



Article Multi-UAV Unbalanced Targets Coordinated Dynamic Task Allocation in Phases

Wenfei Wang¹, Maolong Lv², Le Ru^{1,*}, Bo Lu¹, Shiguang Hu¹ and Xinlong Chang³

- ¹ College of Equipment Management and UAV Engineering, Air Force Engineering University, Xi'an 710043, China
- ² Air Traffic Control and Navigation College, Air Force Engineering University, Xi'an 710043, China
- ³ Xi'an Research Institute of Hi-Tech, Xi'an 710038, China
- * Correspondence: ru-le@163.com; Tel.:+86-137-7200-0382

Abstract: Unmanned aerial vehicles (UAVs) can be used in swarms to achieve multiple tasks cooperatively. Multi-UAV and multi-target cooperative task assignments are difficult. To solve the problem of unbalanced, phased, cooperative assignment between UAVs and tasks, we establish an unbalanced, phased task assignment model that considers the constraints of task execution, time, and target task execution demand. Based on an improved consensus-based bundle algorithm (CBBA), we propose a two-tier task bidding mechanism. According to dynamic changes in new tasks, we study a dynamic assignment strategy and propose a mechanism based on task continuity adjustment and time windows. Finally, a simulation experiment is used to verify the feasibility and effectiveness of the proposed allocation method in multi-UAV target assignment scenarios. The results show that the dynamic task assignment strategy can efficiently assign random new tasks as they arise.

Keywords: multiple UAVs; dynamic task assignment; consensus-based bundle algorithm; coordination



Citation: Wang, W.; Lv, M.; Ru, L.; Lu, B.; Hu, S.; Chang, X. Multi-UAV Unbalanced Targets Coordinated Dynamic Task Allocation in Phases. *Aerospace* 2022, *9*, 491. https:// doi.org/10.3390/aerospace9090491

Academic Editor: Roberto Sabatini

Received: 21 July 2022 Accepted: 29 August 2022 Published: 1 September 2022

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



Copyright: © 2022 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/).

1. Introduction

Under the current military context, unmanned aerial vehicles (UAVs) are widely used in reconnaissance, searching, firefighting, effectiveness evaluation, and other operational tasks [1]. However, in the future air combat environment, UAVs will be faced with complex and changeable environments. The cooperative environment and requirements of UAVs will also become complex [2], which will require efficient cooperation among UAVs to complete various objectives and tasks [3].

The UAV multi-objective assignment problem is a typical NP-hard problem. Reasonable consideration of the influences of practical factors and the constraint relationships between UAVs and targets are very important in establishing cooperative task assignment models and improving design and solution algorithms. The problem of multi-task cooperative assignment has been studied from the two aspects of *random speed* and *time windows*. When considering the constraints of kinematics, time, and resources, an improved genetic algorithm was proposed to solve the problem of random task assignment [4]. In [5], a discrete adaptive whale search optimization algorithm was proposed to study the task assignment problem. It comprehensively considers various constraints, such as limited resources, task priority, and obstacle avoidance. It features a search-intensity adaptive mechanism that balances exploration and development intensity and effectively improves the exploration ability. In [6], a two-stage hybrid auction algorithm was proposed. In its first level, urgent tasks are selected by using a selection function and task attribute values. In its second level, a new objective function is proposed that generates a reasonable task auction sequence, which improves the efficiency and robustness of the task allocation model.

In recent years, many scholars at home and abroad have carried out research on the UAV task assignment problem and proposed a variety of different types of task assignment problem models and solving methods. The problem model establishment is mainly divided

into single-task assignment models and multi-task assignment models. The single task assignment model mainly includes the Multiple Travelling Salesman Problem [7] model and the Vehicle Routing Problem [8] model. The main feature of the model is that the UAV needs to traverse all the task targets. The model is simple but needs to be standardized in the actual problem processing process and is suitable for fewer task targets. In the multitask assignment model, the Mixed-Integer Linear Programming [9] model is beneficial for describing the global constraints of numerical problems, better optimizing the UAV task execution sequence, and improving task assignment efficiency. However, the method is difficult to apply to situations where there is an imbalance between the number of UAVs and the number of tasks. The distribution idea of the Network Flow Optimization [10] model is to use the eigenvectors of network traffic for hierarchical division, which has good optimization characteristics, but the problem of low distribution efficiency often occurs when the number of tasks surges. Therefore, this paper aims at the problem of multi-UAV cooperative task assignment in stages, considers a variety of practical constraints, and combines the different execution requirements of various tasks in different time periods to establish a hierarchical multi-stage multi-UAV cooperative task allocation model so as to achieve reasonable planning of UAV formation collaborative execution tasks in different phases of task execution time windows.

In combat environments, the various battlefield situations are constantly changing. The emergence of new situations can make the original task allocation strategy unable to meet the new task execution requirements. Hence, there is an urgent need to carry out research on dynamic task allocation for multiple UAVs. In [11], aiming to solve the problem of the calculation times and flight distance involved in task allocation are too long, and so a dynamic task allocation mechanism based on a bionic algorithm was proposed. It dynamically allocates UAVs according to the number of task points and optimizes the allocation via a comparison mechanism. In [12], an improved task-assignment algorithm was proposed, which was applied to combat scenes with sudden threats and UAV losses. The algorithm is market-based and simulates the four stages of market bidding. Simulations prove that the algorithm has good robustness and can be well applied to the dynamic assignment of UAV cooperative timing tasks. In [13], a dynamic task allocation algorithm based on a task sequence mechanism was proposed. The available time period and task resources of each UAV are defined, and a conflict-free task allocation scheme is allocated to each UAV in real-time through an auction mechanism and consistency coordination. At present, research on the dynamic task allocation of multiple UAVs has mostly focused on execution costs and path planning. However, in actual situations, the task execution of multiple UAVs has strong time constraints [14], and the relationships between task completion time constraints and task benefits need to be considered [15].

At the current stage, the methods for solving the multi-UAV cooperative task assignment problem mainly include heuristic methods, mathematical programming methods, and intelligent optimization algorithms. Among them, the heuristic methods mainly include clustering methods [16]. The principle is to treat each task as a cluster for clustering and complete the task assignment when the task cluster and the number of system members reach an agreement. The advantage is that the time and space complexity of the algorithm is low, but its robustness and accuracy are poor. Mathematical programming methods mainly include exhaustive methods, integer programming methods, etc. Among them, integer programming methods [17] take small-scale problems as the research object and establish objective functions and constraints around the problem objectives to solve. The algorithm is flexible and easy to solve, but the computational load is large, and the solution efficiency is low, and it is difficult to adapt to the solution of large-scale problems. The principle of the intelligent optimization algorithm [18] is to adjust the solution time and quality to obtain a satisfactory solution within an acceptable time range. Its advantage is that the solution is stable and simple and can be combined with other methods, mainly including evolutionary algorithms and swarm intelligence algorithms. The above-solving algorithms are all applicable to the centralized task assignment mechanism. The centralized

solution method has a simple solution mechanism and can theoretically obtain the optimal task assignment result. However, the algorithm solution is easily limited by the centralized structure and is not suitable for application in dynamic, uncertain task allocation scenarios, the real-time assignment performance is low, and the robustness is poor, which reduces the overall task assignment efficiency to a certain extent.

In the actual multi-UAV cooperative task assignment process, each UAV needs to assign tasks coordinately through information communication interaction. The distributed control system structure [19] has better structural flexibility and autonomy, and the current more widely used task allocation methods are the contract network algorithm and the auction algorithm in the market-like mechanism, and its core idea is to use the market auction mechanism to achieve task allocation [20]. Therefore, this paper adopts an improved algorithm of auction algorithm-consensus-based bundle algorithm to solve the distributed task assignment problem of multiple UAVs. Compared with the previous traditional distributed assignment algorithm, this algorithm adopts a task bidding concurrency mechanism and UAV neighborhood communication. The negotiation mechanism has better task allocation flexibility and efficiency and uses the consistency strategy to effectively avoid the occurrence of task conflicts.

Aiming at the above problems, this paper first establishes a multi-UAV, multi-objective, unbalanced, cooperative task assignment model that comprehensively considers various constraints [21]. An improved consensus-based bundle algorithm (CBBA) is used to solve the model [22,23]. Secondly, according to the dynamics of new task occurrence, a dynamic assignment mechanism based on task continuity adjustment and time-window matching is proposed [24,25] to achieve multi-UAV cooperative dynamic task assignment. Finally, a simulation experiment verifies the feasibility and effectiveness of the proposed method.

2. Materials and Methods

2.1. Task Assignment in Multi-UAV Cooperative Combat

2.1.1. Description of the Multi-UAV Cooperative Task Assignment Decision Problem

The *multi-UAV cooperative task assignment problem* refers to a cooperative task assignment scheme for multiple UAVs that considers their performance and task characteristics. It can meet specific constraints and maximize the benefit of task assignments. In the process of establishing a collaborative task allocation model, it is necessary to analyze the model's constraints and optimization objectives [26]. The task execution constraints are related to timing, time, quantity, and demand.

(1) For the *i*th UAV, the number of tasks it can perform should not exceed the maximum it can perform, namely:

$$\sum_{i=1}^{NT} x_{ij} \le L_t \tag{1}$$

where *I* and *J* represent the UAV set and task set, respectively, where $I \triangleq \{1, ..., N_n\}$ and $J \triangleq \{1, ..., N_t\}$. x_{ij} represents the assignment of UAV *i* to task *j*. If the value is 1, it means that UAV *i* is assigned to the task; otherwise, it is 0 (not assigned). L_t represents the maximum number of tasks that each UAV can perform. *NT* represents the number of tasks.

(2) The number of UAVs that execute the *j*th goal should not exceed the upper limit of the number of UAVs, namely:

$$\sum_{i=1}^{NU} x_{ij} \le L_i \tag{2}$$

where L_i represents the maximum number of UAVs. NU represents the number of UAVs.

(3) When a UAV performs its mission, it needs to abide by certain task execution timing constraints; that is, it should first perform a reconnaissance mission before performing a strike mission, namely:

$$t_r^{T_j} \le t_c^{T_j} \quad (j = 1, 2, \cdots, NT)$$
 (3)

where $t_r^{T_j}$ and $t_c^{T_j}$ indicate the starting times of the reconnaissance and strike missions, respectively.

(4) A UAV's mission needs to be executed within the executable time range of the mission, namely:

$$Et_i \le t_i \le Lt_i \tag{4}$$

where Et_i and Lt_i represent the start and end execution times, respectively. t_i represents the time when the UAV *i* performed the mission.

(5) Different stages of task execution require different numbers of UAVs to perform cooperatively, namely:

$$TarU\left(t_{et_{i}^{k},lt_{i}^{k}}^{j}\right) \in UAV\left(t_{et_{i}^{*k},lt_{i}^{*k}}^{j}\right)$$

$$\tag{5}$$

where $TarU\left(t_{et_{i}^{k},lt_{i}^{k}}^{j}\right)$ represents the required number of UAVs in the *k*th execution period $\left[et_{i}^{k},lt_{i}^{k}\right]$. of the *j*th target. $UAV\left(t_{et_{i}^{*k},lt_{i}^{*k}}^{j}\right)$ indicates the number of UAVs that can reach the task during period $\left[et_{i}^{*k},lt_{i}^{*k}\right]$, and $\left[et_{i}^{k},lt_{i}^{k}\right] \in \left[et_{i}^{*k},lt_{i}^{*k}\right]$.

In model optimization, the main considerations are to maximize the number of tasks performed by the UAVs and minimize the time spent on task execution. The cooperative tasks of the UAVs are defined as:

$$f_1 = \sum_{i=1}^{NU} \left(\sum_{j=1}^{NT} c_{ij}(x_i, p_i) x_{ij} \right)$$
(6)

where c_{ij} represents the revenue from task execution.

We define the time cost as:

$$f_2 = \sum t_{ij} \cdot x_{ij} \tag{7}$$

In summary, the mathematical model of UAV cooperative target assignment is as follows:

$$\min\left(f_{1}^{-1}, f_{2}\right)$$

$$\min\left(f_{1}^{-1}, f_{2}\right)$$

$$s.t.\left\{\begin{array}{c}\sum_{j=1}^{N_{t}} x_{ij} \leq L_{t}\\\sum_{j=1}^{N_{n}} x_{ij} \leq L_{i}\\t_{r}^{T_{j}} \leq t_{c}^{T_{j}}\\Et_{i} \leq t_{i} \leq Lt_{i}\\TarU\left(t_{et_{i}^{k}, lt_{i}^{k}}^{j}\right) \in UAV\left(t_{et_{i}^{*k}, lt_{i}^{*k}}^{j}\right)\\x_{ij} \in \{0, 1\}\end{array}\right.$$
(8)

where the set revenue function satisfies $c_{ij}(x_i, p_i) \ge 0$ under any distribution scheme x_i and task execution sequence p_i .

2.1.2. Consensus-Based Bundle Algorithm

In this paper, the consensus-based bundle algorithm is used to solve the dynamic task assignment problem of multi-UAV cooperation [27]. The assignment process mainly consists of *task selection* and *conflict resolution*. First, based on the construction of their own task package and execution path, each agent calculates the income from adding new tasks

and compares it with the winning bid price of the current task. When the income is higher than the winning bid, it is added to the task package to achieve task selection. Secondly, to achieve consistent mission selection, the agent needs to know all kinds of bidding information with all agents, which mainly contains the winning bid, the winner of the sets, and a timestamp. According to these information interactions, all agents update their task packages and task execution paths in real time. Through continuous task selection and conflict resolution, conflict-free task assignment results are finally obtained. The specific execution mechanism is shown in Figure 1.



Figure 1. Flow chart of the CBBA.

2.1.3. Algorithmic Information Structure

Assume that all UAVs in the UAV formation are set as $L \triangleq \{1, 2, ..., N_n\}$, and the task execution target is represented by $J \triangleq \{1, 2, ..., N_t\}$. The various elements that need to be stored and updated by each UAV in the process of task allocation mainly include:

- (1) Task package b_i : Vector $b_i = \{b_{i1}, b_{i2} \dots, b_{in}\}$ represents the task set assigned to UAV *i*, which is arranged in the order of insertion.
- (2) Task path p_i : $p_i = \{p_{i1}, p_{i2}, \dots, p_{in}\}$ represents the execution list of the task set assigned to UAV *i*, and each task element starts its specific execution according to the task path.
- (3) Winner set $z_i: N_n \times N_t$ -dimensional matrix z_i represents the task bidding information currently acquired by UAV $i; z_{ij}^k = 1$ represents that UAV i confirms that UAV k is the bidding winner of task j; otherwise, it is 0.
- (4) Winning bid price set y_i: Matrix y_i represents the UAV bidding information confirmed by UAV *i*, which corresponds to the matrix of the winner set.
- (5) Timestamp s_i: s_i represents the latest moment of information exchange between UAV *i* and its neighbouring UAVs. This plays an important role in the conflict resolution stage and represents the update degree of information obtained by the current interaction between UAV and other UAVs.

2.1.4. Task Selection Mechanism Improvement

In the process of task selection, in order to maximize the benefits of task execution, task *j* is added to task set b_i in each UAV iteration until the end of the iteration cycle. The total revenue of UAV *i*'s task allocation is:

$$S_i^{p_i} = \sum_j (c_{ij} - \zeta_{ij}) \tag{9}$$

where c_{ii} is the revenue from task execution, which is expressed as follows:

$$\mathbf{c}_{ij}' = \begin{cases} K_{ij} \times e^{-\lambda(t_{ijs} - t_{js})} & t_{ijs} \in [t_{js}, t_{je}] \\ 0 & \text{otherwise} \end{cases}$$
(10)

where t_{ijs} is the start time of UAV *i* performing task *j*, $\lfloor t_{js}, t_{je} \rfloor$ is the executable time range of task *j*, and exp $\left(-\lambda(t_{ijs} - t_{js})\right)$ is a time penalty term for UAV *i* performing task *j*.

Where ζ_{ij} is the mission distance cost of the UAV, which has the specific form:

$$\zeta_{ij} = d_{ij}\gamma\tag{11}$$

where d_{ij} is the straight-line distance between UAV *i* and task *j*, and γ is the distance cost coefficient.

Using the change in revenue of UAV *i* after adding new task *j*, the marginal gain of task *j* can be expressed as:

$$c_{ij}[b_i] = \begin{cases} 0, j \in b_i, \\ \max_{n \le |p_i|+1} S_i^{p_i \oplus_n \{j\}} - S_i^{p_i}, \text{ otherwise.} \end{cases}$$
(12)

where *n* represents all possible positions where the new task is added to the task path; $a \oplus_n \{b\}$ means that *b* is inserted into the *n*th bit of *a*, and the *n*th bit of *a* and the elements that follow it are moved back. We try to insert each UAV at all possible positions in the execution path to determine the position with the maximum marginal gain of the task, add all tasks to the task package and finally, maximize the UAV execution benefit. For multiple-UAV task co-allocation, we construct a winning information matrix *B* of size $N_n \times N_t$ from the set winner z_i and set winning bid y_i information, with rows and columns representing tasks and UAVs, respectively. B_{ij} is UAV *i*'s bid for mission *j*. When z(i, j) = k, there is $B_{kj}^i = y(k, j)$, indicating that UAV *i* thinks that UAV *k* bids for task *j*.

To solve assignment problems where the target tasks are at different execution stages, the bidding mechanism of the original algorithm is improved and two-level task bidding selection is implemented [6,28]. The specific process is as follows.

Firstly, for tasks at different execution stages in the target, the tasks with different execution time stages in the target are ranked according to the starting execution time; that is, $T_j = [T_1, T_2, ..., T_e]$. The first-level task bidding expression of the improved algorithm task selection mechanism is:

$$c'_{ij} = \begin{cases} K_{ij} \times e^{-\lambda(t_{ijs} - t^1_{js})} & t_{ijs} \in \begin{bmatrix} t^1_{js}, t^e_{je} \end{bmatrix} \\ 0 & \text{otherwise} \end{cases}$$
(13)

where t_{ijs} is the execution start time of UAV *i*'s task *j*, and t_{js}^k and t_{je}^e are the start time of task T_1 and the end time of task T_e in T_j , respectively. K_{ij} is the coefficient. The bidding function is mainly used to screen out the UAV sets that meet the target T_j execution time period $\left[t_{js}^1, t_{je}^e\right]$, and prepare for the next specific task bidding event.

After screening out the UAV sets that meet the task execution time period, the second task bidding expression is:

$$c_{ij}^* = \begin{cases} K_{ij} \times e^{-\lambda(t_{ijs} - t_{js}^n - t^*)} & t_{ijs} \in \left[t_{js}^n, t_{je}^n\right] \\ 0 & \text{otherwise} \end{cases}$$
(14)

where $\begin{bmatrix} t_{js}^n, t_{je}^n \end{bmatrix}$ is the start and end times of task n in $\begin{bmatrix} t_{js}^1, t_{je}^e \end{bmatrix}$, t^* is the difference in the start execution times of adjacent tasks, that is, $t^* = t_{js}^{n+1} - t_{js}^n$. The bidding function is mainly used to assign the task sets of different execution time periods in target j to a specific UAV for execution.

After the task package is built, it needs to be populated with information such as the winner set, winning price set and time stamp. Each UAV needs to judge whether its own task package built on the basis of the other UAVs' task bidding information is true or not, so it also needs to resolve any task conflict to obtain the final consistent task allocation scheme.

2.1.5. Conflict Resolution

Conflict resolution is based on task construction and solves the problem of task assignment conflict through information exchange between UAVs [29]. The information exchanged between UAVs mainly includes the winner set, winning price set, and time stamp. When a task only needs to be performed by a single UAV, its conflict resolution mechanism is consistent with the CBBA. The main processes are to judge whether the task can be obtained by comparing its own bidding information after all UAVs complete an information exchange and to obtain the winner set and winning price set information of other UAVs. When UAV *i* bids the highest for task *j*, it will obtain it and release all tasks *j* and subsequent tasks in the task set of all other UAVs containing the task *j*. When the timestamp is updated after complete information exchange occurs, the update form is as follows:

$$s_{ik} = \begin{cases} \tau_r, & g_{ik} = 1\\ \max_{m:g_{im}=1} s_{mk}, & \text{otherwise} \end{cases}$$
(15)

where $g_{ik} = 1$ represents information interaction between UAV *i* and UAV *k*; otherwise, it is 0; and τ_r indicates the time of message receipt.

In the process of conflict resolution, via information interaction between all UAVs, the information receiver of the UAV will update the corresponding information according to the winner set, winning bid set, and timestamp of the current moment. When UAV *i* receives information from neighbouring UAV *k*, it will react according to the time stamp and UAV winning information. The UAV reaction mechanism mainly includes an:

• Update mechanism :
$$y_{ij} = y_{kj}$$
, $z_{ij} = z_{kj}$;
• Reset mechanism : $y_{ij} = 0$, $Z_{ij} = \emptyset$; (16)

• Leaving mechanism :
$$y_{ij} = y_{ij}$$
, $z_{ij} = z_{ij}$.

According to the above processing mechanism, when a UAV changes the information in the winner set and winning bidding set, it is necessary to check whether there is any task update or reset in the task set. When there is, it is necessary to release the task and all subsequent tasks to achieve consistent task allocation. When the information of the sender UAV *k* and receiver UAV *i* are updated, we check the information of UAV *n* that UAV *k* thinks is assigned to task *j*. When $B_{mj}^i = 0$ exists and the number of UAVs executed by task *j* does not meet the requirement, the allocation matrix is updated by $B_{mj}^i = B_{mj}^k$ and UAV *i* is assigned to task *j*. When the number of UAVs executing task *j* meets the requirement, if their existing bidding value is less than that considered by UAV *k*, that is, $\min(y_{nj}^i \forall n) < y_{mj}^k$, the task is updated by $B_{nj}^i = 0$, $B_{mj}^i = B_{mj}^k$. To avoid deadlock when the bidding value is the same, the UAV with the higher identification is selected. Throughout the process, task selection and conflict resolution are carried out in a continuous cycle. When the winner set of all UAVs and the information of the winner bidding set no longer change, it indicates that the task assignment results of all UAVs have reached consistency, and task assignment ends.

2.2. Dynamic Task Assignment Based on Improved CBBA

2.2.1. Overall Flow of the Algorithm

In the process of task allocation, all UAVs perform tasks according to the previously established task allocation, and new tasks appear at time T, which influences the original task allocation scheme. The task allocation sequence needs to be adjusted to regain the optimal consistent allocation result [30,31]. In the process of solving the problem by using the improved CBBA, task continuity adjustment and task allocation adjustment through the time window are used in the task package construction stage to optimize the efficiency of the task allocation results and improve the dynamic effect of the final task allocation scheme [32]. The specific algorithm flow is shown in Figure 2:



Figure 2. Algorithm overall flow.

Step 1: On the basis of the initial UAV task allocation scheme, we judge whether there is an event occurring in a new task at time *T*. If so, we go to Step 2; if not, go to Step 5 directly to output the optimal task allocation scheme.

Step 2: If there are any new tasks, the improved CBBA approach is used to solve them. Firstly, we reconstruct the task package and task execution path of the UAVs that meet the task execution conditions. On the basis of the task package construction and task execution path, we optimize and adjust the task allocation scheme through task continuity adjustment and task adjustment based on the time window.

Step 3: On the basis of task package construction and task assignment scheme adjustment, conflict resolution is carried out to obtain consistent task assignment results for all UAVs.

Step 4: Judge whether the final task allocation is consistent. If so, go to Step 5 to output the optimal task allocation scheme; if not, go to Step 2 to re-allocate tasks.

Step 5: Select the optimal task assignment scheme as the adjusted optimal scheme for output.

In the process of task assignment, for UAVs before and after continuous task execution revenue and cost considerations, we comprehensively consider the combination of the currently executed task and related tasks to adjust the task allocation sequence. At the same time, for each task allocation relationship within the execution time window, reasonably plan the time sequence of the execution window of each task, the task allocation efficiency can be better optimized. The detailed processes of these two parts are introduced below.

2.2.2. Task Continuity Adjustment Distribution

If at moment *T* a new task appears that needs to be performed, the UAV that meets the task execution conditions deletes its original task allocation and rebuilds the work and task execution path. On this basis, combined with all UAVs and task execution path information, we optimize the arrangement of pre- and post-tasks of all tasks in UAV *i*'s task execution sequence [33,34].

The set of all UAVs is *I*, and the task execution sequence of UAV *i* is $T_i = \{t_{i1}, t_{i2}, \dots, t_{ih}\}$, where *h* represents the order of the task in the execution sequence. The conflict-free task set at any execution time after *t* is $T_t = \{t_{g1}, t_{g2}, \dots, t_{gw}\}$, where *g* is the individual UAV in the task execution sequence to which the task belongs, and *w* is the total number of tasks. Let *AS* as the algorithm iteration counter, and make its initial value as 0. The algorithm flow is as follows:

Step 1: Firstly, UAV *i* is selected from UAV set *I*;

Step 2: Select task t_{ih} from the task execution sequence T_i of UAV *i*;

Step 3: Select the task t_{gw} and t_{ih} indices from the conflict-free task set T_t that was executed at any time after time *t* and exchange them. Record the trip costs ζ_i and ζ_j and

the original trip costs ς_i and ς_j executed by UAV *i* and UAV *j*, respectively.

Step 4: Calculate $d_i = \frac{\varsigma_i}{r}$ and $d_j = \frac{\varsigma_j}{r}$, and if d_i and d_j are both within the threshold ς_j

range $[d^-, d^+]$, go to Step 5; otherwise, go to Step 3;

Step 5: Adjust the assignment of tasks;

Step 6: Judge whether the adjustment is completed; if so, go to Step 7, otherwise, go to Step 2;

Step 7: Let AS = AS + 1, and determine whether $AS \le NU$; if so, then go to Step 8, otherwise, go to Step 1;

Step 8: Output the final allocation result.

2.2.3. Task Allocation Adjustment Based on Time Windows

A *task window* refers to the period between the earliest and latest times a task may be started. Within a task window, the different execution times of each task will influence the probability of each task being executed [35]. Within the task window, different task execution schedules will cause the interval between execution times of adjacent tasks to be too long or too short, thus making the overall task execution sequence too close or too sparse, which can also affect the overall task execution benefits.

We define u_{ij} to represent the execution relationship between UAV *i* and task *j*. When $u_{ij} = 1$, it means that UAV *i* performs task *j*; otherwise, when the value is 0, it means that there is no execution relationship; du indicates the execution time of a task. The earliest start time of a task is *st* and the latest is *et*, the start time of the current task is *t*, dt is the time that the UAV takes to reach the task location, and the earliest task end time in the time window is represented as st + du - dt. The latest end time of the tasks in the time window is expressed as et - du + dt. *M* is used to represent the proportion of the task time window of the UAV to start the task at time *t* in the overall time window. When there is a time set *P* of all tasks that can be started within the time window, and the time set of tasks that can be started to be executed at time *t* is P_t , then $M(t) = |P_t|/|P|$, $t \in [st, et]$. The allocation adjustment objective is expressed as $\sum_{j \in N} M_j(t)u_{ij}$, and its result is minimized to optimize the overall task allocation

adjustment. If a task is executed at *t*, the earliest execution time cannot be earlier than *st*, and the earliest execution time can start from t - du + dt. Therefore, the earliest execution time of a task executed at *t* is max{t - du + dt, st}. Similarly, the latest start time of a task executed at time *t* is max{t - du + dt, st}. To sum up, for a time window [st, et], P = [st, et - du + dt], |P| = et - st - du + dt + 1, and the time window matching degree of the task executed at *t* time is $M(t) = (\min{t, et - du + dt} - \max{t - du + dt} + 1)/(et - st - du + dt + 1)$.

When the earliest task end time in the time window is earlier than the latest end time, that is, when st + du - dt < et - du + dt, and when $t \in [st, st + du - dt]$, there are $\min\{t, et - du + dt\} = t$ and $\max\{t - du + dt, st\} = st$; then, $M(t) = \frac{(t-st-1)}{(et-st-du+dt+1)}$. Simi-

$$M(t) = \begin{cases} \frac{t - st + 1}{et - st - du + dt + 1}, st \le t \le st + du - dt\\ \frac{du - dt + 1}{et - st - du + dt + 1}, st + du - dt < t \le et - du + dt\\ \frac{et - t + 1}{et - st - du + dt + 1}, et - du + dt < t \le et\\ 0, t < st, t > et \end{cases}$$
(17)

The pre-task set is defined as tP, and the follow-up task set is defined as tN. When $tP = \emptyset$, dt_1 is defined as the distance between UAV i and the currently executed task, ad is the expected task execution time, and $st = \max\{t, ad + dt_1\}$. On the contrary, when $tP \neq \emptyset$, t_p is defined as the start time of the prior task, t_{pd} is the execution time of the prior task, dt_2 is the distance between the locations of prior and current tasks, and $s_t = \max\{t, t_p + t_{pd} + dt_2\}$. t_{ce} is defined as the end time of the current task, dt_3 represents the distance between the follow-up and current task locations, t_n is defined as the start time of the follow-up task, t_{du} is defined as the execution time of the current task. When $tN = \emptyset$, $r = t_{ce}$. When $tN \neq \emptyset$, $r = \min\{t_{ce}, t_n - t_{du} - dt_3\}$. The algorithm flow is as follows.

Step 1: Firstly, obtain the current task, its predecessor and the subsequent task set. Calculate the earliest execution time range *st* and the latest execution time range *et*;

Step 2: Calculate the of *M* every task at time *t*;

Step 3: Adjust the assignment of tasks according to the size of *M*;

Step 4: Judge whether the allocation target $\sum_{j \in N} M_j(t) u_{ij}$ reaches the minimum. If so,

go to Step 5; otherwise, go to Step 3;

Step 5: Output the adjusted allocation scheme;

Step 6: Judge whether there is still a need for allocation adjustment; if so, go to Step 1; otherwise, go to Step 7;

Step 7: The adjustment and distribution are finished.

Combining the above steps, the implementation process for dynamic adjustment of tasks based on a time window is as shown in Figure 3:



Figure 3. Assignment process.

3. Results

3.1. Simulation Experiment Settings

Aiming at the problem of multi-target, unbalanced, phased, collaborative task assignment of heterogeneous multi-UAVs, when the numbers of UAVs and targets are inconsistent, the number of UAVs is allocated according to the requirements at different stages of the target task. To solve this problem, simulation verification of the proposed method was carried out. In order to facilitate the simulation experiment for the problem, the unit of task allocation time in the simulation scenario of the following case is seconds, and the unit of flight distance in kilometers and in (4) and (5) in the performance testing section of the algorithm in Section 3.3.2, the time unit is 100 s. The UAV object studied in the simulation experiment is considered a mass point, and there are research indicators about the UAV, which mainly include position coordinates, flight speed, and mission execution time. The UAV object studied in the simulation and the relevant research indicators in this paper are consistent with the references [15,26]. The hardware configuration used for the algorithm simulation experiments in this chapter is Intel(R) Core(TM) i7-9750HF CPU @ 2.60GHz Processor, memory is 32 GB, and the graphics card is NVIDIA GeForce GTX 1060 Ti. The software environment is MATLAB 2020a. The simulation settings are as follows.

Scenario 1: A formation of UAVs performs combat missions. There are five UAVs of two types: two type-I UAVs and three type-II UAVs. The two types of UAVs can execute tasks in cooperation for different mission types. Combat missions are mainly divided into two categories: reconnaissance and strike. There are four of each in this scenario. The initial situation is given in the following Tables 1 and 2:

Teals Identity	Task Coordinates –	Task Type	
lask identity		Reconnaissance Class	Strike Class
1	(-1.3,2.6)	\checkmark	
2	(0.3,2.7)		
3	(-1.1,2.5)		
4	(1.1,4.4)		
5	(0.3,4.5)	\checkmark	
6	(0.7,5.5)	\checkmark	
7	(1.7,5.2)	\checkmark	
8	(0.7,2.0)		\checkmark

Table 1. Task target information.

Table 2. UAV information.	Table 2.	UAV	information.
---------------------------	----------	-----	--------------

UAV Identity	Initial Position	Ability Type	Maximum Number of Tasks That Can Be Executed	Speed (m/s)
1	(0.05,3.3)	Ι	5	30
2	(1.4,5.3)	Ι	5	35
3	(1.9, -0.9)	II	5	50
4	(-0.9, 5.4)	II	5	40
5	(2.0,2.3)	II	5	30

For the operational replanning of cooperative task execution by multiple UAVs, we must conduct cooperative task redistribution as unexpected new tasks arise. For this problem, simulation verification of the proposed method was carried out. The simulation scenario was set as follows.

Scenario 2: UAVs perform combat missions in formation. There are five UAVs of two types: two Type-I and three Type-II. At first, the operational tasks carried out in the course of operations are mainly divided into two types: reconnaissance and strike, of

which there are six and nine, respectively. Type-I UAVs perform reconnaissance missions, while Type-II UAVs perform strike missions. One new task is suddenly generated, and its corresponding execution relationship can be executed by the two types of UAVs. The specific initial situation is listed in the following Tables 3 and 4.

Tack Identity		Task Type		
lask identity	Task Coordinates	Reconnaissance Class	Strike Class	
1	(-1.3,2.1)			
2	(-0.6,3.0)			
3	(1.0,3.3)			
4	(2.4,4.2)			
5	(-1.1,2.5)			
6	(-0.5, 2.0)			
7	(1.7,2.8)			
8	(-0.3,0.8)			
9	(0.6, -1.1)			
10	(1.2,4.4)	\checkmark		
11	(1.7,0.8)			
12	(-1.8, -1.0)			
13	(-1.5,0.5)			
14	(-1.3,2.8)			
15	(-1.9,3.1)			

Table 3. Task objective information.

Table 4. UAV information.

UAV Identity	Initial Position	Ability Type	Maximum Number of Tasks That Can Be Executed	Speed (m/s)
1	(0.1,3.3)	Ι	5	30
2	(1.4,5.2)	Ι	5	35
3	(2.0, -1.0)	II	5	50
4	(-0.9, 5.4)	II	5	40
5	(2.1,2.3)	II	5	30

3.2. Simulation Results and Analysis

Simulation Scenario 1: According to the initial state of each target in Scenario 1, we use the improved CBBA collaborative target allocation algorithm to allocate targets as follows:

As shown in Figure 4, the overall execution time for the UAV formation to complete the collaborative tasks is 362.634 s, with the UAV formation taking 254.634 s to make a task decision and complete all task assignment processes when faced with the current task environment. UAVs A1 and A5 arrived at the position of reconnaissance mission one in 31 s and 37 s, respectively, when they performed the mission in coordination, and the mission execution time was 15 s. UAVs A1 and A5 cooperated to perform the task in phases during execution periods {31,46} and {37,52} of Reconnaissance Task 1. After reconnaissance, they flew from the current mission point to the position of strike mission two, and they started their mission in 69 s and 69.7 s, respectively, and they cooperated to perform the task during execution periods {69,74} and {69.7,74.7} of Strike Task 2, respectively. The initial coordinated mission of the UAVs A3 and A4 was reconnaissance mission six, and they arrived at the mission point at 35 s and 32 s, respectively, both with a mission execution time of 5 s. UAVs 3 and 4 cooperated to perform the task in phases during execution periods (35,40) and (32,37) of Reconnaissance Task 6. After reconnaissance, UAV A4 flew from the current mission point to the position of strike mission four and carried out this mission in conjunction with UAV A2, which arrived at the position of strike mission four at 39 s. UAVs 2 and 4 cooperated to perform the task in phases during the execution periods {32,47} and {39,54} of Strike Task 4, respectively. After that, UAV A2 and UAV A4 cooperated to perform reconnaissance mission five, and their respective time to reach the mission position were 67 s and 72 s, respectively. UAVs 2 and 4 cooperated with each other in phases during execution periods {67,82} and {72,87} of reconnaissance mission five, respectively. After reconnaissance, UAV A1 took 23 s to fly from the position of strike mission two to the position of strike mission three. UAVs 1 and 4 cooperated with each other in execution periods {97,102} and {98,103} of strike mission six to the position of reconnaissance mission seven. UAV A3 took 27 s to fly from the position of reconnaissance mission seven in {67,82}. After the reconnaissance, UAV A5 flew from the position of strike mission two to the position of strike mission of strike mission eight at 84 s. UAVs 3 and 5 cooperated with each other to carry out the mission in stages during execution time periods {93,108} and {84,99} of strike mission eight, respectively. To sum up, it can be seen that the proposed algorithm can well solve the problem of multi-UAV, multi-task, unbalanced, cooperative, and phased task assignment.



Figure 4. Task assignment result. (a) Task execution result; (b) UAVs task schedule.

Simulation Scenario 2: The proposed task dynamic adjustment mechanism is simulated and tested. In the case of no new tasks, the original task allocation route and assignment timing result are shown in Figure 5.



Figure 5. Task assignment result. (a) Original execution result; (b) UAVs task schedule.

As shown in Table 5, the overall execution time for the UAV formation to complete the coordinated tasks is 345.884 s, with the UAV formation taking 230.884 s to make a task decision and complete the process of assigning all tasks in the face of the current task environment. In the absence of new tasks, the execution by the UAVs is as follows: UAV 1

performs the type-I task sequence as $15 \rightarrow 11 \rightarrow 14 \rightarrow 12$, the execution time of various tasks is 5 s, and the time when UAV A1 arrives at the initial task point and starts to perform the task is 10 s. For its subsequent task set, the time when the UAV starts to execute the task is 64 s, 76 s, and 86 s, respectively. The time to complete the overall task is 91 s. UAV 2 performs the type-I task sequence as $13 \rightarrow 10$, the execution time for each type of task is 5 s, where the moment when UAV A2 arrives at the initial task point and starts the task is 15 s, after which it takes 25 s to fly from that task point to the location of task 10, the moment when the UAV starts that task is 45 s, and the overall task execution is completed in 50 s. UAV 3 performs the type-II task sequence as $9 \rightarrow 6$, the execution time for each type of task is 15 s, where the moment when UAV A3 arrives at the initial task point and starts its task is 18 s, after which it takes 34 s to fly from that task point to the location of task 6, the moment when the UAV starts that task is 67 s and the overall task execution is completed in 82 s. UAV 4 performs the type-II task sequence as $5 \rightarrow 2 \rightarrow 1$, the execution time for each type of task is 15 s, where the moment when UAV A4 arrives at the initial task point and starts its task is 10 s, and for its subsequent set of tasks, the UAV starts its execution at 47 s and 69 s, respectively, and the overall task completion time is 84 s, and UAV 5 performs the type-II task sequence as $7 \rightarrow 8 \rightarrow 4 \rightarrow 3$, the execution time for each type of task is 15 s, where the moment when UAV A5 arrives at the initial task point and starts its task is 21 s, and for its subsequent set of tasks, the UAV starts its execution at 49 s, 83 s, and 99 s, respectively, and the overall task completion time is 114 s. The total revenue of task allocation is 1456.

Table 5. Original task assignment results of UAV.

UAV Identity	Task Execution Path	Flight Distance	Task Execution Time
1	$(15,I) \rightarrow (11,I) \rightarrow (14,I) \rightarrow (12,I)$	16	{10,15},{64,69},{76,81},{86,91}
2	(13,I)→(10,I)	9	{15,20},{45,50}
3	(9,II)→(6,II)	5	{18,33},{67,82}
4	$(5,II) \rightarrow (2,II) \rightarrow (1,II)$	14	{10,25},{47,62},{69,84}
5	$(7,II) \rightarrow (8,II) \rightarrow (4,II) \rightarrow (3,II)$	20	{21,37},{49,64},{83,98},{99,114}

When a new task appears, its task allocation route and assignment timing result change as follows:

(1) New tasks occur when multiple UAVs perform tasks in the early stage, as shown in the following simulation (Figure 6):



Figure 6. Task assignment result. (a) Improved execution result; (b) UAVs task schedule.

As shown in Table 6, the overall execution time for the UAV formation to complete the coordinated tasks is 157.003 s, with the UAV formation taking 43.003 s to make a task decision and complete the process of assigning all tasks in the face of the current task

environment. The time of the emergence of a new task is {45,55}. After the emergence of a new task, the improved CBBA is used to change the execution of all UAVs based on the initial task allocation scheme: UAV 1 carries out type-I task sequence as $15 \rightarrow 14 \rightarrow 16$, the execution time for each type of task is 5 s, where the moment when UAV A1 arrives at the initial task point and starts its task is 10 s, and for its subsequent set of tasks, the UAV starts its execution at 76 s and 82 s, respectively, and the overall task completion time is 87 s; UAV 2 carries out type-I task sequence as $13 \rightarrow 10 \rightarrow 11 \rightarrow 12$, the execution time of the new task 10 is 10 s, the execution time of the remaining types of tasks is 5 s, the moment when UAV A2 arrives at the initial task point and starts executing the task is 15 s, and for its subsequent set of tasks, the UAV starts executing at 45 s, 64 s and 86 s, respectively, and the overall task execution completion time is 91 s; UAV 3 carries out type-II task sequence as $9 \rightarrow 8 \rightarrow 6 \rightarrow 4$, the execution time for each type of task is 15 s, with UAV A3 arriving at the initial task point and starting its task at 18 s, and for its subsequent set of tasks, the UAVs start execution at 49 s, 67 s and 83 s, respectively, and the overall task completion time is 98 s; UAV 4 carries out type-II task sequence as $5 \rightarrow 2 \rightarrow 1$, the execution time for each type of task is 15 s, where the moment when UAV A4 arrives at the initial task point and starts its task is 10 s, and for its subsequent set of tasks, the UAV starts its execution at 47 s and 69 s, respectively, and the overall task completion time is 84 s; And UAV 5 carries out type-II task sequence as $7 \rightarrow 3$, the moment when UAV A5 arrives at the initial task point and starts the task is 21 s and the task execution time is 16 s, after which it takes 62 s to fly from this task point to the location of task 3, the moment when the UAV starts this task is 99 s and the task execution time is 15 s, and the overall task execution completion time is 114 s. The total revenue of the UAV mission allocation is 5215.3.

(2) New tasks occur in the late stage of multi-UAV missions, as shown in the simulation below (Figure 7):

Table 6. UAV dynamic task assignment resul
--

UAV Identity	Task Execution Path	Flight Distance	Task Execution Time
1	(15,I)→(14,I)→(16,I)	10	{10,15},{76,81},{82,87}
2	$(13,I) \rightarrow (10,III) \rightarrow (11,I) \rightarrow (12,I)$	15	{15,20},{45,55},{64,69},{86,91}
3	$(9,II) \rightarrow (8,II) \rightarrow (6,II) \rightarrow (4,II)$	21	{18,33},{49,64},{67,82},{83,98}
4	$(5,II) \rightarrow (2,II) \rightarrow (1,II)$	14	{10,25},{47,62},{69,84}
5	(7,II)→(3,II)	15	{21,37},{99,114}



Figure 7. Task assignment result. (a) Improved execution result; (b) UAVs task schedule.

As shown in Table 7, the overall execution time for the UAV formation to complete the coordinated tasks is 216.206 s, with the UAV formation taking 102.206 s to make a mission decision and complete the process of assigning all tasks in the face of the current mission environment. The time of the emergence of a new task is {82,92}. The improved CBBA is used to change the execution of all UAVs based on the initial task allocation scheme: UAV

1 carries out type-I task sequence as $15 \rightarrow 10 \rightarrow 14 \rightarrow 16$, the execution time of the sudden new task 16 is 10 s, the execution time of the remaining types of tasks is 5 s, the moment when UAV A1 arrives at the initial task point and starts executing the task is 10 s, for its subsequent set of tasks, the moments when the UAV starts executing are 45 s, 76 s and 82 s respectively, and the overall task execution is completed in 92 s; UAV 2 carries out type-I task sequence as $13 \rightarrow 11 \rightarrow 12$, the execution time for each type of task is 5 s, where the moment when UAV A2 arrives at the initial task point and starts its task is 15 s, and for its subsequent set of tasks the UAV starts its execution at 64 s and 86 s, respectively, and the overall task completion time is 91 s; UAV 3 carries out type-II task sequence as $9 \rightarrow 8 \rightarrow 6 \rightarrow 4$, the execution time for each type of task is 15 s, with UAV A3 arriving at the initial task point and starting its task at 18 s, and for its subsequent set of tasks, the UAVs start execution at 49 s, 67 s and 83 s, respectively, and the overall task completion time is 98 s; UAV 4 carries out type-II task sequence as $5 \rightarrow 2 \rightarrow 1$, the execution time for each type of task is 15 s, where the moment when UAV A4 arrives at the initial task point and starts its task is 10 s, and for its subsequent set of tasks, the UAV starts its execution at 47 s and 69 s, respectively, and the overall task completion time is 84 s; And UAV 5 carries out type-II task sequence as $7 \rightarrow 3$, the moment when UAV A5 arrives at the initial task point and starts the task is 21 s and the task execution time is 16 s, after which it takes 62 s to fly from this task point to the location of task 3, the moment when the UAV starts this task is 99 s and the task execution time is 15 s, and the overall task execution completion time is 114 s. The total revenue of the UAV mission allocation is 5283.3.

Table 7. UAV	dynamic	task assignment	results
--------------	---------	-----------------	---------

UAV Identity	Task Execution Path	Flight Distance	Task Execution Time
1	$(15,I) \rightarrow (10,I) \rightarrow (14,I) \rightarrow (16,I)$	16	{10,15},{45,50},{76,81},{82,92}
2	$(13,I) \rightarrow (11,I) \rightarrow (12,I)$	11	{15,20},{64,69},{86,91}
3	$(9,II) \rightarrow (8,II) \rightarrow (6,II) \rightarrow (4,II)$	21	{18,33},{49,64},{67,82},{83,98}
4	$(5,II) \rightarrow (2,II) \rightarrow (1,II)$	14	{10,25},{47,62},{69,84}
5	(7,II)→(3,II)	15	{21,37},{99,114}

3.3. Algorithm Performance Evaluation

3.3.1. Performance Evaluation of Multi-UAV and Multi-Target Collaborative Task Assignment in Stages

In order to test the influence of different combinations of UAVs performing tasks collaboratively at different times on the task allocation algorithm, six UAVs are set up to assign tasks to 24 targets, with equal numbers of UAV types and task types. Among them, there are two types of tasks, one is to assign UAVs to cooperate in three phases at different times, and the other is to assign UAVs to cooperate in two phases at different times. The specific simulation results are as follows.

As shown in Figure 8, UAVs 1, 2 and 4 cooperate with each other in execution time periods {20.9,25.9}, {26,31}, and {23.8,28.8} of Reconnaissance Task 1. After completing the reconnaissance task, UAVs 1 and 4 are in the execution time periods {55.3,60.3} and {51.8, 56.8} of Strike Task 2, respectively. In the same way, it can be seen that different UAVs cooperated to complete other target tasks. It can be seen that target tasks with different cooperative execution stages can be achieved at different times. Different UAVs cooperate in executing the target tasks in combination, and the allocation result of the algorithm is not affected by the combination of target tasks with different requirements of execution stages, which well the problem of the numbers of UAVs and target tasks being unbalanced and the target tasks needing to be executed cooperatively in stages.



Figure 8. Task assignment result. (a) Task execution result; (b) UAVs task schedule.

3.3.2. Dynamic Allocation Algorithm Performance Evaluation

(1) In order to further compare the performance of dynamic task allocation by the CBBA, two new tasks were added on the basis of Scenario 1 for comparative simulation, and the results were compared with the CBBA's complete reallocation strategy. In the CBBA used for the complete reallocation strategy, the UAV meets the task execution requirements, clears its original task assignment, and reconstructs its task package and task execution path for the new task, finally achieving the dynamic assignment of the new task. The reassignment result of the CBBA's complete reallocation strategy is shown in Figure 9.



Figure 9. Task reassignment result. (a) Original reassignment result; (b) UAVs task schedule.

As shown in Table 8, the overall execution time for the UAV formation to complete the collaborative tasks is 362.093 s, with the UAV formation taking 248.093 s to make a task decision and complete all task assignment processes when faced with the current task environment. The times of the emergence of new tasks are {10,20} and {82,92}. After the emergence of new tasks, the implementation of UAVs using CBBA for complete redistribution based on the initial task allocation scheme changes as follows: UAV 1 carries out type I task sequence as $15\rightarrow11\rightarrow14\rightarrow16$, the execution time of the sudden new tasks 15 and 16 is 10 s, the execution time of the remaining types of tasks is 5 s, the moment when UAV A1 arrives at the initial task point and starts executing the task is 10 s, for its subsequent set of tasks, the UAV starts executing at 64 s, 76 s and 82 s respectively, and the overall task execution completion time is 92 s; UAV 2 carries out type I task sequence as $13\rightarrow10\rightarrow12$, the execution time for each type of task is 5 s, the moment when UAV A2 arrives at the initial task point and starts is 5 s, for its subsequent set of tasks, the UAV starts task is 15 s, for its subsequent set of tasks, the UAV starts executing at 64 s, 76 s and 82 s respectively, and the overall task execution completion time is 92 s; UAV 2 carries out type I task sequence as $13\rightarrow10\rightarrow12$, the execution time for each type of task is 5 s, the moment when UAV A2 arrives at the initial task point and starts its task is 15 s, for its subsequent set of tasks, the moment when the UAV starts its execution is 45 s and 86 s respectively, and the time when the overall task is completed is 91 s; UAV 3 carries out type II task sequence as $9 \rightarrow 6$, the moment when UAV A3 arrives at the initial task point and starts its task is 18 s, after which it takes 34 s to fly from this task point to the position of task 6, the moment when the UAV starts this task is 67 s, the execution time for each type of task is 15 s and the overall task execution is completed in 82 s; UAV 4 carries out type II task sequence as $5\rightarrow 2\rightarrow 1$, the execution time for each type of task is 15 s, the moment when UAV A4 arrives at the initial task point and starts executing the task is 10 s, for its subsequent set of tasks, the UAV starts executing at 47 s and 69 s respectively, and the overall task execution is completed in 84 s; And UAV 5 carries out type II task sequence as $7\rightarrow 8\rightarrow 4\rightarrow 3$, the moment when UAV A5 arrives at the initial task point and starts its task is 21 s, with a task execution time of 16 s. For its subsequent set of tasks, the UAVs start execution at 49 s, 83 s, and 99 s, with an execution time of 15 s for each type of task, and the overall task execution is completed in 114 s. The total revenue of the UAV mission allocation is 1552.4.

(2) For the same scenario, the specific situation of task continuity adjustment allocation in the improved CBBA is as shown in Figure 10.

Table 8. UAV mission execution relationship.

UAV Identity	Task Execution Path	Flight Distance	Task Execution Time
1	$(15,III) \rightarrow (11,I) \rightarrow (14,I) \rightarrow (16,III)$	18	{10,20},{64,69},{76,81},{82,92}
2	$(13,I) \rightarrow (10,I) \rightarrow (12,I)$	10	{15,20},{45,50},{86,91}
3	(9,II)→(6,II)	13	{18,33},{67,82}
4	$(5,II) \rightarrow (2,II) \rightarrow (1,II)$	14	{10,25},{47,62},{69,84}
5	$(7,II) \rightarrow (8,II) \rightarrow (4,II) \rightarrow (3,II)$	22	{21,37},{49,64},{83,98},{99,114}



Figure 10. Task reallocation result. (a) Improved reassignment result; (b) UAVs task schedule.

As shown in Table 9, the overall execution time for the UAV formation to complete the collaborative tasks is 362.093 s, with the UAV formation taking 248.093 s to make a task decision and complete all task assignment processes when faced with the current task environment. The times of the emergence of new tasks are {10,20} and {82,92}. After the emergence of new tasks, compared with the CBBA's complete reallocation strategy, the changes in the task execution sequence after using the improved CBBA with the task continuity adjustment allocation strategy are as follows: UAV 1 carries out type I task sequence as $15\rightarrow16$, the moment when UAV A1 arrives at the initial task point and starts its task is 10 s, after which it takes 62 s to fly from that mission point to the location of task 16, the moment when the UAV starts that task is 82 s, the execution time for each type of task is 10 s, and the overall task execution completion time is 92 s; UAV 2 carries out Type I task sequence as $13\rightarrow10\rightarrow11\rightarrow14\rightarrow12$, the execution time for each type of task is 5 s, the moment when UAV A2 arrives at the initial task point and starts executing the task is 15 s, for its subsequent set of tasks, the UAV starts executing at 45 s, 64 s, 76 s and 86 s respectively, and the overall task completion time is 91 s; UAV 3 carries out Type II task sequence as $9 \rightarrow 8 \rightarrow 6 \rightarrow 4$, the execution time for each type of task is 15 s, the moment when UAV A3 arrives at the initial task point and starts its task is 18 s, for its subsequent set of tasks the UAV starts its execution at 49 s, 67 s and 83 s respectively, and the overall task completion time is 98 s; UAV 4 carries out Type II task sequence as $5 \rightarrow 2 \rightarrow 1$, the execution time for each type of task is 15 s, the moment when UAV A4 arrives at the initial task point and starts executing the task is 10 s, for its subsequent set of tasks, the UAV starts executing at 47 s and 69 s respectively, and the overall task execution is completed in 84 s; And UAV 5 carries out Type II task sequence as $7 \rightarrow 3$, the moment when UAV A5 arrives at the initial task point and starts the task is 21 s and the task execution time is 16 s, after which it takes 62 s to fly from this task point to the location of task 3, the moment when the UAV starts this task is 99 s and the task execution time is 15 s, and the overall task execution completion time is 114 s. The total revenue of the UAV mission allocation is 5251.5.

(3) For the same scenario test, the CBBA was used to add the adjustment allocation based on the task time window on the basis of the former test, as shown in Figure 11.

Table 9. UAV mission execution relationship.

UAV	' Identity	Task Execution Path	Flight Distance	Task Execution Time
	1	(15,III)→(16,III)	9	{10,20},{82,92}
	2	$(13,I) \rightarrow (10,I) \rightarrow (11,I) \rightarrow (14,I) \rightarrow (12,I)$	19	{15,20},{45,50},{64,69},{76,81},{86,91}
	3	$(9,II) \rightarrow (8,II) \rightarrow (6,II) \rightarrow (4,II)$	16	{18,33},{49,64},{67,82},{83,98}
	4	$(5,II) \rightarrow (2,II) \rightarrow (1,II)$	14	{10,25},{47,62},{69,84}
	5	(7,II)→(3,II)	10	{21,37},{99,114}



Figure 11. Task reassignment result. (a) Improved reassignment result; (b) UAVs task schedule.

As shown in Table 10, the overall execution time for the UAV formation to complete the collaborative tasks is 224.013 s, with the UAV formation taking 100.013 s to make a task decision and complete the process of assigning all tasks in the face of the current task environment. The times of the emergence of new tasks are {10,20} and {82,92}. After the emergence of new tasks, the implementation of UAVs using the task continuity adjustment allocation in the improved CBBA and the adjustment allocation based on the time window change based on the task allocation scheme of the complete redistribution strategy CBBA: UAV 1 carries out type I task sequence as $15 \rightarrow 10 \rightarrow 14 \rightarrow 16$, the execution time of the new tasks 15 and 16 is 10 s, the execution time of the remaining types of tasks is 5 s, the moment when UAV A1 arrives at the initial task point and starts executing the task is 10 s, for its subsequent set of tasks, the moments when the UAV starts executing are 45 s, 76 s and 82 s respectively, and the overall task execution is completed in 92 s; UAV 2 carries out type I task sequence as $13 \rightarrow 11 \rightarrow 12$, the execution time for each type of task is 5 s, the moment when UAV A2 arrives at the initial task point and starts executing the task is 15 s, for its subsequent set of tasks the UAV starts executing at 64 s and 86 s respectively, and the overall task completion time is 91 s; UAV 3 carries out type II task sequence as $9 \rightarrow 8 \rightarrow 6 \rightarrow 4$, the execution time for each type of task is 15 s, the moment when UAV A3 arrives at the initial task point and starts its task is 18 s, for its subsequent set of tasks the UAV starts its execution at 49 s, 67 s and 83 s respectively, and the overall task completion time is 98 s; UAV 4 carries out type II task sequence as $5\rightarrow 2\rightarrow 1$, the execution time for each type of task is 15 s, the moment when UAV A4 arrives at the initial task point and starts executing the task is 10 s, for its subsequent set of tasks, the UAV starts executing at 47 s and 69 s respectively, and the overall task execution is completed in 84 s; And UAV 5 carries out type II task sequence as $7\rightarrow 3$, the moment when UAV A5 arrives at the initial task point and starts the task is 21 s and the task execution time is 16 s, after which it takes 62 s to fly from this task point to the location of task 3, the moment when the UAV starts this task is 99 s and the task execution time is 15 s, and the overall task execution completion time is 114 s. The total revenue of the UAV mission allocation is 5419.2.

(4) To further test the performance of the improved CBBA in unbalanced staged collaborative task assignments for multiple UAVs and targets, 1-20 UAVs were set up for collaborative execution of 15 tasks. The task assignment performance was compared with those of the original CBBA and contract network algorithms. The results are shown in Figure 12.

Table IV. UAV IIIISSIOII EXECUTIOII IETATIOIISIII	Table 10.	UAV	mission	execution	relationship	p
---	-----------	-----	---------	-----------	--------------	---

UAV Identity	Task Execution Path	Flight Distance	Task Execution Time
1	$(15,III) \rightarrow (10,I) \rightarrow (14,I) \rightarrow (16,III)$	11	{10,20},{45,50},{76,81},{82,92}
2	$(13,I) \rightarrow (11,I) \rightarrow (12,I)$	12	{15,20},{64,69},{86,91}
3	$(9,II) \rightarrow (8,II) \rightarrow (6,II) \rightarrow (4,II)$	16	{18,33},{49,64},{67,82},{83,98}
4	$(5,II) \rightarrow (2,II) \rightarrow (1,II)$	14	{10,25},{47,62},{69,84}
5	(7,II)→(3,II)	10	{21,37},{99,114}



Figure 12. Comparison of algorithm performance tests. (**a**) Algorithm benefit comparison; (**b**) Algorithm running time comparison.

As shown in Figure 12, compared with the original CBBA and contract network algorithms, the improved CBBA has a higher profit value when the task allocation reaches the final saturation state. This is because the original CBBA and contract network algorithms do not consider the imbalance between the numbers of UAVs and target tasks. At the same time, the assignment of target tasks with different execution stages is not satisfied well. The running time of the CBBA is higher than that of the original CBBA and contract net algorithms. This is because the improved CBBA is much more considerate of the matching

relationship between UAVs and tasks and reasonably allocates UAVs to perform tasks in different task execution stages. In summary, it can be seen that the improved CBBA has better allocation performance in solving the problem of multi-UAV and multi-target unbalanced coordinated task allocation in stages.

(5) To further test the applicability and effectiveness of the algorithm in UAV dynamic task assignment scenarios, 1 to 18 UAVs were assigned to 16 target tasks, including nine type I tasks, five type II tasks, and two dynamic tasks. The UAV type ratio was 1:1. The simulation test is shown in Figure 13.



Figure 13. Comparison of algorithm performance tests. (**a**) Algorithm benefit comparison; (**b**) Algorithm running time comparison.

As shown in Figure 13, in the process of UAVs performing multiple tasks, the benefit of the improved algorithm is higher than that of the algorithm of the complete reprogramming strategy. In the early stage, due to there being a small number of UAVs and too many tasks, the benefits of the TCW-CBBA, TW-CBBA, and TC-CBBA algorithms are roughly the same. When the number of UAVs is approximately 3–7, the profit values of the algorithms are ranked from high to low as TCW-CBBA, TW-CBBA, and TC-CBBA. In the later period, due to the excessive number of UAVs and the relatively small number of tasks, the total profit values of multiple UAVs tend to be the same. In terms of running time, the TCW-CBBA, TW-CBBA, and TC-CBBA algorithms taking longer than TC-CBBA because the TCW-CBBA and TW-CBBA algorithms taking longer than TC-CBBA because the TCW-CBBA and TW-CBBA algorithms consider the effect of the task window when assigning tasks.

4. Discussion

To solve the problem of multi-UAV and multi-target cooperative task assignment while considering the constraints of multi-type task assignment and mismatched numbers of UAVs and target tasks, the CBBA was improved to achieve task assignment by stages. The improved CBBA was used to improve the dynamic task allocation process of multi-UAVs from two aspects: continuous task planning and dynamic adjustment of task execution time windows. The simulation results show that the proposed multi-UAV cooperative multi-objective phased assignment algorithm and multi-UAV dynamic task assignment problem and enable timely response to unexpected new tasks. The next step will focus on the joint planning of task execution time windows among different UAVs.

5. Conclusions

For the multi-UAV phased cooperative task allocation problem, the multi-UAV multitask cooperative phased allocation model was established by considering the constraints of task execution timing constraint, task execution time constraint, task execution quantity constraint, and task execution demand constraint. An improved consensus-based bundle algorithm was proposed to improve the algorithm task packet and task execution path construction mechanism, to realize the cooperative execution of multiple tasks by multiple UAVs, and to increase the design of a two-tier task bidding mechanism to better meet the execution needs of various tasks in different time periods. Aiming at the problem of task reassignment of sudden new tasks in the process of task execution, on the basis of the original complete replanning strategy, a task continuity adjustment mechanism and a task time window-based adjustment mechanism were added. The task continuity adjustment mechanism is used to comprehensively weigh the execution relationship between preorder tasks and subsequent tasks in the task execution path and better take into account the relationship between the overall task execution benefits and execution costs. Use the adjustment mechanism based on the task time window to reasonably plan the task execution time arrangement of various tasks in the task execution path of various UAVs, optimize the overall task execution list, and improve the task execution efficiency. Through specific simulation cases, it is verified that the proposed cooperative phased task assignment algorithm can well realize the coordinated execution of different tasks by multiple UAVs in different time periods and meet the execution requirements of various time periods of the tasks. At the same time, the improved dynamic task allocation mechanism proposed in this paper can well deal with the allocation conflict between the original task sequence and the sudden new task and better take into account the task execution benefits and execution costs in the task execution path after replanning. This makes the overall task execution time window planning and arrangement more reasonable and effectively improves the overall task dynamic allocation efficiency and effectiveness.

Author Contributions: Conceptualization, W.W. and L.R.; methodology, W.W. and M.L.; software, W.W. and B.L.; formal analysis, W.W. and M.L.; writing-original draft preparation, W.W.; writingreview and editing, W.W., X.C. and S.H.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Young Talent Fund of Association for Science and Technology in Shaaxi, China under grant 20220101.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are reflected in the paper, if there is a need, please contact us.

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

Nomenclature

- The assignment of UAV *i* to task *j*. x_{ij}
- Lt The maximum number of tasks that each UAV can perform.
- NTThe number of tasks.
- NU The number of UAVs.
- Ι The UAV set.
- I The task set.
- L_i The maximum number of UAVs.
- The maximum number of tasks that each UAV can perform.
- The starting times of the reconnaissance mission.
- $\begin{array}{c} L_t \\ t_r^{T_j} \\ t_c^{T_j} \end{array}$ The starting times of the strike mission.
- Et_i The start execution time.
- Lt_i The end execution time.
- The *k*th execution period $|et_i^k, lt_i^k|$ of the *j*th target. $t^{j}_{et^{k}_{i},lt^{k}_{i}}$
- The distribution scheme. x_i
- p_i The task execution sequence.
- The time spent by UAV *i* to perform task *j*. t_{ij}

- Et_i The time for UAV *i* to start its mission.
- Lt_i The time for UAV *i* to end its mission.
- t_i The time when the UAV *i* performed the mission.
- b_i The task package.
- p_i The task path.
- z_i The winner set.
- y_i The winning bid price set.
- s_i The timestamp.
- $S_i^{p_i}$ The total revenue of UAV *i*'s task allocation.
- c_{ij} The revenue from task execution.
- t_{ijs} The start time of UAV *i* performing task *j*.
- t_{js} The start time of task *j*.
- t_{je} The end execution time of task *j*.
- $\dot{\zeta}_{ij}$ The mission distance cost of the UAV.
- d_{ij} The straight-line distance between UAV *i* and task *j*.
- γ The distance cost coefficient.
- K_{ij} The profit function coefficient of UAV *i* performing task *j*.
- B_{ij} The UAV *i*'s bid for mission *j*.
- T_i The starting execution time set.
- t^* The difference in the start execution times of adjacent tasks.
- τ_r The time of message receipt.
- g_{ik} The information interaction between UAV *i* and UAV *k*.
- *T* The time for new tasks to appear.
- T_i The task execution sequence of UAV *i*.
- T_t The conflict-free task set at any execution time after t.
- *h* The order of the task in the execution sequence.
- *g* The individual UAV in the task execution sequence to which the task belongs.
- *w* The total number of conflict-free tasks.
- d^- The minimum threshold of the cost ratio.
- d^+ The maximum threshold of the cost ratio.
- *AS* The algorithm iteration counting notation.
- u_{ij} The execution relationship between UAV *i* and task *j*.
- *du* The execution time of a task.
- *st* The earliest start time of a task.
- *et* The latest start time of a task.
- *dt* The time that the UAV takes to reach the task location.
- *M* The proportion of the task time window of the UAV to start the task at time *t* in the overall time window.
- *P* The time set of all tasks that can be started within the time window.
- *tP* The pre-task set.
- *tN* The follow-up task set.
- *ad* The expected task execution time.
- dt_1 The distance between UAV *i* and the currently executed task.
- *dt*₂ The distance between the locations of prior and current tasks.
- *dt*³ The distance between the follow-up and current task locations.
- t_p The start time of the prior task.
- t_{pd} The execution time of the prior task.
- t_{ce} The end time of the current task.
- t_n The start time of the follow-up task.
- t_{du} The execution time of the current task.

References

- 1. Elmeseiry, N.; Alshaer, N.; Ismail, T. A Detailed Survey and Future Directions of Unmanned Aerial Vehicles (UAVs) with Potential Applications. *Aerospace* 2022, *8*, 363. [CrossRef]
- 2. Eaton, C.M.; Chong, E.K.P.; Maciejewski, A.A. Multiple-Scenario Unmanned Aerial System Control: A Systems Engineering Approach and Review of Existing Control Methods. *Aerospace* **2016**, *3*, 1. [CrossRef]
- Liu, X.Y.; Yan, C.; Zhou, H. Towards Flocking Navigation and Obstacle Avoidance for Multi-UAV Systems through Hierarchical Weighting Vicsek Model. *Aerospace* 2021, 8, 286. [CrossRef]

- 4. Jia, Z.; Yu, J.; Ai, X.; Xu, X.; Yang, D. Cooperative multiple task assignment problem with stochastic velocities and time windows for heterogeneous unmanned aerial vehicles using a genetic algorithm. *Aerosp. Sci. Technol.* **2018**, *76*, 112–125. [CrossRef]
- 5. Fan, C.; Han, S.; Li, X.; Zhang, T.; Yuan, Y. A modified nature-inspired meta-heuristic methodology for heterogeneous unmanned aerial vehicle system task assignment problem. *Soft Comput.* **2021**, *25*, 14227–14243. [CrossRef]
- Kumar, D.; Baranwal, G.; Raza, Z.; Vidyarthi, D.P. A Novel Hybrid Auction Algorithm for Multi-UAVs Dynamic Task Assignment. IEEE Access 2020, 8, 86207–86222.
- Shima, T.; Rasmussen, S.; Gross, D. Assigning micro UAVs to task tours in an urban terrain. *IEEE Trans. Control Syst. Technol.* 2007, 15, 601–612. [CrossRef]
- Koerkamp, N.K.; Borst, C.; Mulder, M.; van Paassen, M.M. Supporting Humans in Solving Multi-UAV Dynamic Vehicle Routing Problems. *IFAC* 2020, 52, 359–364. [CrossRef]
- Hong, Y.; Jung, S.; Kim, S.; Cha, J. Multi-UAV Routing with Priority using Mixed Integer Linear Programming. In Proceedings of the 2020 20th International Conference on Control, Automation and Systems (ICCAS), Busan, Korea, 13–16 October 2020; pp. 699–702.
- Zardashti, R.; Yazdanpanah, M.J.; Nikkhah, A.A. Nonlinear Multiobjective Time-Dependent TF/TA Trajectory Planning Using a Network Flow-Based Algorithm. J. Aerosp. Eng. 2016, 29, 04015041. [CrossRef]
- 11. Shi, J.; Tan, L.; Lian, X.; Xu, T.; Zhang, H.; Zhang, Y. A multi- unmanned aerial vehicle dynamic task assignment method based on bionic algorithms. *Comput. Electr. Eng.* 2022, 99, 107820. [CrossRef]
- Oh, G.; Kim, Y.; Ahn, J.; Choi, H.L. Market-Based Distributed Task Assignment of Multiple Unmanned Aerial Vehicles for Cooperative Timing Mission. J. Aircr. 2017, 54, 2298–2310. [CrossRef]
- 13. Fu, X.; Feng, P.; Gao, X. Swarm UAVs Task and Resource Dynamic Assignment Algorithm Based on Task Sequence Mechanism. *IEEE Access* **2019**, *7*, 41090–41100. [CrossRef]
- 14. Cao, W.J.; Xu, S.H. A New Multi-UAV Cooperation Method. ISCID 2016, 2, 231–234.
- 15. Zhu, J.Y.; He, Q.F.; Wang, T.Z.; Wang, Z.T. Multiobjective mission planning for UAV under uncertain environment. *MECIT* 2014, 556–562, 4435–4438. [CrossRef]
- 16. Yin, X.; Zhang, K.; Li, B.; Sangaiah, A.K.; Wang, J. A task allocation strategy for complex applications in heterogeneous cluster-based wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 1550147718795355. [CrossRef]
- Ragi, S.; Mittelmann, H.D. Mixed-Integer Nonlinear Programming Formulation of a UAV Path Optimization Problem. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017; pp. 406–411.
- 18. Zhen, Z.Y.; Xing, D.J.; Gao, C. Cooperative search-attack mission planning for multi-UAV based on intelligent self-organized algorithm. *Aerosp. Sci. Technol.* **2018**, *76*, 402–411. [CrossRef]
- Sugano, R.; Takeuchi, N.; Sekiguchi, K.; Nonaka, K. Layer Structured Multiple-UAV System Mixing Centralized and Distributed Agents. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2020; pp. 915–920.
- Kumar, D.; Baranwal, G.; Raza, Z.; Vidyarthi, D.P. Fair Mechanisms for Combinatorial Reverse Auction-Based Cloud Market. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2019; Volume 107, pp. 267–277.
- 21. Zhang, J.; Xing, J.H. Cooperative task assignment of multi-UAV system. Chin. J. Aeronaut. 2021, 33, 2825–2827. [CrossRef]
- Choi, H.L.; Brunet, L.; How, J.P. Consensus-Based Decentralized Auctions for Robust Task Allocation. *IEEE Trans. Robot.* 2009, 25, 912–926. [CrossRef]
- Hunt, S.; Meng, Q.; Hinde, C.J. An Extension of the Consensus-Based Bundle Algorithm for Multi-Agent Tasks with Task Based Requirements. In Proceedings of the 2012 11th International Conference on Machine Learning and Applications, Boca Raton, FL, USA, 12–15 December 2012; Volume 2, pp. 451–456.
- 24. Zhang, Y.; Feng, W.; Shi, G.; Jiang, F.; Chowdhury, M.; Ling, S.H. UAV Swarm Mission Planning in Dynamic Environment Using Consensus-Based Bundle Algorithm. *Sensors* **2020**, *20*, 2307. [CrossRef]
- Bai, X.S.; Cao, M.; Yan, W.S. Event- and time-triggered dynamic task assignments for multiple vehicles. *Auton. Trans. Robot.* 2020, 44, 877–888. [CrossRef]
- 26. Qin, B.; Zhang, D.; Tang, S.; Wang, M. Distributed Grouping Cooperative Dynamic Task Assignment Method of UAV Swarm. *Appl. Sci.* 2022, 12, 2865. [CrossRef]
- Hunt, S.; Meng, Q.; Hinde, C.; Huang, T. A Consensus-Based Grouping Algorithm for Multi-agent Cooperative Task Allocation with Complex Requirements. *Cogn. Comput.* 2014, *6*, 338–350. [CrossRef] [PubMed]
- Chen, X.; Liu, Y.; Yin, L.; Qi, L. Cooperative Task Assignment and Track Planning For Multi-UAV Attack Mobile Targets. J. Intell. Robot. Syst. 2020, 100, 1383–1400. [CrossRef]
- 29. Zitouni, F.; Harous, S.; Maamri, R. A Distributed Approach to the Multi-Robot Task Allocation Problem Using the Consensus-Based Bundle Algorithm and Ant Colony System. *IEEE Access* 2020, *8*, 27479–27494. [CrossRef]
- 30. Yang, M.; Bi, W.; Zhang, A.; Gao, F. A distributed task reassignment method in dynamic environment for multi-UAV system. *Appl. Intell.* **2021**, *52*, 1582–1601. [CrossRef]
- Yang, Y.; Liu, X.X. Task Assignment Based on Improved Dynamic Contract Net and Ant Colony Search Strategy. In Proceedings of the Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), Shenyang, China, 20–22 December 2013; pp. 2880–2883.

- 32. Murray, C.C.; Karwan, M.H. An Extensible Modeling Framework for Dynamic Reassignment and Rerouting in Cooperative Airborne Operations. *Nav. Res. Logist.* 2010, 57, 634–652. [CrossRef]
- 33. Yao, W.; Qi, N.; Wan, N.; Liu, Y. An iterative strategy for task assignment and path planning of distributed multiple unmanned aerial vehicles. *Aerosp. Sci. Technol.* **2019**, *86*, 455–464. [CrossRef]
- Zhang, Z.S.; Liu, H.; Wu, G.H. A Dynamic Task Scheduling Method for Multiple UAVs Based on Contract Net Protocol. Sensors 2010, 22, 4486. [CrossRef]
- 35. Chen, J.; Qing, X.; Ye, F.; Xiao, K.; You, K.; Sun, Q. Consensus-based bundle algorithm with local replanning for heterogeneous multi-UAV system in the time-sensitive and dynamic environment. *J. Supercomput.* **2021**, *78*, 1712–1740. [CrossRef]