

Article

An Adaptive Task Planning Method for UAVC Task Layer: DSTCA

Ting Duan , Qun Li, Xin Zhou *  and Xiaobo Li

College of Systems Engineering, National University of Defense Technology, Changsha 410073, China; duanting18@nudt.edu.cn (T.D.); liquan@nudt.edu.cn (Q.L.); lixiaobo@nudt.edu.cn (X.L.)

* Correspondence: zhouxin09@nudt.edu.cn

Abstract: With the rapid development of digital intelligence, drones can provide many conveniences for people's lives, especially in executing rescue missions in special areas. When executing rescue missions in remote areas, communication cannot be fully covered. Therefore, to improve the online adaptability of the task chain link in task planning with a complex system structure as the background, a distributed source-task-capability allocation (DSTCA) problem was constructed. The first task chain coordination mechanism scheme was proposed, and a DSTCA architecture based on the task chain coordination mechanism was constructed to achieve the online adaptability of the swarm. At the same time, the existing algorithms cannot achieve this idea, and the DSTCA-CBBA algorithm based on CNP is proposed. The efficiency change, agent score, and time three indicators are evaluated through specific cases. In response to sudden changes in nodes in the task chain link, the maximum spanning tree algorithm is used to reconstruct the task chain link in a short time, thereby completing the mission task assigned to the drone entity. Meanwhile, the experimental results also prove the effectiveness of the proposed algorithm.

Keywords: UAV complex; mission chain; adaptive task planning; distribution coordination mechanism



Citation: Duan, T.; Li, Q.; Zhou, X.; Li, X. An Adaptive Task Planning Method for UAVC Task Layer: DSTCA. *Drones* **2024**, *8*, 553. <https://doi.org/10.3390/drones8100553>

Academic Editor: Abdessattar Abdelkefi

Received: 1 August 2024

Revised: 26 September 2024

Accepted: 27 September 2024

Published: 6 October 2024



Copyright: © 2024 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

Since the beginning of this century, the development of war forms and the emergence of networked task systems have put forward new demands and challenges for resource task allocation in complex operations [1]. For the intelligent complex task system, the characteristics of modern warfare and the application of distributed command mode are not enough to describe the status quo of coordinated tasks and the uncertain factors on the battlefield. On the one hand, the acceleration of task tempo and the improvement of weapon maneuvering performance make the battlefield change rapidly, putting high requirements on the real-time performance of the allocation algorithm.

On the other hand, the concept of Mosaic warfare proposes that the hybrid decision-making ability of man-machine integration and the autonomous decision-making and collaborative ability of intelligent equipment means that intelligent equipment can independently recognize the battlefield environment, independently determine the task plan, and independently implement the task plan under the high dynamic and strong confrontation task scenario [2]. Intelligent equipment complex relying on the unified self-organization structure, platform, and mechanism, build the trust ability between heterogeneous multi-platforms, cluster distributed collaborative perception, distributed intelligent decision and distributed fire attack, and finally accelerate the battlefield frontier Observation Orientation Decision Action (OODA) loop operation efficiency [3]. This has inspired the thinking of this paper, which is devoted to studying the autonomous adaptability of UAV complex online mission chains [4].

Vidal [5], an early scholar, proposed that two airframes realize a distributed layered hybrid system architecture, emphasizing the autonomy of the system. In 2004, M.Dias proposed a centralized global optimization scheme and proposed the problem of multi-machine collaboration under a dynamic environment [6], mainly oriented to centralized

allocation. After that, document [7] establishes an organized formation mode tracking and uses a distributed group game controller better to meet the new organization goal of the information-sharing graph. The document [8] considers the perception degradation of UAVs, and a radar tracking model is proposed to improve the perception technology to assist decision-making. document [9] uses the data-driven distributed formation control method of multi-population evolutionary games to improve control performance. With the development of UAV technology, scholars have begun to study UAV dynamic task allocation and cooperative control in complex environments [10,11], then the intelligent optimization method is used to optimize [12].

Most of the existing studies rely on the center of charge for centralized allocation [13,14]. The general research idea of the centralized allocation problem is to build a model according to the problem, and then choose a more appropriate model to solve it, to achieve a higher algorithm consistency or a shorter time to get the solution of the scheme allocation [15–17]. The distributed problem is to design a collaborative mechanism matching the problem. When the pre-allocated scheme encounters a problem, the distribution node dynamically coordinates the task allocation scheme according to the collaborative mechanism. Dynamic changes in the environment pose a threat to preconceived mission chain schemes. Aiming at specific emergent tasks, literature [18] proposes an emergent threat countermeasure STC mechanism to help the algorithm quickly respond to the task environment changes caused by emergent threats, and Literature [19] designed a chain reconstruction method in which tasks are inserted into chains.

Mission chain adaptability dynamic planning is a task adaptability mechanism in the task domain. Its main objectives are to plan the updated list of task targets online for new task target chains, support chain reconstruction rules, and detect and evaluate the integrity and compatibility of attack chains. Therefore, the trigger conditions are as follows: a new task is issued by the superior, a new target threat is discovered, and a failed node exists on the original chain. Therefore, this paper gradually shifts the research focus from a small number of centralized static distributions to distributed dynamic distributions. In other words, Distributed Source-Task-Capability allocation (DSTCA) is introduced. The research focuses on resource-task-capability allocation to describe complex resource allocation. At the same time, it can meet the new operational requirements in the complex task system and greatly strengthen the adaptability and flexibility of the UAV complex under the premise of the loss of some individuals and other unexpected situations.

2. DSTCA Method Based on Contract Network Protocol

DSTCA belongs to the Distributed Resource Management (DRM) category, the core is the collaboration mechanism between C2, and the research is still in the initial stage [20]. Contract Net Protocol (CNP) is a market mechanism based on auction theory, which has developed into a relatively mature coordination mechanism in MAS coordination technology and has a broad application prospect. Therefore, this chapter takes CNP as a collaborative mechanism for solving DSTCA problems between point domain charge nodes, and combined with the task background, proposes a mission chain-oriented CNP method, and studies and discusses six key questions in DSTCA, namely, bidding conditions, tendering strategy, bidding strategy, winning strategy, protocol mechanism and contract type (Table 1).

Table 1. Parameter description table.

Symbol	Specification
C	protocol
$CL_i(A)$	Task requirement group
A	Environment/situation set
E	Task-System set
W	ResourceSet
$Y(ti)$	Decision variable matrix

Table 1. Cont.

Symbol	Specification
nT	TaskSet
nA	Agents

2.1. Problem Description

DSTCA is the matching situation of all resources and tasks in the case of contract occurrence, denoted as $DSTCA = \langle C, E \rangle$. Let set C represent the mission chain contract network protocol, denoted as $C = \langle A, G \rangle$, A represents a certain assumed environment, and G represents a matching relationship triggered when A occurs. $E = \langle W, Ta \rangle$ indicates the matching set of task resource tasks, and Ta indicates the task set. The hypothetical environment set can be expressed as $A = a_1, a_2 \dots a_{na}$, the operational requirement resource (system) set can be expressed as $W = w_1, w_2 \dots w_{nw}$, and the mission requirement set can be expressed as $nT = nT_1, nT_2 \dots nT_{nt}$. For the hypothetical environment, the combination of task requirements is $CL_i(A) = \langle c_i, W_j, nT_k \rangle$. If there is a definite influence relationship between c_i and W_j , When matching $\langle W_j, nT_k \rangle$, it is matched again according to C . Define $x \in X$ to be a static space of DSTCA, where how to make distributed adjustment make unmanned complex system platform and task match dynamically, make $sw_l \in SW, l \neq k$ and C_j relation, make the following formula be established at t_i moment:

$$maxZ(t_i) = \sum_{i=1}^{Nt} tfd_i(t_i) [1 - \prod_{k=1}^{na} \prod_{j=1}^{nw} P(X(ij), Y(t_i)), ps_{ijk}(t_i)] \tag{1}$$

where $tfd_i(t_i)$ is the task change rate after normalization at t_i time; as decision variable matrix $y_{jk} \in (0, 1), x_{ij} \in (0, 1)$

$$Y(t_i) = Y_{jk}(t_i), j \in [1, na], k \in [1, nw] \tag{2}$$

for the matching between T_i and c_i situation. Suppose that $y_{jk}x_{ij}$ has a 0–1 distribution relationship between nT_k and nA_g under $ci = SW$ determined at time t_i . $y_{jk} = 1$ on the premise of $x_{ij} = 1$ means that the resource w_{nw} is allocated to the task at time t_i . Other cases do not constitute distributive relations $ps_{ijk}(t_i)$ represents the influence probability of A set at time t_i . In addition, the matching of tasks and resources should also meet the constraints of weapon capability, resource storage, and other matching conditions.

2.2. Formalize the Task Package Bundle

The motif is a basic unit with high frequency, uniform distribution, and a simple connection structure. The mission chain of the UAV complex includes detection, control [21,22], and task. Based on UAV functions and combined with the characteristics of UAV complex tasks, it is necessary to establish a function-oriented UAV complex modular model. Based on the DoDAF2.0 framework, this paper uses the idea of five functional modules of UAVs and further divides the task entities of the multi-UAVs complex into five types: cluster maneuver, reconnaissance and detection, information processing, decision making, and action taking.

Among them, the cluster maneuver mission chain contains multiple complex maneuver task entities, the reconnaissance and exploration mission chain contains multiple reconnaissance and exploration task entities, the information processing mission chain contains multiple information processing task entities, the decision-making mission chain contains multiple decision-making task entities and the action taking mission chain contains multiple action task entities.

To make use of predetermined marked data and node attribute information for chain planning and prediction, scholars have proposed several supervised algorithms [23,24], which combine network structure information with node and edge-level attributes and

develop a learning algorithm named supervised random walk for chain prediction in social networks. The adaptive planning of the target mission chain studied in this paper is a kind of task network, which is a heterogeneous network HCN with multiple nodes and chain types [25].

$Bundle_l (l \in [1, [loiter]_n])$ Each Bundle contains task packages with different functions, defined as $Bundle_l = \langle TLa, TLb, TLc, TLd, TLe; T \rangle$, according to the mission task decomposition, the specific task package is obtained, and the UAV is assigned to the task package, which contains multiple task entities. $Bundle_l$ does not have the function of assigning a single task and supports intra-pack structure reorganization, and entity bidding to create a new task, update the task pack, and add to the mission chain.

2.3. Constraint Condition

Aiming at online adaptive dynamic task planning of UAV complex chain, this paper designs some constraints as follows:

Constraint 1: If the equipment platform is replaced or added with a component that is most critical to the decision-making of the complex system, The decision time before and after replacement and addition is t_w and t'_w , the contribution rate of the equipment system w to the complex system is:

$$R'_w = (t'_w - t_w) / t_w \quad (3)$$

Constraint 2: The dynamic agent is set as the entity executing the task, $nA \subseteq [1, n]$, and the assignment of task xx shall meet the following requirements:

$$D_i(x_i(nT), x_j(nA)) := \min ||x_j(nA) - x_i(nT)|| - R_d < 0 \quad (4)$$

where R_d is the minimum distance between the agent and target during task assignment.

Constraint 3: According to the order of target type, target model, target state, and target behavior, the threat degree of the target is determined according to the target behavior.

Constraint 4: The task completion cost dependent on environment state A includes the cost of the optimal control problem completed at time $t = T$, where the optimal control input is $u_i^* \rightarrow U_i : [0, T]$ and the minimization function J_i :

$$J_i(u_i(k); D_0, x_i(nT)) = \int_0^t f(x_i(t), u_i(t)) dt \quad (5)$$

Constraint 5: In the total task allocation cost of task nT , for nT_{nt} , given a D_0 , the total utility should follow:

$$U_{nT_i}(nA; D_0) = r_{nT_i} - \rho_{nT_i} R_{nT_i}(nA, D_0; D_{nT_i}) \quad (6)$$

Constraint 6: The calculation of the efficiency ratio shall follow:

$$Xf = (S_d + R) / J_i(u_i(k); D_0; x_i(nT)) \quad (7)$$

3. Task Chain Contract Network Protocol(TC-CNP) Method

3.1. Based on the Basic Idea of CNP

From the perspective of decision-makers, the task allocation process based on CNP mainly includes four stages: Tender, Bidding, Winning, and signing, as shown in Figure 1. Each agent can simultaneously serve two roles, namely, Tenderer and Bidder. In the CNP-based task assignment process, participants are composed of task allocators and recipients and make decisions as bidders and tenderers according to their respective tendering mechanisms and bidding mechanisms [26]. Different problems can specify different tendering and bidding mechanisms to achieve the optimal task resource allocation scheme. As for the DSTCA problem, the adjustment of the distribution scheme needs to meet certain conditions, and the specific execution ways of the tendering mechanism and bidding mechanism will be different under different conditions.

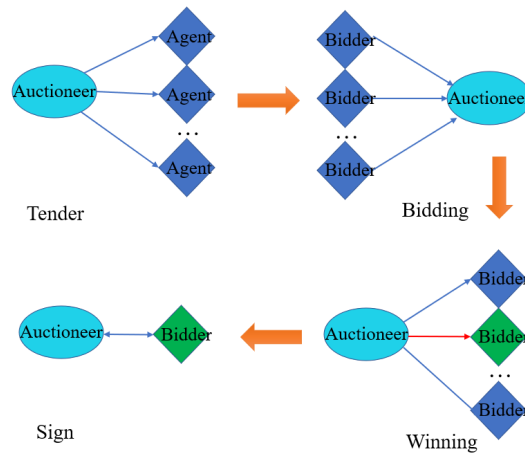


Figure 1. The basic execution process diagram of CNP.

3.2. TC-CNP Method

Contract network protocol network is more suitable for solving distributed problems [27], based on CNP, many algorithms are proposed to solve the task allocation problem, including CBBA, MDP, and model prediction [28–30]. The main purpose of task assignment is to maximize the efficiency of task execution through the interaction and negotiation between agents [31,32]. The auction mechanism is embedded in it, mainly so that each agent can compete for tasks fairly and freely when interacting [33–35].

Under the network structure, the unmanned complex equipment platform can exercise a variety of new integrated task capabilities. The integrated task capability is an important way to improve the system efficiency and also puts forward the demand for the dynamic structure adjustment of the complex system. The main purpose of DSTCA is three aspects. First, if the UAV node currently performing the task fails, the control relationship between tasks and system resources in the system can be quickly adjusted to maintain the survivability of the complex system and improve its adaptability of the complex system. The second is to expand the effective task area of a complex task, in the effective tasks area, as early as possible to complete the task, and enhance the task capability; The third is to improve task accuracy, reduce attack behavior error, and increase the probability of killing the target. Based on the agent cooperation mechanism in the contract network protocol, a new mission chain-based contract network protocol based on the DSTCA model is proposed by considering the individual’s ability to contribute to the task system.

For $\forall sc_i \in SC$, if it exists, the system will automatically trigger the default trigger conditions for the target assigned by the system, and the DSTCA model shall be executed when the following seven immediate effectiveness protocols are met for task execution.

Contract 1: If there are two alternative nodes u_i that are flat and there is no accusative organizational structure relationship, the node with a larger residual capacity is regarded as having a larger contribution rate, and the node with a larger residual capacity is selected for attack.

$$C_1(u_i, sc_i) :< State(u_i, R'_W), State(sc_i, R'_W) > \rightarrow true/false. \tag{8}$$

Contract 2: If difficulty RFL-3 occurs during task execution, the reduction rate of residual value is less than the conventional rate, and the reduction rate of fuel consumption is greater than the average, the help message will be released;

$$C_2(yh_i, v_i) :< State(yh_i, T), State(V_i, T) > \rightarrow true/false. \tag{9}$$

Contract 3: When the target task set of W_i changes, RWL-1 and the DSTCA allocation scheme change at the current moment, a more suitable alternative node c_j is selected within a certain period to form a new task combination.

$$C_3(c_i, W_j, E_l) :< State(c_i, T), State(W_j, T), State(E_l, T) > \rightarrow true/false. \tag{10}$$

Contract 4: If the new target and the original target conflict detection RWL-4, no conflict, directly added to the own target task list; If conflict detection finds a conflict, the original target will continue to be executed and the information will be released again to iterate the idle node. If there is no idle node, immediate efficacy contract 6 will be executed.

$$C_4(Ta_{nt}, CL_i) :< State(aT_{nt}, T), State(CL_i, T) > \rightarrow true / false. \quad (11)$$

Contract 5: If there is a new target node, RWL-3, each node will evaluate its own residual capacity value, and the one with the largest residual value will have priority. COA planning will be reissued and contract 7 will be executed later.

$$C_5(sc_i, List_j, C_6) :< State(sc_i, R'_W), State(List_j, R'_W), State(C_6, R'_W) > \rightarrow true / false. \quad (12)$$

Contract 6: If there is a new target beyond the original plan, or the node suddenly fails, re-planning and other cases of RFL-2, meet the normal speed of use of fuel consumption and capability value, and the minimum time cost is regarded as the largest contribution rate and the highest reward value.

$$C_6(CL_i, List_j, cost_k) :< State(CL_i, C_2), State(List_j, C_2), State(cost_k, T) > \rightarrow true / false. \quad (13)$$

Contract 7: If the node fails RWL-4, the current node will publicly release information and inform itself whether it has advanced authorization node E_l . If yes, it will send authorization change instructions to the replacement node; otherwise, it will not send; Specific content implementation performance contract 7;

$$C_7(E_l, C_6) :< State(E_l, T), State(C_6, T) > \rightarrow true / false. \quad (14)$$

3.3. Capability-Based Bids

A bidding strategy is a way for the bidder to quickly find a suitable contract partner, while the battlefield situation is complex, the environment changes dynamically at any time, and the time window is relatively small. For the DSTCA problem, the bidding scope should be reduced as far as possible. This paper proposes a Capability-based bidding strategy under the constraint of the adaptive protocol. According to the specific functions required by the task point, the task replaceable node set is narrowed down, and then the bidding specification is sent to the appropriate node.

The UAV complex task entities include complex maneuver task entity, reconnaissance and detection task entity, information processing task entity, decision-making task entity, and action-taking task entity, which have their respective functions as shown in Table 2.

Table 2. Task entity table.

Type	Swarm Maneuver Task Entity	Reconnaissance Probe Task Entity	Information Dealing Task Entity	Decision Making Task Entity	Take Action Task Entity
Symbol	TLa	TLb	TLc	TLd	TLe
Function	Reconnaissance, strike, decision	Reconnaissance, strike	Information transmission, monitoring	Control and command	Reconnaissance and attack

For target nT_i , the task matrix can be expressed as:

$$\Sigma Ta_l(nT_i; C) >= 1, (l = 1 || l == 0) \quad (15)$$

where, the indicates that Ta under task i , the matrix that meets task requirements and mission chain contract network protocol is greater than 1.

Similarly, the system matrix can be expressed as:

$$\Sigma E_{en}(t; W_i) \geq 1, (en = 1 || en == 0) \quad (16)$$

Satisfy that each system provides more than one capability, each task activity selects more than one system, and different capabilities satisfy linear summation.

The function matrix represents that UAV has decision-making, detection and control, observation and attack, and other functions. Let's call it gn ,

$$\forall gn \in E, (gn = 1 || gn = 0), \sum gn_{E_{en}} \geq 1 \quad (17)$$

The main influencing factors include orientation Angle, where, the pitch Angle, fuel value RY , task damage value ZS , weapon remaining number SY , and position information WZ .

$$QJ_j = \begin{cases} QJ_j + RY_j \wedge \neg ZS_j \wedge SY_j \wedge WZ_j; nT \notin \emptyset \\ QJ_j - RY_j \wedge \neg ZS_j \wedge SY_j \wedge WZ_j; nT \in \emptyset \end{cases} \quad (18)$$

The overall capability value is determined by the UAV's position information and where, RY , ZS , SY and WZ information, where in WZ information is affected by orientation Angle, pitch Angle and height:

$$WZ_j = Y_{tili-pitch}^i \oplus C_{directed}^j \oplus H_j \quad (19)$$

where, Individual capability GT_j is a correlation function based on individual functions of UAV, including reconnaissance capability ZC_c , strike capability DJ_c , information transmission capability XI_c , command and control capability ZK_c . The calculation method is as follows:

$$ZC_c(nT_j, Ta_l, T) = \begin{cases} 1; C_{self}(RY_j \wedge WZ_j \wedge \neg ZS_j) \vee C_{time}(nT_j, Ta_l, T) \\ 0; else \end{cases} \quad (20)$$

$$DJ_c(nT_j, Ta_l, T) = \begin{cases} 1; C_{self}(RY_j \wedge SY_j \wedge WZ_j \wedge \neg ZS_j) \vee C_{time}(nT_j, Ta_l, T) \\ 0; else \end{cases} \quad (21)$$

$$XI_c(nT_j, Ta_l, T) = \begin{cases} 1; C_{self}(RY_j \wedge \neg ZS_j) \vee C_{time}(nT_j, Ta_l, T) \\ 0; else \end{cases} \quad (22)$$

$$ZK_c(nT_j, Ta_l, T) = \begin{cases} 1; C_{self}(RY_j \wedge WZ_j \wedge \neg ZS_j) \vee C_{time}(nT_j, Ta_l, T) \\ 0; else \end{cases} \quad (23)$$

3.4. Tendering Strategy Based on TC-CNP

The bid value was set as the weight value of the edge, and tendering was conducted according to the bid value to realize mission chain reconstruction the Bidder was tested to see whether the range was met, and bidders were allowed to participate in tendering if the range was met. Tenderer can find target sets $RW^*(nT_j) \in D_{rws}$ for tendering bids based on current information about other nodes:

$$RW^*(nT_j) = [rw_j \in Bundle_l \wedge \exists QJ_j > \alpha \wedge GT_j \notin \emptyset]. \quad (24)$$

Only when the individual capability GT_j is not empty and the global capability QJ_j meets the tendering threshold, the tendering value for the target nT_j can be calculated.

For $\forall rw_j \in Bundle_l$, tendering cannot be conducted because C_i cannot be satisfied, so tendering can be conducted according to the following formula:

$$nT_j^* = \operatorname{argmax} U_i(nT_j) \quad (25)$$

$$U_i(nT_j) = U_i[S_i(nT_j)] - U_i(S_i) \quad (26)$$

The tendering strategy is given as shown below the Algorithm 1.

Algorithm 1 The tendering strategy $Bids_S$

Input: $RW^*; nT_j$; Current status information

Output: $Bids \leftarrow B(nT_j)$

if: $Bi(nT_j) \leq 0$

$bijl = 0$

elseif:

$bijl = 1$

 Find S_i and return; & Compute the $U_i(nT_j)$

 The updated label value is based on formula 25

 Update S_i

endif

end

3.5. Bidding Strategy Based on Maximum Spanning Tree

Bidding mainly selects appropriate Bids from bidders, relies on the function of the winning bidder, the bid value is set as the weight value of the edge, and bids according to the size of the bid value to realize mission chain reconstruction. Bidders can participate in the bidding according to their ability measurement if they meet the range. Bidders bid according to the current information status of other nodes. The key point of the winning strategy is to select the best task executor according to a certain evaluation function ([bestInEdge]). The bidding strategy is shown in Algorithm 2:

Save the best bid information and the best score value, and save the current position with the best score information.

Algorithm 2 The Bidding strategy W_{Bids_S}

Initialization: $G = \langle V, E \rangle$; root, score

Output: spanning-tree

Initialize: $F \leftarrow []$; $T' \leftarrow []$; score $\leftarrow []$

for: each $v \in V$

do: $bestInEdge \leftarrow \operatorname{argmax}_e = (u,v) \in E \text{ score}[e]$

$F \leftarrow F \cup bestInEdge$

for: each $e \in E$

do: $score'_e \leftarrow score[e] \leftarrow score[bestInEdge]$

if: $T = (V, F)$ is a spanning tree

return

else

$C \leftarrow$ a cycle in F

$G' \leftarrow \text{CONTRACT}(G, C)$

$T' \leftarrow \text{MAXSPANNINGTREE}(G', \text{root}, \text{score}')$

$T \leftarrow (T, C)$

return T

endif

endfor

end

4. Improvement on CBBA Algorithm-DSTCA Results

According to the characteristics and solution process of the DSTCA problem, the TC-CNP solution framework based on contract network protocol is extended to CNP. In the process of UAV complex offensive and defensive tasks, by establishing the immediate effectiveness protocol of the mission chain and designing the auction algorithm based on the immediate effectiveness protocol, the decision-making framework can be known as $Desicion = \langle ES, RW, BS, TA; Bids_S, W_{Bids_S}, FT \rangle$. It is mainly composed of seven parts: bidding conditions, task notice, bidding description, task brief, tendering strategy, bidding strategy, and time.

The principle of action is to calculate the utility of each agent to the target, which is the bid. Based on the bid value proposed by each individual, agents communicate with each other within a certain time range to get the best bid. The most critical point is that they must know each other's bid, which is a discrete task allocation problem. CBBA is a distributed auction protocol that provides a demonstrably good solution to the problem of multi-agent multitasking in local area networks [36,37]. It mainly includes the iteration between two stages: one is the package construction stage, where each agent generates an ordered task package locally, which conforms to the composition of the expected task package and can be directly indexed to the initial function chain; the other is the consensus stage, where conflicts are resolved by using consensus algorithm through the local communication between neighboring agents.

The algorithm is mainly improved based on the CBBA algorithm to detect whether there is a task status according to the update, otherwise start to announce the task, and quickly locate the task package, according to the tendering strategy, and bidding strategy to determine the content of the task book to issue the set of objects, and then determine whether to bid [38]. First of all, it is necessary to determine the radius range of task packages, within which task packages will be reassigned, to ensure that there will not be too many or too few UAV agents involved in task reassignment.

Each Agent determined to participate in the bidding task adds a new task to the task package and then uses the mission chain contract network protocol in Section 3.3 to calculate the highest score as the bidding value. Use the bid values of the bid as the bid set in Section 3.4 to generate the bid strategy. Successful bidding agents complete new tasks without the need for additional consensus stages. For an Agent with a premium outbid, the task is released from the package and updated through communication. The specific algorithm flow is shown in Figure 2.

First, initialize the working parameters, set up the bundle and target range complex, initialize the reward value, Then calculate the score of each Agent (calculate the score through the distance formula, then use the reward-punishment function), then determine whether to participate in the bid and update the bid value by running the CBBA bundle. Return the index task with the best bid by calculating the bid, checking the fitness of each Agent and task, and checking for a match. Then save the best bid task location, save the bidding information, and algorithm flow refers to Algorithm 3 pseudo-code.

The Sale Contract is a new mission released, from the new mission objective System W_j to provide cooperative engagement capability against the nT_j unit, the sale contract is only useful if the buyer can afford the mission.

$$Ui(nT_j)_{sale}^k = \max Ui(nT_j)^k, (k = 1, 2, \dots, nt) \quad (27)$$

Replacement contract is task rescheduling, urgent task processing. The bidding unit does not meet the requirements of the value of the residual capacity, and insufficient capacity is found in the process of executing the task, which is secondary scheduling, so the replacement target needs to be found for replacement.

$$Ui(nT_k)_{swap}^k = \max Ui(nT_j, nT_i), (i, j \in [1, nt]) \quad (28)$$

Exchange contracts exist in emergencies and where the reward value is high.

$$U_i^k_{replace}(nT_j, nT_i) = \max U_{i_{replace}}(nT_j, nT_i), (i, j \in [1, nt]) \quad (29)$$

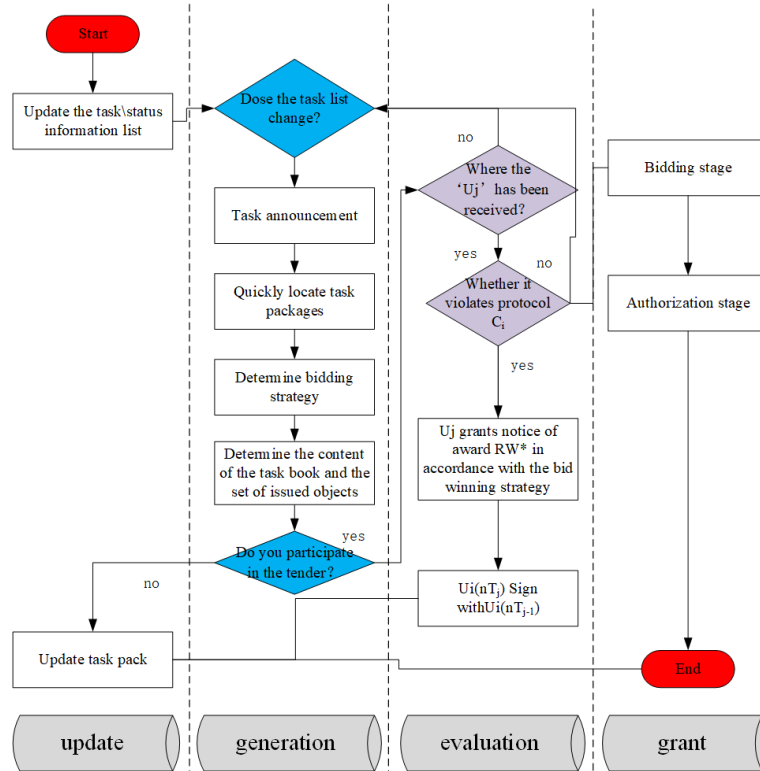


Figure 2. The basic execution process diagram of RW-CBBA.

Algorithm 3 DSTCA-CBBA

Input: initialize working variables

Output: score

```

while(doneflag == 0) & task_idx == 0
    check for conflicts
    (Ta, Ti) ← CBBA – Params, CBBA – Data, Graph, t, T;
for: n = n + 1 ∈ [1, N]
    Bid – values ← (CBBA – Data(n), agents(n), tasks – Ta, n)
    complex ← (R, Ta, Ti > 0)
end
if
else
    T = T + 1; Maintainloop
end
all_scores ← Rij, Ui, x, xlj
UpdateCluster(ci) & MaxSpanningTree
for: each ei ∈ E do
    minDist ← argmin Dist(ei, cj) j ∈ 1, 2, ... k
if minDistance ∉ l(ei)
    then ml(ei) ← minDistance
    changed ← true
end
end
end
end
    
```

5. Discussion

In this paper, the UAV complex task case background is used, and the comprehensive information of A country’s Marines to recover lost land—the island as the experimental basis. The list of UAV complex resources and the number of available UAVs is shown in Table. Dynamic task link adaptive scheme planning is shown in Figure 3.

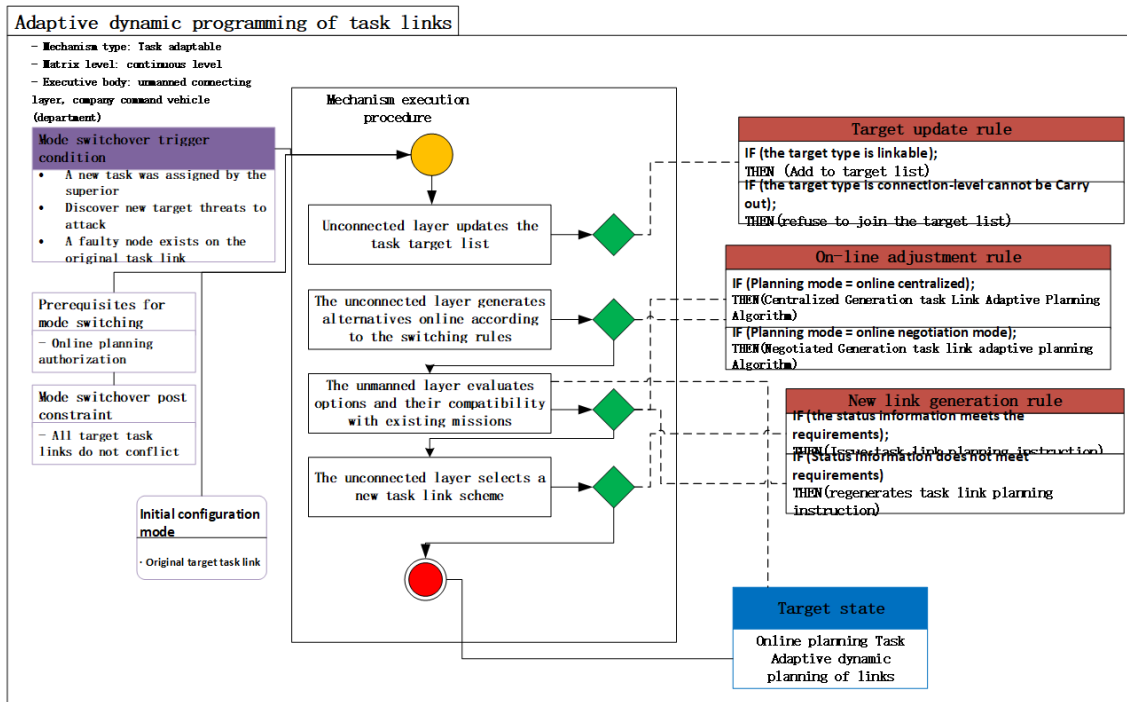


Figure 3. Adaptive dynamic programming of task links.

The command is responsible for the dynamic link adaptability planning, which belongs to the task adaptability mechanism of the task domain. Its main objectives are to update the list of task objectives online, plan for new task target links, support link reconstruction rules, and detect and evaluate the integrity and compatibility of attack links. The triggering conditions are as follows: a new task is assigned from the superior, a new target is discovered, and a faulty node exists on the existing link. The mode-switching rule base includes the target update rule, original task link reconstruction rule, new link generation rule, link integrity, compatibility detection and evaluation rule, etc. The specific steps of the implementation of the mechanism are:

- (1) Update the task target list
- (2) Generate alternatives online according to switching rules
- (3) Evaluate options and their compatibility with existing mandates
- (4) Select a new task link scheme

Based on Section 3, we improved the CBBA algorithm, and the simulation model of multi-task assignment is established. First, 10 UAV entities are selected to perform 20 tasks. The running diagram of the simulation process is shown in Figures 4 and 5. The task cluster was established to improve the efficiency of task allocation and execution. The parameters were set as $n_A = 10, n_T = 20$ and $n_A = 8, n_T = 15$, respectively, and the tests in Table 3 were carried out (Figures 6 and 7). The dotted line in the Figure represents the trajectory process performed by the UAV entity, and the task target is represented by the square entity. The mission packet is represented by a dotted circle, the asterisk is represented by the drone entity, and DSTCA-CBBA determines the direction of the dotted line. Therefore, the normal operation of Figures 4–6 has shown the preliminary implementation of the algorithm simulation. Then the score, utility, and time need to be analyzed and compared.

Table 3. UAVS complex resource table.

Types of Drones	Quantity	Function	Selection Quantity
Reconnaissance UAV	18	Reconnaissance and detection	2
Decision UAV	20	Command, decision, strike	2
Attack UAV	16	strike	3
Reserve UAV	18	Reconnaissance, strike, decision	2
Information processing UAV	5	Information transfer and detection	1

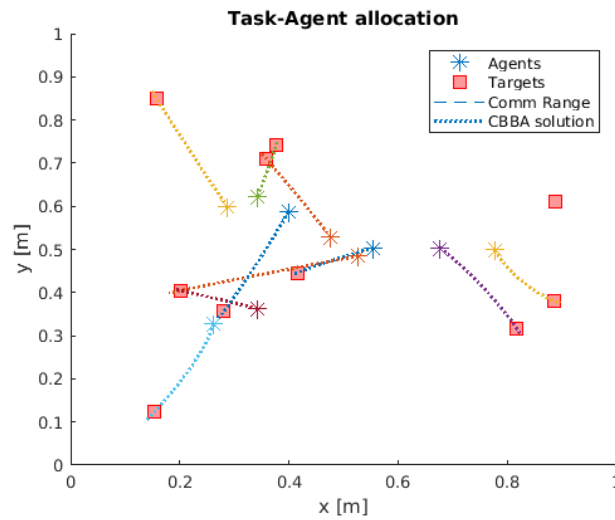


Figure 4. The simulation process diagram for the new task appears.

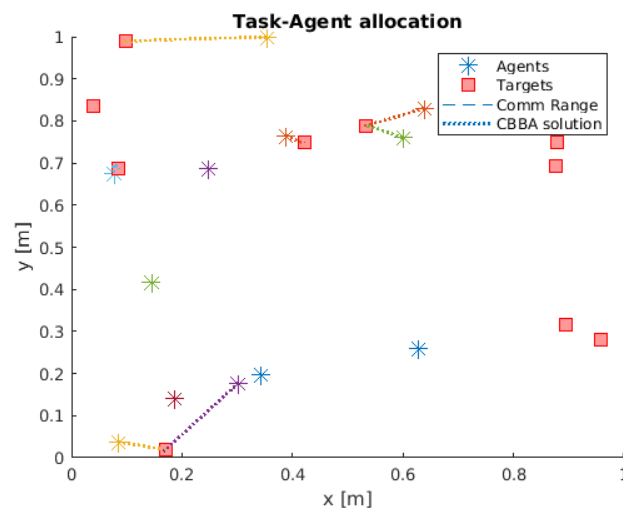


Figure 5. The separate adjustment process of the new task.

At the beginning of the war, the pre-task planning scheme was generated. First, 10 agents are set to perform 10 tasks, and the performance control parameter is $J(Ui(s)) = 1/2 \int_0^t |Ui(T_k)|^2 dt$. Use ode45 differential solver to solve, adjust parameter $lambda(T_k) = 1, k \in [0, 1]$, By changing the location information through setting, the UAV Agent must try to reach the nT_k position at time $t = t_{(f, T_k)}$. *Tasks.task - type* sets the specific type of task, which consists of task entities with different functions. Set, the number of iterations. The task scheduling scheme is shown in Figure 4. Therefore, the DSTCA model is used to calculate task redistribution. As shown in Figures 4 and 5 the parameter value of Agent-task is 10–10, and the parameter setting of 20–20, respectively. The asterisk indicates the agent,

the red square indicates the target, the dotted line indicates the path, and a blue dotted line indicates the CBBA assignment method.

Then, set the second case. The number of Agents is different from the number of tasks. Not every Agent can perform tasks, and some agents are idle. Set parameter $nt = 10, nw = 20$, other parameters remain unchanged, as shown in Figure 6. The change task marked with a red circle is to increase the demand, which requires the assistance of other agents.

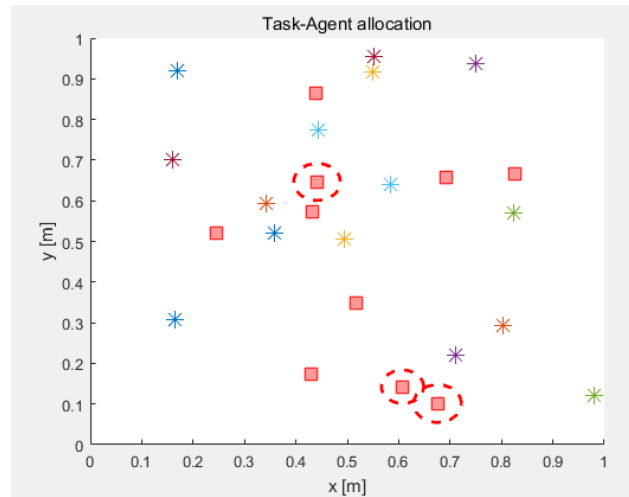


Figure 6. $nA = 10, nT = 20$ simulation process diagram.

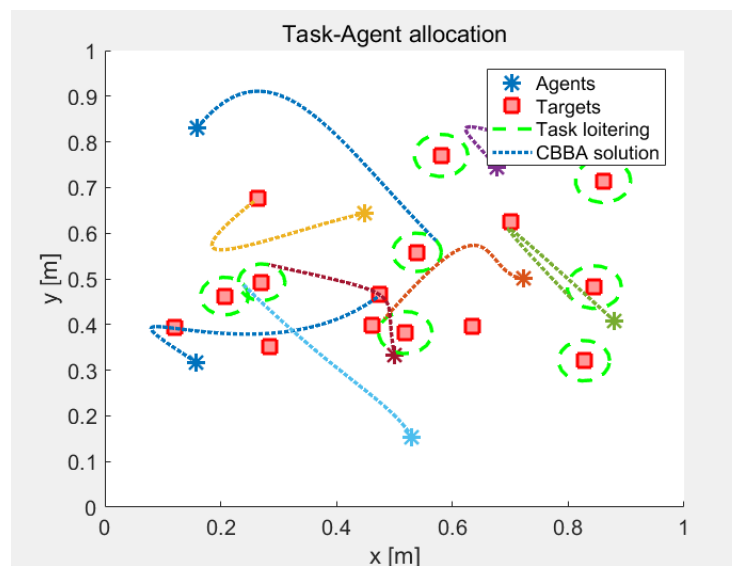


Figure 7. $nA = 8, nT = 15$ simulation process diagram.

The solution framework in Section Capability-Based Bids and Improvement on the CBBA algorithm in Section Bidding Strategy is used to establish the three-dimensional data graph of tasks, agents, and utility, as shown in Figure 7.

By using the solution framework in Section 3 and the improved CBBA algorithm in Section 4, the three-dimensional data graph of tasks, Agents, and time cost is established, as shown in Figures 8–11. The time cost is relatively independent of the number of tasks and Agents. When the number of tasks increases, the global utility value increases linearly with the number of tasks Figure 12, while the increase of utility increases exponentially with the number of Agents. The larger the problem-solving space, the more task requirement w_i needs to be executed. Agents will choose the nearest mission area, and the higher the reward value will be.

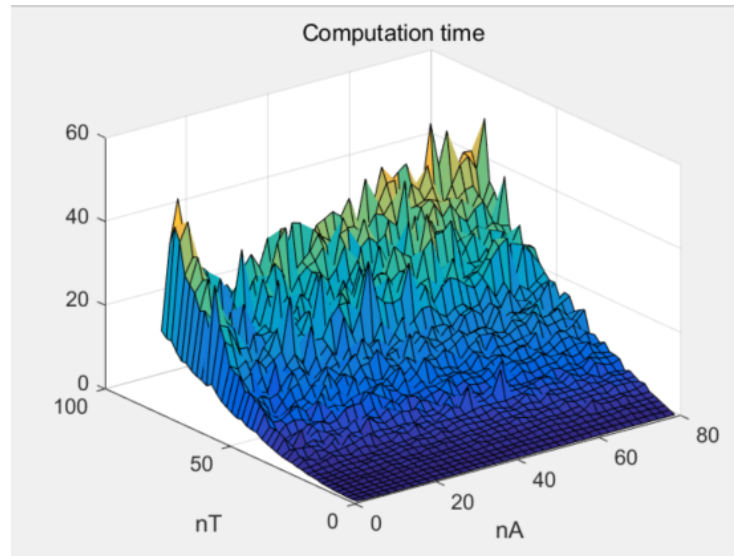


Figure 8. Computation time based on the $nA = 10$, $nT = 20$, $k = 500$.

Next, to verify the validity of the instant-efficacy protocol, it is necessary to further compare and analyze the original algorithm (without setting the instant-efficacy protocol) and the auction algorithm based on the instant-efficacy protocol. The comprehensive evaluation index is used for analysis as follows:

(1) Comparative analysis of task completion time

Firstly, the completion time of the task was compared and analyzed, and the number of iterations was set as $k_f = 100$ (from Figures 13–16). For a reassigned task bid, set the reward range to $reward \in [0, 0.3]$. We use the task execution time in the first column as a comparison, and the results show that the overall running time of the algorithm increases with the number of tasks of the agent, and the time consumption increases slowly. Does not occupy a very large time cost.

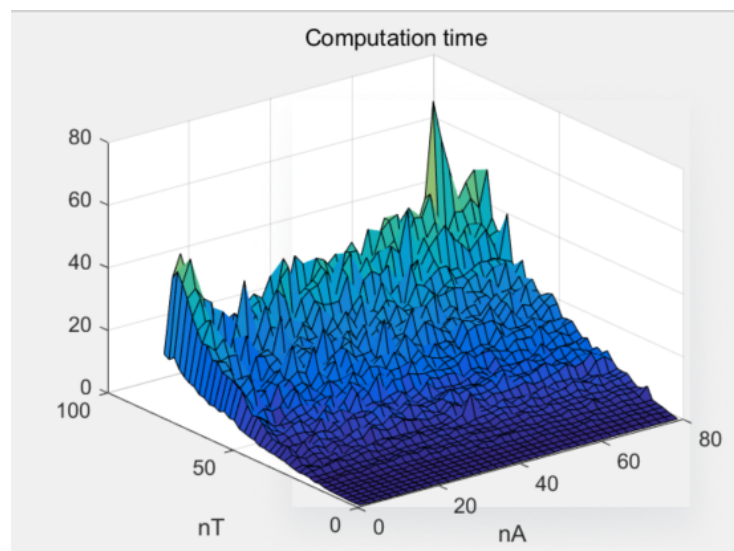


Figure 9. Computation time based on the $nA = 10$, $nT = 20$, $k = 1000$.

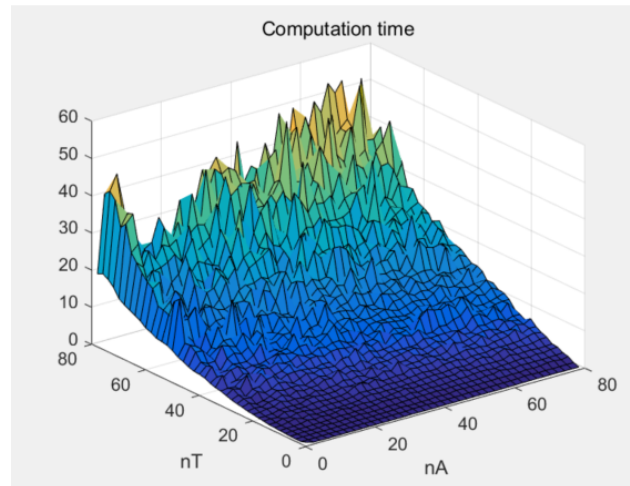


Figure 10. Computation time based on the $nA = 8$, $nT = 15$, $k = 500$.

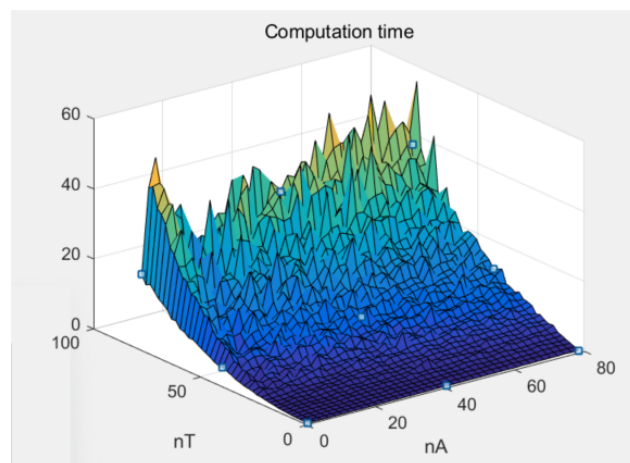


Figure 11. Computation time based on the $nA = 8$, $nT = 15$, $k = 1000$.

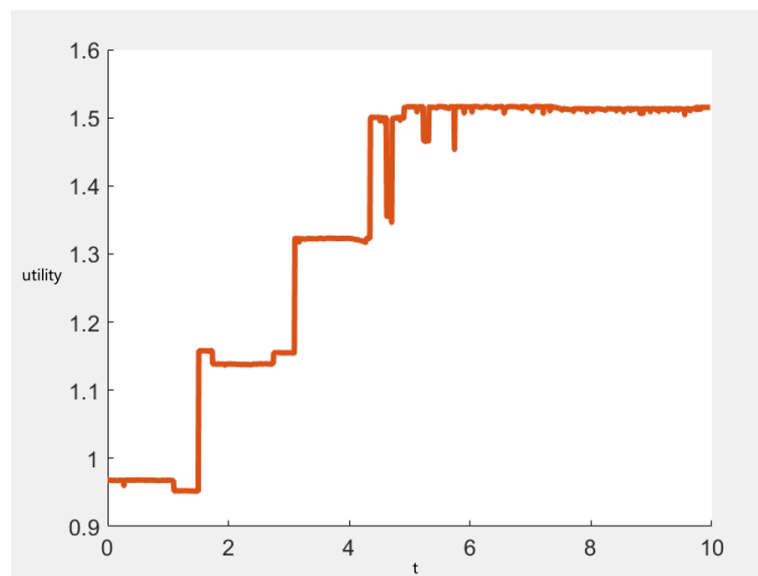


Figure 12. Utility at time t.

Using the control variable method, the number of tasks and Agents and the number of dynamically adjusted tasks remain unchanged. The improved algorithm DSTCA-CBBA

algorithm is compared with the original algorithm CBBA. The experimental results show that with the increase in the number of iterations, the completion time of the DSTCA-CBBA algorithm tends to be stable and lower than that of the original algorithm CBBA. As shown in the figure below, when the parameter of the algorithm is set to 8–15, the algorithm is 1-a for 500 iterations and 1-b for 500 iterations for 10–20. Similarly, when the parameter is set to 8–15, the algorithm is 2-a for 1000 iterations and 2-b for 1000 iterations for 10–20.

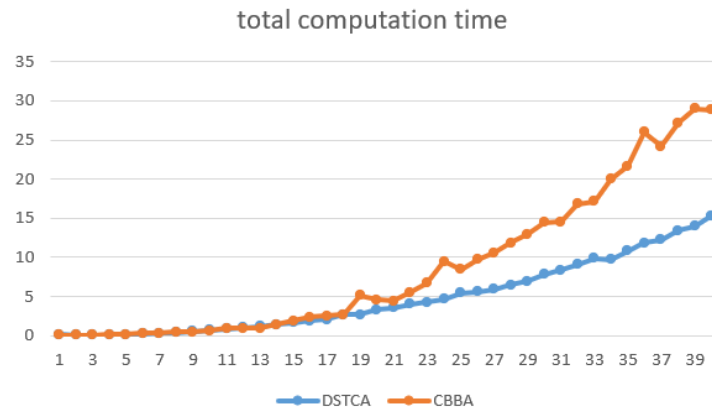


Figure 13. The algorithm iterates comparison graph 1-(a).

(2) Task completion utility comparative analysis.

In the fourth part and the fifth part, two sets of different experimental parameters $nA = 10, nT = 20$ and $nA = 8, nT = 15$ are set respectively. To ensure the completeness of the experiment and test the algorithm index, the total utility calculation convergence of these two sets of parameters is carried out respectively. The experimental results are shown in Figure 17 and 18. The total utility a and b show different utility result values. In the case of different parameter Settings, the Settings of task and agent are 10–20 and 8–15 respectively, and if the Settings are the same color and background cloth, it is easy to mistake that two pictures of the same graph are placed. Therefore, it is carefully observed that although they have different parameter Settings, they all converge to a single vertex. The convergence process is represented by the gradient change, so the utility value can converge to 8 as the task is executed. Then the improved algorithm is compared with the original CBBA algorithm.

Set A3, and set the number of iterations to (Figure 19), (Figure 20), and (Figure 21) respectively. using the real-time efficacy protocol-based CSCTA-CBBA algorithm and the original algorithm CBBA for comparative analysis. Calculate utility analysis for F2 times.

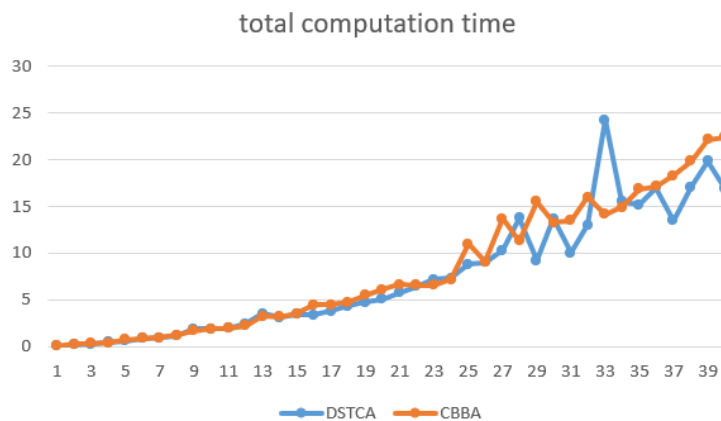


Figure 14. The algorithm iterates comparison graph 1-(b).

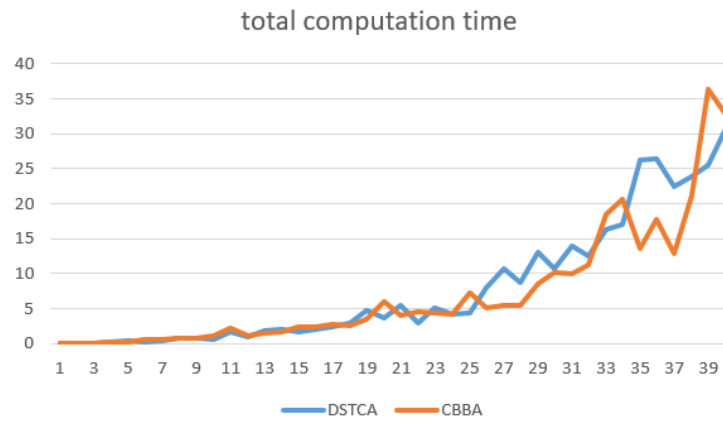


Figure 15. The algorithm iterates comparison graph 2-(a).

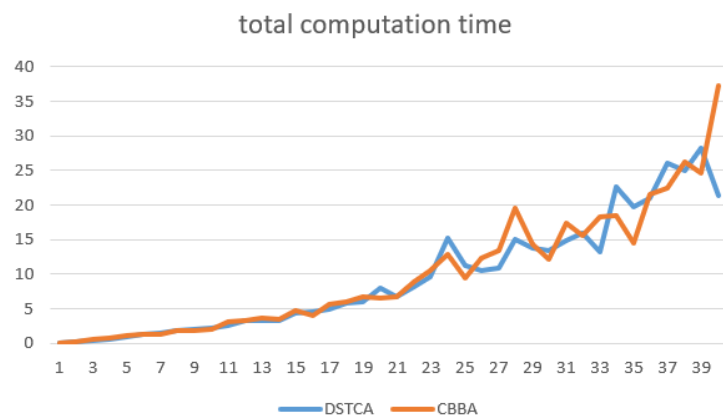


Figure 16. The algorithm iterates comparison graph 2-(b).

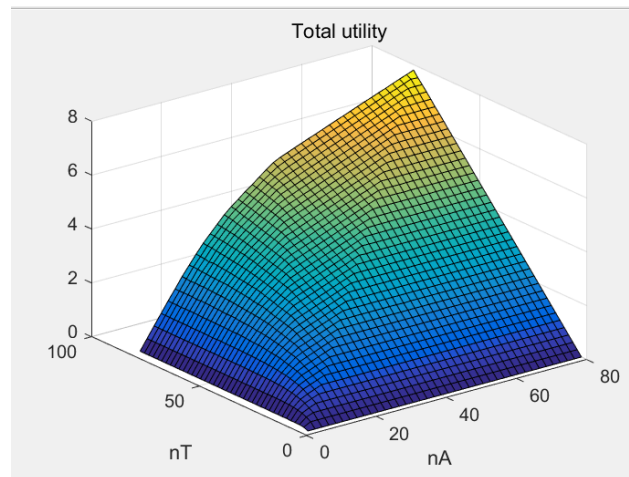


Figure 17. Total Utility (a) with the parameters $nA = 10, nT = 20$.

According to the experimental results, the number of iterations is respectively $k_f = 100$ (Figure 19), $k_f = 500$ (Figure 20), and $k_f = 1000$ (Figure 21). With the same Settings of other parameters, the completion efficiency of the DSTCA-CBBA algorithm is higher than that of the original algorithm CBBA.

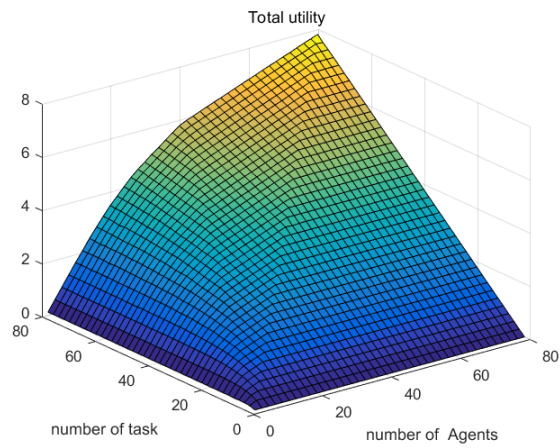


Figure 18. Total Utility (b) with the parameters $nA = 8, nT = 15$.

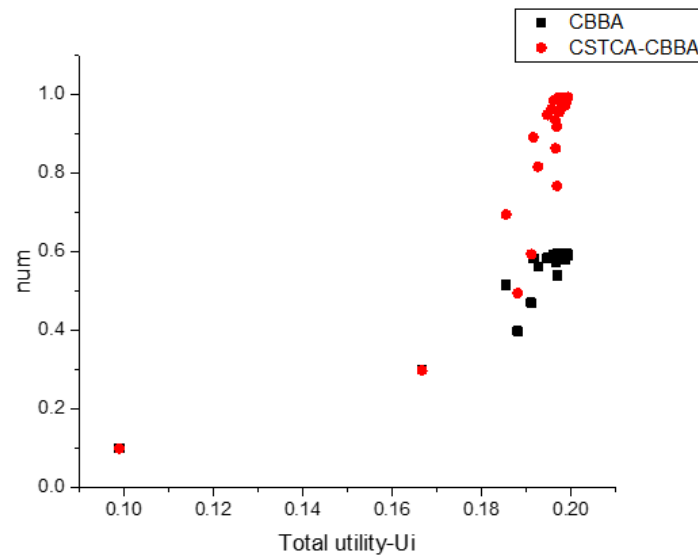


Figure 19. The utility calculation comparison diagram of the algorithm was iterated 100 times.

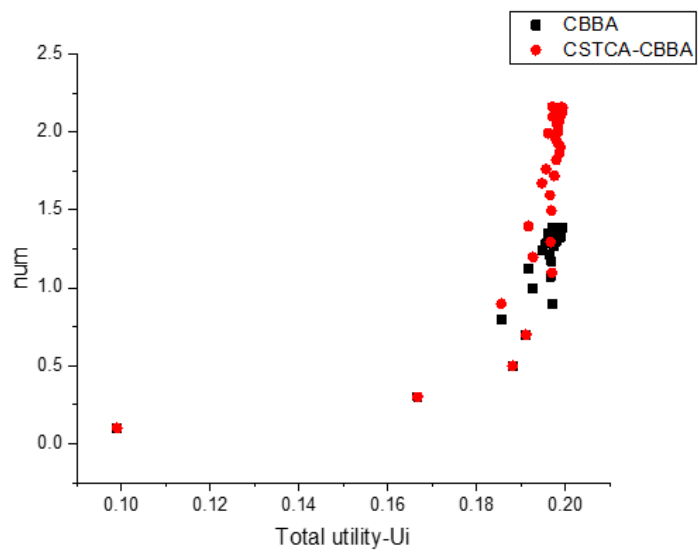


Figure 20. The utility calculation comparison diagram of the algorithm was iterated 500 times.

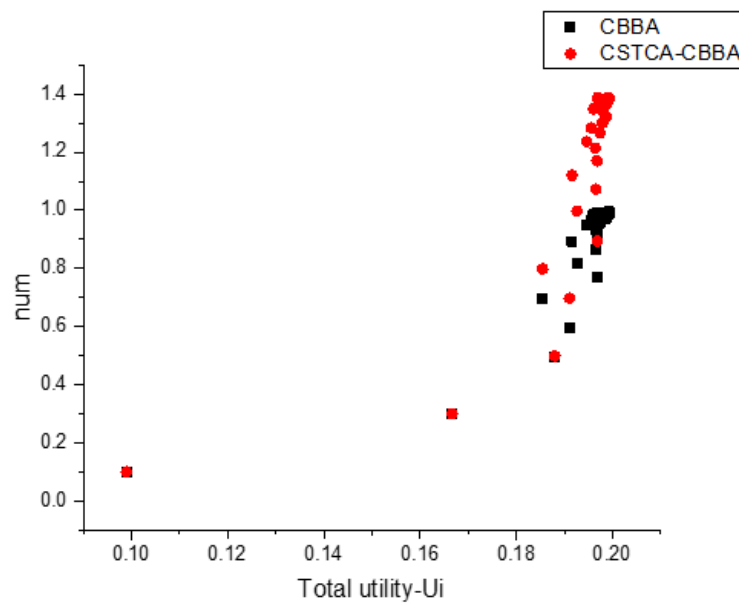


Figure 21. The utility calculation comparison diagram of the algorithm was iterated 1000 times.

6. Conclusions

In this paper, a DSTCA model is constructed based on contract network protocol to solve the mission chain adaptability problem of the UAV complex, and the resource-task-capability allocation method is proposed for the first time. According to the application scenario, TC-CNP is proposed based on CNP, and the bid value, bidding strategy, and bidding strategy based on the capability of TC-CNP are designed. Then using the improved CBBA algorithm, the DSTCA-CBBA algorithm is designed, and a specific case is used to verify the experiment. Through the concrete simulation verification, it is found that the algorithm is more effective in the case of iteration 100 times, 500 times, and 1000 times from the two indexes of time and efficiency, and also verifies the effectiveness of the DSTCA method.

As shown in Figures 13–15, DSTCA is marked in blue, and CBBA original algorithm is marked in orange. When there is no task change at the beginning, the task consumption time of the DSTCA algorithm is relatively small. The time consumption of both algorithms fluctuates with the sudden change of tasks, and then gradually flattens out when the cluster starts executing tasks. When the task changes, the difference in time consumption between the two algorithms is not large. However, you can still see that DSTCA consumes slightly less time. This shows that the algorithm is effective.

In future work, we plan to simulate the adaptive work in the environment of Ubuntu to verify whether the method effectively improves the adaptive performance of the distributed mission link of UAVs, and verify the effectiveness of the DSTCA method by using the simulation means.

Author Contributions: Conceptualization, T.D.; Data curation, T.D.; Funding acquisition, X.Z.; Methodology, Q.L., X.Z. and X.L.; Writing—original draft, T.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 72101263.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

- Zhang, Q.; Liang, X.; Yang, A.; Wu, A.; Wang, N. UAV Swarm Air task Strategy Based on Walf Group Role Matching and Labor Division on Mechanism. *Nucl. Phys.* **2023**, *13*, 104–116.
- Zou, L.; Zhang, M.; Bai, J. Modeling framework for intelligent unmanned swarm operation simulation under OODA-L pattern. *Guofang Keji Daxue Xuebao/J. Natl. Univ. Def. Technol.* **2021**, *43*, 163–170.
- Liu, L.; Li, J.; Xia, B.; Zhao, D.; Tan, Y. Weapons equipment portfolios selection based on equipment system contribution rates. *J. Syst. Eng. Electr.* **2021**, *32*, 584–595.
- Duan, T. A Multi-UAV Complex Task Scheduling Method Based on Adaptive Network. Master's Thesis, National University of Defense Technology, Changsha, China, 2020.
- Vidal, R.; Shakernia, O.; Kim, H.J.; Shim, D.H.; Sastry, S. Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation. *Robot. Autom. IEEE Trans.* **2002**, *18*, 662–669. [[CrossRef](#)]
- Dias, M.B. *Traderbots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments*; Carnegie Mellon University: Washington, DC, USA, 2004.
- Rreiro-Gomez, J.; Mas, I.; Ocampo-Martinez, C.; Sanchez-Peña, R.; Quijano, N. Distributed formation control of multiple unmanned aerial vehicles over time-varying graphs using population games. In Proceedings of the Decision & Control, Las Vegas, NV, USA, 12–14 December 2016; IEEE: New York, NY, USA, 2016.
- Deng, C.; Wang, S.; Wang, J.; Xu, Y.; Chen, Z. LiDAR Depth Cluster Active Detection and Localization for a UAV with Partial Information Loss in GNSS. *Unmanned Syst.* **2024**, 1–13. [[CrossRef](#)]
- Barreiro-Gomez, J.; Mas, I.; Giribet, J.I.; Moreno, P.; Ocampo-Martínez, C.; Sánchez-Peña, R.; Quijano, N. Distributed data-driven UAV formation control via evolutionary games: Experimental results. *J. Frankl. Inst.* **2021**, *358*, 5334–5352. [[CrossRef](#)]
- Jzab, C.; Jxa, B. Cooperative task assignment of multi-UAV system. *Chin. J. Aeronaut.* **2020**, *33*, 2825–2827.
- Chen, C.; Bao, W.; Men, T.; Zhu, X.; Wang, J.; Wang, R. NECTAR-An Agent-Based Dynamic Task Allocation Algorithm in the UAV Swarm. *Complexity* **2020**, *2020*, 6747985. [[CrossRef](#)]
- Yang, W.Z.; Xin, Y. Multi-UAV Task Assignment Based on Quantum Genetic Algorithm. *J. Phys. Conf. Ser.* **2021**, *1824*, 012010. [[CrossRef](#)]
- Braquet, M.; Bakolas, E. Greedy Decentralized Auction-based Task Allocation for Multi-Agent Systems. *IFAC-PapersOnLine* **2021**, *54*, 675–680. [[CrossRef](#)]
- Yu, X. Extended-CBBA-Based Task Allocation Algorithm for On-orbit Assembly Spacecraft. In Proceedings of the 2019 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 17–19 October 2019; IEEE: New York, NY, USA, 2019; pp. 883–888.
- Patil, A.; Bae, J.; Park, M. An Algorithm for Task Allocation and Planning for a Heterogeneous Multi-Robot System to Minimize the Last Task Completion Time. *Sensors* **2022**, *22*, 5637. [[CrossRef](#)] [[PubMed](#)]
- Chen, Y.; Rosolia, U.; Ames, A.D. Decentralized Task and Path Planning for Multi-Robot Systems. *IEEE Robot. Autom. Lett.* **2021**, *6*, 4337–4344. [[CrossRef](#)]
- Mansouri, M.; Pecora, F.; Schüller, P. Combining Task and Motion Planning: Challenges and Guidelines. *Front. Robot. AI* **2021**, *8*, 637888. [[CrossRef](#)] [[PubMed](#)]
- Han, S.; Fan, C.; Li, X.; Luo, X.; Liu, Z. A modified genetic algorithm for task assignment of heterogeneous unmanned aerial vehicle system. *Meas. Control* **2021**, *54*, 1. [[CrossRef](#)]
- Husheng, W.U.; Hao, L.I.; Renbin, X. A blockchain bee colony double inhibition labor division algorithm for spatio-temporal coupling task with application to UAV swarm task allocation. *J. Syst. Eng. Electr.* **2021**, *32*, 1180–1199. [[CrossRef](#)]
- Zhu, Y.; Tang, S.; Yu, W.; Li, Q.; Li, Y. Conditional contract net protocol based distributed guide-weapon assignment algorithm. *Syst. Eng. Electr.* **2011**, *33*, 2673–2677.
- Shen, Z.; Zhou, G.; Huang, H.; Huang, C.; Wang, Y.; Wang, F.-Y. Convex Optimization-Based Trajectory Planning for Quadrotors Landing on Aerial Vehicle Carriers. *IEEE Trans. Intell. Veh.* **2023**, *9*, 138–150. [[CrossRef](#)]
- Li, W.; Ge, Y.; Guan, Z.; Gao, H.; Feng, H. NMPC-based UAV-USV cooperative tracking and landing. *J. Frankl. Inst. Eng. Appl. Math.* **2023**, *360*, 7481–7500. [[CrossRef](#)]
- Li, J.; Zhao, D.; Jiang, J.; Yang, K.; Chen, Y. Capability Oriented Equipment Contribution Analysis in Temporal Combat Networks. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 696–704. [[CrossRef](#)]
- Fedorovich, O.; Lukhanin, M.; Prokhorov, O.; Pronchakov, Y.; Leshchenko, O.; Fedorovich, V. Modeling of logistics of war reserve stockpiling for successful combat operations. *Radioelectron. Comput. Syst.* **2023**, 183–196. [[CrossRef](#)]
- Li, J.; Ge, B.; Yang, K.; Chen, Y.; Tan, Y. Meta-path based heterogeneous combat network link prediction. *Phys. A Stat. Mech. Its Appl.* **2017**, *482*, 507–523. [[CrossRef](#)]
- Bai, X.; Fielbaum, A.; Kronmüller, M.; Knoedler, L.; Alonso-Mora, J. Group-Based Distributed Auction Algorithms for Multi-Robot Task Assignment. *IEEE Trans. Autom. Sci. Eng.* **2022**, *20*, 1292–1303. [[CrossRef](#)]
- Smith Reid, G. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Trans. Comput.* **1980**, *29*, 1104–1113. [[CrossRef](#)]
- Ghanei, F.; Dantu, K.; Modares, J.; Rantanen, M.; Mastronarde, N. Performance of the asynchronous consensus based bundle algorithm in lossy network environments. In Proceedings of the 2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM), Sheffield, UK, 8–11 July 2018.

29. Kim, K.S.; Kim, H.Y.; Choi, H.L. Minimizing communications in decentralized greedy task allocation. *J. Aerosp. Comput. Inf. Commun.* **2019**, *16*, 340–345. [[CrossRef](#)]
30. Samiei, A.; Ismail, S.; Sun, L. Cluster-based hungarian approach to task allocation for unmanned aerial vehicles. In Proceedings of the NAECON 2019—IEEE National Aerospace and Electronics Conference, Dayton, OH, USA, 15–19 July 2019.
31. 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.* **2022**, *78*, 1712–1740. [[CrossRef](#)]
32. Fu, X.; Feng, P.; Li, B.; Gao, X. A Two-Layer Task Assignment Algorithm for UAV Swarm Based on Feature Weight Clustering. *Int. J. Aerosp. Eng.* **2019**, *2019*, 3504248. [[CrossRef](#)]
33. 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](#)]
34. Zhang, J.; Zhu, Y.; Wang, T.; Wang, W.; Wang, R.; Li, X. An Improved Intelligent Auction Mechanism for Emergency Material Delivery. *Mathematics* **2022**, *10*, 2184. [[CrossRef](#)]
35. Yan, S.; Xu, J.; Song, L.; Pan, F. Research on cooperative task assignment scheme of heterogeneous UAVs. In Proceedings of the 2022 5th International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, China, 27–30 May 2022.
36. Geng, N.; Meng, W.; Gong, D.; Chung, P.W.H. How Good are Distributed Allocation Algorithms for Solving Urban Search and Rescue Problems? A Comparative Study with Centralized Algorithms. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 478–485. [[CrossRef](#)]
37. Wu, S.; Liu, X.; Wang, X.; Zhou, X.; Sun, M. Multi-robot Dynamic Task Allocation Based on Improved Auction Algorithm. In Proceedings of the 2021 6th International Conference on Automation, Control and Robotics Engineering, CACRE 2021, Dalian, China, 15–17 July 2021; pp. 57–61.
38. Wang, J.; Jia, G.; Xin, H.; Hon, Z. Research on dynamic task allocation method of heterogeneous multi-UAV based on consensus based bundle algorithm. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 2214–2219.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.