizes the task allocation and sequencing based on the multitraveling salesman problem (mTSP) under the framework of a heuristic algorithm. The lower level is based on mixed-integer linear programming (MILP) to solve the holding time assignment problem, thereby constructing a conflict-free system operation strategy. To verify the proposed algorithm for MCTP, numerical experiments were carried out in both the virtual city and real city. The different configurations of the optimization objective weight are discussed. At the same time, aiming at conflict-free cooperative trajectory planning, the bilevel optimization algorithm framework and the holding time optimization model are verified. The results show a wide range and variety of application prospects for multi-UAV system deployment in complex urban environments in the future. Future research could discuss the influence of the variable weight of UAVs in parcel delivery based on the proposed MCTP model. Real-time tasks and online multi-UAV system planning are also worthy problems for operational flexibility reasons.

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

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Acknowledgments:** The research is supported by the Center for Intelligent Connected Vehicles and Transportation, Tsinghua University, Beijing, China.

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

## References

- Liu, M.; Liu, X.; Zhu, M.; Zheng, F. Stochastic drone fleet deployment and planning problem considering multiple-type delivery service. *Sustainability* **2019**, *11*, 3871. [CrossRef]
- Eun, J.; Song, B.D.; Lee, S.; Lim, D.-E. Mathematical investigation on the sustainability of UAV logistics. *Sustainability* **2019**, *11*, 5932. [CrossRef]
- Park, J.; Kim, S.; Suh, K. A comparative analysis of the environmental benefits of drone-based delivery services in urban and rural areas. *Sustainability* **2018**, *10*, 888. [CrossRef]
- Amazon. Amazon Prime Air. 2016. Available online: <https://www.amazon.com//Amazon-Prime-Air/b?ie=//UF8&node=8037720011> (accessed on 9 December 2019).

5. DHL. Unmanned Aerial Vehicles—Ready for Take-Off? 2014. Available online: <https://www.logistics.dhl/global-en/home/insights-and-innovation/thought-leadership/trendreports/unmanned-aerial-vehicles.html> (accessed on 22 March 2019).
6. Perboli, G.; Rosano, M. Parcel delivery in urban areas: Opportunities and threats for the mix of traditional and green business models. *Transp. Res. Part C Emerg. Technol.* **2019**, *99*, 19–36. [[CrossRef](#)]
7. Salama, M.; Srinivas, S. Joint optimization of customer location clustering and drone-based routing for last-mile deliveries. *Transp. Res. Part C Emerg. Technol.* **2020**, *114*, 620–642. [[CrossRef](#)]
8. Macias, J.E.; Angeloudis, P.; Ochieng, W. Optimal hub selection for rapid medical deliveries using unmanned aerial vehicles. *Transp. Res. Part C Emerg. Technol.* **2020**, *110*, 56–80. [[CrossRef](#)]
9. Francisco, M. Organ delivery by 1000 drones. *Nat. Biotechnol.* **2016**, *34*, 684. [[CrossRef](#)]
10. Hopcroft, J.E.; Schwartz, J.T.; Sharir, M. On the Complexity of Motion Planning for Multiple Independent Objects; PSPACE-Hardness of the “Warehouseman’s Problem”. *Int. J. Robot. Res.* **1984**, *3*, 76–88. [[CrossRef](#)]
11. Aggarwal, S.; Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. *Comput. Commun.* **2020**, *149*, 270–299. [[CrossRef](#)]
12. Bagherian, M.; Alos, A. 3D UAV trajectory planning using evolutionary algorithms: A comparison study. *Aeronaut. J.* **2015**, *119*, 1271–1285. [[CrossRef](#)]
13. Azadeh, S.S.; Bierlaire, M.; Maknoon, M.Y. A two-stage route optimization algorithm for light aircraft transport systems. *Transp. Res. Part C Emerg. Technol.* **2019**, *100*, 259–273. [[CrossRef](#)]
14. Lin, X.; Wang, C.; Wang, K.; Li, M.; Yu, X. Trajectory planning for unmanned aerial vehicles in complicated urban environments: A control network approach. *Transp. Res. Part C Emerg. Technol.* **2021**, *128*, 103120. [[CrossRef](#)]
15. Rasche, C.; Stern, C.; Kleinjohann, L.; Kleinjohann, B. A distributed multi-uav path planning approach for 3d environments. In Proceedings of the 5th International Conference on Automation, Robotics and Applications, Wellington, New Zealand, 6–8 December 2011; pp. 7–12.
16. Ergezer, H.; Leblebicioğlu, K. Path planning for multiple unmanned aerial vehicles. In Proceedings of the 2012 20th Signal Processing and Communications Applications Conference (SIU), Mugla, Turkey, 18–20 April 2012; pp. 1–4.
17. Moon, S.; Oh, E.; Shim, D.H. An integral framework of task assignment and path planning for multiple unmanned aerial vehicles in dynamic environments. *J. Intell. Robot. Syst.* **2013**, *70*, 303–313. [[CrossRef](#)]
18. Ergezer, H.; Leblebicioğlu, K. 3D path planning for multiple UAVs for maximum information collection. *J. Intell. Robot. Syst.* **2014**, *73*, 737–762. [[CrossRef](#)]
19. Furini, F.; Persiani, C.A.; Toth, P. The time dependent traveling salesman planning problem in controlled airspace. *Transp. Res. Part B Methodol.* **2016**, *90*, 38–55. [[CrossRef](#)]
20. Persiani, C.A.; Bagassi, S. Route planner for unmanned aerial system insertion in civil non-segregated airspace. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2013**, *227*, 687–702. [[CrossRef](#)]
21. Hönig, W.; Preiss, J.A.; Kumar, T.S.; Sukhatme, G.S.; Ayanian, N. Trajectory planning for quadrotor swarms. *IEEE Trans. Robot.* **2018**, *34*, 856–869. [[CrossRef](#)]
22. Sharon, G.; Stern, R.; Felner, A.; Sturtevant, N.R. Conflict-based search for optimal multi-agent pathfinding. *Artif. Intell.* **2015**, *219*, 40–66. [[CrossRef](#)]
23. Solovey, K.; Salzman, O.; Halperin, D. Finding a needle in an exponential haystack: Discrete RRT for exploration of implicit roadmaps in multi-robot motion planning. *Int. J. Robot. Res.* **2016**, *35*, 501–513. [[CrossRef](#)]
24. Araki, B.; Strang, J.; Pohorecky, S.; Qiu, C.; Naegeli, T.; Rus, D. Multi-robot path planning for a swarm of robots that can both fly and drive. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Marina Bay Sands Convention Centre, Singapore, 29 May–3 June 2017; pp. 5575–5582.
25. Silver, D. Cooperative Pathfinding. *Aiide* **2005**, *1*, 117–122.
26. Čáp, M.; Novák, P.; Kleiner, A.; Selecký, M. Prioritized planning algorithms for trajectory coordination of multiple mobile robots. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 835–849. [[CrossRef](#)]
27. Wagner, G.; Choset, H. Subdimensional expansion for multirobot path planning. *Artif. Intell.* **2015**, *219*, 1–24. [[CrossRef](#)]
28. Yu, J.; LaValle, S.M. Optimal multirobot path planning on graphs: Complete algorithms and effective heuristics. *IEEE Trans. Robot.* **2016**, *32*, 1163–1177. [[CrossRef](#)]