IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 25, 2019, accepted July 6, 2019, date of publication July 15, 2019, date of current version August 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2928524

Function Chain-Based Mission Planning Method for Hybrid Combat SoS

HUA HE, WEIPING WANG, YIFAN ZHU, XIAOBO LI, AND TAO WANG
Institute of Simulation Engineering, National University of Defense Technology, Changsha 410073, China
Corresponding author: Hua He (hehua3566@qq.com)

This work was supported in part by the Natural Science Foundation of China, under Grant 61273198.

ABSTRACT With the development of unmanned equipment technology, new types of intelligent weapons such as loitering missile and unmanned aerial vehicle(UAV) are widely used. The combat system of systems (SoS) presents the characteristics of multifunctional combat equipment and hybrid manned and unmanned force units. At present, the research of mission planning is mostly oriented to a single factor, does not consider a variety of operational activities. Therefore, how to describe the hybrid operational organization of multiple functions and force units and to accomplish mission planning for various operational activities has very practical significance. In this paper, the FINC(Force, Intelligence, Networking, C2) model is extended, a new model framework of unmanned combat force with the function of integrated observe and attack is proposed, which is called Force Intelligent Network Command and Control Model with autonomous platform (FINCA). Four types and eight different function chains are proposed for the hybrid Combat SoS. Through the capability requirements and resource constraints, the operational tasks are mapped to the types and quantities of function chains that needed to accomplish this task, and the multi-priority list dynamic reconstruction scheme (MPLDCS) method is used to generate the function chain planning scheme. The many-objective optimization algorithm NSGA-III is improved, let mission execution time, connection change induced consumption, and the real-time remaining number of UAV composed the objective function to find all possible Pareto front. Finally, the method proposed in this paper was validated by the case of the modular and reconfigurable fire brigade proposed by the US.Army to participate in the border operations.

INDEX TERMS Hybrid combat SoS, mission planning, function chain, FINCA, MPLDCS, improved NSGA-III.

I. INTRODUCTION

New intelligent weapons such as loitering missile and UAV with the function of observation and attack are characterized by miniaturization, multi-platform, low cost and high maneuverability [1]. To make it become more flexible and more widely used in a variety of soft and hard killing missions. With the wide application of new intelligent weapons such as loitering missiles and UAVs, related mission planning has become a research hotspot. With the addition of new intelligent weapons, the networked features of hybrid combat SoS with multiple types of combat nodes have some new characteristics, how to describe complex combat SoS and based on this perform mission planning has very practical significance.

The associate editor coordinating the review of this manuscript and approving it for publication was Baozhen Yao.

Most of the existing research on mission planning is aimed at simple single function, such as cooperative search [2], cooperative attack, etc., without considering complex mission planning for multiple combat activities. According to the OODA loop theory, operational activities can be divided into four processes: Observe, Orient, Decide, and Act [3]. In the F2T2EA theory of the kill chain, the entire operational process is divided into six phases: Find, Fix, Track, Target, Engage and Assess [43]. Therefore, the existing methods cannot meet the actual needs of the operations. however, the new intelligent weapon have multiple functions, and different equipment can be combined to form different functions. Naturally, this paper proposes a function chain-based method to mission planning. The combat SoS is divided into different types of nodes, and the combination of different nodes is determined according to the function, and different function chains are used as basic units and combined to perform tasks.

Another advantage of using the function chain as the basic unit is that the equipment in the networked combat organization does not need to frequently switch the node connection structure during the participation in the combat activities, such as UAV participating in different activities belongs to different command and control nodes. Because switching the connection will cause time consumption, and second, it will be affected by communication interference in the battlefield, etc. So it is not suitable to change the connection structure frequently. After the basic task order is determined, the paper uses the MPLDCS method to schedule and generate a feasible execution scheme. For the ever-changing battlefield, the time of battle, the consumption of connection changes, and the residual usage of UAVs are all crucial, these factors are closely related to the order of task execution, so it is necessary to select the mission plan.This is a typical multi-objective optimization problem.

The evolutionary multi-objective algorithm NSGA-III, which is based on the reference point-based non-dominated sorting is selected as the optimization method. Compared with other algorithms, it has the advantages of fast convergence and maintaining the diversity of population [4]. According to the characteristics of the hybrid combat SoS mission planning problem, the algorithm is improved. In order to improve the timeliness of mission planning, adaptive genetic operators are used to generate the offspring population, and the NSGA-III algorithm is improved in the process of crossover and mutation, by associating with the non-dominated sorting hierarchy of chromosomes, such that good chromosomes can be retained with high probability.

II. BACKGROUND AND RELATED WORK

In order to better understand the complex military organization modeling and mission planning, it is necessary to understand the related concepts and research status from the aspects of combat SoS network modeling, mission planning methods and practices, and optimization methods for solving problems.

A. NETWORK DESCRIPTION OF THE COMBAT

The combat SoS can be seen as a "super system" composed of functional entities, which are complex in nature and interact to achieve the common goals [5]. Some important visions about complex military organizations have been proposed, including the emergence behavior among other component systems. Alberts *et al.* have pointed out that integrated systems need to generate and utilize information advantages, effectively integrate C2 (command and control), weapon systems and forces; this can improve information dissemination, intelligence sharing and coordination capabilities [6]. Therefore, the combat SoS is not random assembly of weapon systems, but a comprehensive application of multiple combat systems under the information network and multiple relationships. Therefore, the combat SoS itself is a complex network, because the network can reflect the important characteristics of the combat SoS, that is, how the combat system self-organizes on the basis of the network. This network is not only a simple network similar to a communication network, but also a hybrid network which includes command, control, environment, society and other elements [7].

Complex networks have also been used in military. For example, Cares uses complex networks to define and deduce the IACM model of the information age [8]. Carley *et al.* proposed the PCANC model of military organizational structure [9], [10], however, the nodes and edges in the proposed method are uniform and should be distinguished under specific research problems and background. Dekker simulates the relationship between different network topologies and operational effectiveness, and applies the traditional social network analysis (SNA) to the military, and proposed the FINC (Force, Intelligence, Networking and C2) model to support the description of heterogeneous nodes, and to analyze the C4ISR system of air and land warfare [11].

B. MISSION PLANNING METHOD AND PRACTICE

At present, the research on mission planning includes two aspects. One is to establish a mathematical model of mission planning and use algorithm to solve it, the other is to carry out related experiments and applications.

1) MISSION PLANNING MODEL SOLUTION

Many scholars use modeling methods to study mission planning problems. The current research includes multi-UAV cooperative task assignment, loitering missile cooperative task assignment, weapon target allocation, etc. The main modeling methods include multiple traveling salesman problem(MTSP) model, vehicle routing problem (VRP) model [12], mixed integer linear programming (MILP) model [13]. Solutions includes network optimal model based method [14], tabu search based method [15], graph theory model based method [12], A* algorithm [16], market mechanism based method [17], [18] and evolutionary algorithms such as genetic algorithms (GA) [19], ant colony algorithm (ACA) [20], etc. These methods have specific advantages, but they also have limitations. For example, A* algorithm, as a deterministic algorithm, has the shortcomings of time and space complexity in solving large-scale combinatorial optimization problems. With the increase of the scale of the problem, the difficulty of solving it increases sharply. Because of its randomness in nature, genetic algorithm has many inferior search processes in the process of solving, which leads to low efficiency and accuracy in solving large-scale combinatorial optimization problems. The optimization ability based on market mechanism method is according to different object negotiation and competition. When the task allocation problem is large, the negotiation traffic between objects will greatly increase and the efficiency of problem solving will be reduced.

2) MISSION PLANNING RELATED EXPERIMENTS

From 2000 to 2004, the US Air Force Laboratory Flight Control Department (AFRL/VACA) and the US Air Force

Institute of Technology used Network Flow Optimization carried out the multi-machine collaborative multi-task allocation research in the context of the Wide Area Search Munitions (WASM) project. In 2008, the dynamic mission planning, collaborative control algorithm and search mode planning and other research were carried out based on the Cooperative Operations in Urban TER-rain (COUNTER) project, which included basic search algorithm research, ground station development and flight test. Based on the continuous monitoring task, Stanford University conducted a single/multi-UAV search strategy research for continuous surveillance tasks [21], [22]. Considering the mission performance, the UAV itself is designed, and a super system design framework based on collaborative optimization is proposed, tests and validate at Boeing's Vehicle Swarm Technology Laboratory (VSTL) [23]. The US Army Aviation Application Technology Department (AATD) launched the autonomous cooperative operations of UAVs, and Rockwell Science Corporation (RSC) led the establishment of a joint research project team composed of research institutions, government departments and industry to develop and validate the multi-UAV collaborative operational capability, the hardware platforms that participate in the task are usually of different types (such as vertical take-off and landing UAVs and fixed-wing UAVs), and their power system and sensor processing capabilities are also different, software platforms (such as mission planning, path selection, sensors data processing, perception capabilities, etc.) and their functions may change with the change of mission requirements [24].

C. MULTI-OBJECTIVE OPTIMIZATION METHOD

The mission planning problem has been transformed into multi-objective optimization problem (MOP) research for many years [25], and many multi-objective optimization algorithms have been derived. These algorithms originally aimed at two goals, such as NSGA-II [26] and SPEA2 [27], which are based on Pareto domination, but for three or more goals, these methods are not very effective, because the proportion of non-dominated solutions in the population will be very high. Therefore, many improved multi-objective optimization methods have been derived to improve the search ability of the algorithm to meet many objective problems. For example, the Indicator Based Evolutionary Algorithm (IBEA) [28], through a variety of performance indicators to guide the search process of the population; Grid-based Evolutionary Algorithm(GrEA) [29], which introduces a new dominance relationship and the natural partitioning advantages of the grid to enhance the selection pressure of the optimal direction; based on the preference-inspired co-evolutionary algorithm (PICEAs) [30], [31], avoids the random search for high-dimensional and multi-objective optimization problems to guide the scheme with optimization to approach the Pareto front. But these methods also have some shortcomings. For example, in the performance-based improvement method, when the number of targets is large, the cost of performance index calculation becomes too large; the new dominant relationship in the dominant relationship of grid-based evolutionary has a greater relevance with the problem itself; in the method based on preference information, the random generated preference information makes the optimization efficiency not high.

Based on the improved goal decomposition strategy, NSGA-III maintains the diversity of candidate solutions by generating a set of predefined and regularly distributed reference points. Finally, the enhancement algorithm converges, the algorithm has been proved better than most multi-objective optimization methods in many multi-objective optimization benchmark problems. The algorithm framework of NSGA-III and NSGA-II is roughly the same, except that the selection mechanism is different. NSGA-II uses the crowding distance to select individuals of the same non-dominated level, while NSGA-III uses a reference point-based approach to select individuals. The NSGA-III adopts the reference point-based method to solve the problem of poor convergence and diversity of the algorithm when the crowding distance continues to be used in the multi-objective optimization problem with three or more targets. Therefore, NSGA-III improves the convergence effect of the algorithm while ensuring the diversity of the population [4], [32].

It can be seen from the above description that FINC method can describe the heterogeneous nodes of combat SoS, but for the new intelligent combat weapons, the node type of FINC cannot be described. Therefore, we propose to extend FINC to FINCA, plus autonomous node(A), to meet the actual needs of the combat SoS. Most of the existing researches plan simple single-function tasks. The new intelligent weapons have multiple operational capabilities. If the mission planning only considers single tasks and single capabilities, it is easy to waste resources. In the 1970s, Colonel Boyd based on his experience in air combat decomposed a combat action into Observe, Orient, Decide and Act four processes, that is OODA loop theory [3]. The OODA loop is the basic loop for operational capability generation. The former US.Air Force chief of staff, General Ronald Fogleman, proposed the concept of kill chain, which divides the series of cyclic processing of attack targets from detection to destruction into six stages: Find Phase, Fix Phase, Track Phase, Target Phase, Engage Phase, and Assess Phase(abbreviated as F2T2EA) [43]. In the study of network structure, R. Milo *et al.* proposed the concept of motif according to the repeated occurrence of some important connection structures in the network [33]. Guofeng *et al.* validated the efficiency of the motif [34]. Militaryly, Y. Lee and T. Lee proposed two basic structures of independent and joint attacks as motifs for heterogeneous networks, measuring attack opportunities to measure operational efficiency [35]. Combining the concepts of OODA, kill chain and motif, a function chain-based mission planning method is proposed.

From the above description of the mission planning method, we can see that many researchers have combined with the biomimetic method to find the optimal solution, such as ant colony algorithm, genetic algorithm and so on. However, with the increasing requirements of information warfare, the commonly used task assignment solving methods can no longer meet the needs of information warfare. Therefore, this paper selects the NSGA-III algorithm based on timeliness and population diversity, and customizes and improves it according to the characteristics of the problem. Finally, the effectiveness and advancement of improvement of the algorithm are verified by the combat case.

III. BUILDING A COMBAT SoS FUNCTION CHAIN BASED ON THE FINCA MODEL

The formation of operational network is a gradual evolution process like other networks, and there are many uncertainties. However, a certain operational network is likely to have some typical network characteristics to some extent. For example, the satellite communications network in the Army Land Battle Network has the characteristics of a star network, while the tactical warfighter information network may have a scale-free characteristics [36]. Drawing on the latest theoretical achievements of complex network research, abstracting the combat SoS into a two-dimensional network topology will become an effective method for the analysis and research of war complex SoS. The ever-changing operation style requires that the overall structure of command and control and its supporting systems and algorithms are constantly updated and evolved to accommodate the demands of modularity, intelligence, and networking.

A. FINCA MODEL

The FINC [11] method is a military organization analysis method based on traditional social network analysis(SNA) method. Social networks can be described by graph models. The nodes in the graph are called actors, the edges represent the relationships and interactions between actors. SNA is mainly used to describe and measure the relationship between actors. The FINC method extends SNA to the military field. While retaining its basic analysis methods, it distinguishes the attributes and roles between actors according to the characteristics of the military system, and provides an effective method for military organization analysis.

FINC is short for Force, Intelligence, Networking and C2(Command and Control). The basic idea of FINC is to establish a network model consisting of network nodes such as force, intelligence and command, and the edges that describe the communication association between nodes. Based on the model, we can analyze the cooperation ability of units, network centrality and observe performance of organizations. However, for the new type of combat forces, loitering missile and UAV with the function of observation and attack, etc., are both force node and intelligence node, and also have certain autonomous decision-making capabilities. For this kind of node, the existing FINC model lack description capabilities, so, naturally, we propose to extend the node type and define this kind of hybrid node as autonomous node (*A*), because such nodes have their own process of observe to autonomous decision-making, this process is described as a self-cycling edge.

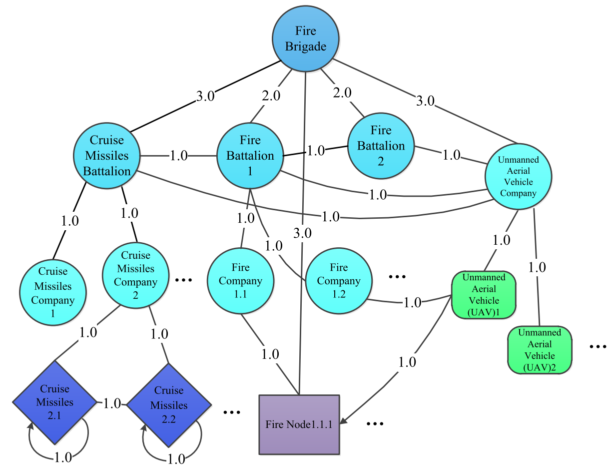

FIGURE 1. Example of FINCA Model.

Fig.1 is a simple example of a hybrid brigade-level combat SoS. For the sake of simplicity, only a few organizational nodes are drawn.in the figure, the rectangular nodes represent the force units (F), the rounded rectangle nodes represent the intelligence units(I), the edge represents the information transmission channel(N), the circular nodes represent the C2 units(C), and the rhombus nodes represent the autonomous units(A), where, directed edges indicate that information can only be transmitted in one direction, undirected edges indicate that information can be transmitted in both directions, the weights on each edge represent the time delay transmission on the information channel, the information delay for the self-circulation on the A node describes the time when the node observe information to makes its own decision.

B. OPERATIONAL CAPABILITY REQUIREMENT DECOMPOSITION

According to the operational characteristics of the hybrid combat SoS and the operational capability requirements of the modular reconfigurable long-range fire brigade proposed by the US army, this paper devides the capability requirements of the hybrid combat SoS into eight capability values: observe capability, mid-short range fixed target attack capability, mid-short range mobile target attack capability, long-range fixed target attack capability, long-range mobile target attack capability, inspection attack capability, temporary decision making capability, and assessment capability.

According to the geographical environment in which the task performed, the number of targets and the scope of the operation, these specific parameters can be quantified to solve the eight capability requirement values required to complete the task, namely the capability requirement vector. Suppose that for a task T_i, the capability demand vector V_i of T_i, can be defined as:

$$v_i = (v_{i1}, v_{i2}, v_{i3}, v_{i4}, v_{i5}, v_{i6}, v_{i7}, v_{i8}) \tag{1}$$

For convenience, the above eight abilities record as the first to eighth abilities, and naturally, the *j-th* abilities of the *i-th* function chain are recorded as β_{ij}. From the perspective of the capability requirements to accomplish combat mission, the function chain combination formed should have the ability to complete the task, that is, to satisfy the capability requirement vector of the formula(1). Assume that the number of eight function chains required to complete a combat mission is $q_k (k = 1, 2, \ldots, 8)$, the following inequalities need to be satisfied:

$$(q_1, q_2, \ldots, q_8) \cdot \begin{pmatrix} \beta_{11} & . & \beta_{18} \\ . & . & . \\ \beta_{81} & . & \beta_{88} \end{pmatrix} \geqslant v_i \tag{2}$$

Refer to the common formula for calculating the cooperate effect of multiple fires in joint fire strike: $p = 1 - \prod_i (1 - p_i)$, p_i is the effect of single factor, and p is the result of multi-factor synthesis [38]. so there is:

$$p_{ij} = 1 - \prod_{s \in (1,8)} (1 - \beta_{sj})^{q_{is}} \tag{3}$$

It needs to meet $p_{ij} \geq v_{ij}$, p_{ij} is the *j-th* capability value of the multi-function chain to work together on the of task i, v_{ij} is the *j-th* capability requirements value of task i.

C. FUNCTION CHAIN NUMBER CONSTRAINTS

Considering the non-recyclability of weaponry and the number of F-nodes and A-nodes, the number of F-nodes and A-nodes in the mission-required function chain during mission planning need meets the following constraints:

$$\sum_{i=1}^{8} \sum_{k \in K} q_{ij} \leq N_F, \quad K \in \{2, 4, 6, 7\} \tag{4}$$

$$\sum_{i=1}^{8} \sum_{k \in K} q_{ij} \leq N_A, \quad K \in \{3, 5, 6, 8\} \tag{5}$$

where q_{ij} is the number of *k-th* function chain usage of task I, N_F and N_A are the number of F and A nodes respectively.

D. FUNCTION CHAIN CONSTRUCTION BASED ON COMBAT LOOP

Colonel Boyd decomposed a combat operation into Observe, Orient, Decide, and Act processes, forming a combat loop, or OODA loop [3]. The theory of OODA has been successfully applied by Americans in the air combat plan of the Gulf War, the development of sophisticated weapons in the US (such as the F15 and F16 fighters designed by Boyd) and informationizion command system C4ISR. Which provides new means and ways for the scientific and rationalization of military strategic decision-making. As shown in Fig.2. Among them, Observe and Act are mainly based on technical means, Orient and Decide are also includes the psychological process of decision makers, intelligence analysts and domain experts.

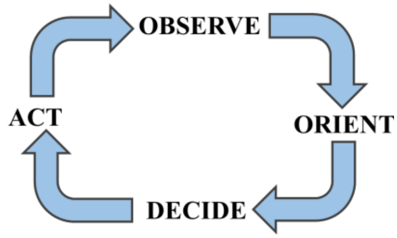

FIGURE 2. **OODA Loop.**

The hybrid combat SoS combines the basic elements of network forces into an organic whole, thus forming the core structure of operational capability. The OODA loop is the basic loop for the generation of operational capability, the concept of the combat loop is generalized and has different meanings for specific tasks. For a observe task, the combat loop process can be understood as determining the observe node and observe based on the judgment of the existing information. According to this idea, the basic function chain requirements are determined according to the decomposed mission, and the function chain is composed of the basic nodes shown in Fig.1. The modern battlefield is a networked battlefield. The number of nodes and the connection method of the combat system can be designed according to the overall requirements of the task chain. Based on this method, the interaction between the function chains can be reduced, and the connections between the functional nodes can respond to the task more quickly.

According to the characteristics of hybrid operations, different tasks are mapped to several function chains with simple structure, frequent used and have practical significance, and corresponding function chain models are established. In order to simplify the calculation, the intelligence node is defined as a UAV with observe function only.

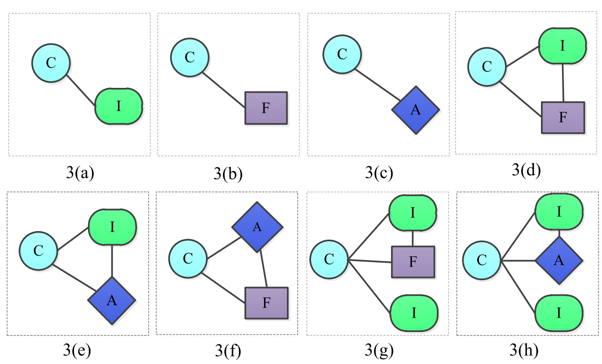

FIGURE 3. **Basic functional chain model.**

(1) Observe chain: As shown in Fig.3(a), the structure is connected by a UAV node and a C2 node. The C2 node (C) sends observe orders to the UAV(I) according to the requirements of the combat mission, and the UAV observe the target area and sends the observe information to the C2 node. For convenience, the function chain is denoted as L1.

(2) Attack chain: as shown in Fig.3(b) and 3(c), 3(b) is a structure connected by the C2 node(C) and force node(F), and as shown in Fig.3(c) a structure in which a C2 node(C) and a autonomous node(A) are connected. According to the combat mission and the observe information, the C2 node sends attack orders to the force node, and the force node attack the specified target and the area. This function chain is denoted as L2(I) and L2(II), respectively.

(3) Observe and attack integrated chain: as shown in Fig.3(d), 3(e), and 3(f), 3(d) is a structure connected by C2 node(C), force node(F) and intelligence node(I).The C2 node sends observe orders to the intelligence node for the uncertain target according to the combat mission and the observe information, and then the intelligence node sends the observe information to the force node directly, and the force node attack the specified target and the area. The chain is also denoted as L3(I). 3(e) is a structure connected by C2 node(C), autonomous node (A) and intelligence node (I), the C2 node sends observe orders to the intelligence node for moving or obscuring target according to the combat mission and the observe information, and then the intelligence node sends the observe information to the autonomous node directly, and the autonomous node performs observe and attack on the specified target and area. This function chain is denoted as L3(II). 3(f) is a structure connected by C2 node (C), autonomous node (A) and force node (F), the C2 node sends a command to the autonomous node for the high-value moving target according to the combat mission and the observe information, and then The autonomous node direct sends the observe information to the force node, the autonomous node guidance and cooperates with force node performs attack on the specified target and the area. This function chain is denoted as L3(III).

(4) Observe control attack and evaluation integrated chain: as shown in Fig.3(g) and 3(h), the structure connected by intelligence node (I), the C2 node (C), the node with the attack function (A or F) and the intelligence node (I). For the regional target, the C2 node dispatches the UAV to performs observe, and after receiving the observe information, command the force node to attack, and after the strike, dispatches the UAV again to performs observe and evaluates the combat effect to determine whether a second attack will be carried out. This function chain is denoted as L4(I) and L4(II), respectively.

IV. PLANNING MISSION GENERATION CONSIDERING FUNCTION CHAIN RECONFIGURATION

A. TASK RELATION DESCRIPTION

According to the Hierarchical task network (HTN) [39], [40] theory, the overall mission of the war can be gradually decomposed into a series of subtasks. Suppose the mission is decomposed into a task set $T = \{T_1, T_2, T_3, \ldots T_k\}$ (k is the number of tasks in the task set). In the process of continuous refinement of task granularity, there are more and more interdependent relationships between tasks. If all tasks with dependencies are connected by directed edges, they form a directed graph between tasks. That is, the task graph [41].

In this article, we use task graphs to represent the relationship between them: The task graph is defined as a binary array (V, E), where V is a non-empty vertex set. E is a set of point pairs consisting of vertices in V, called edge sets. $E(i,j)$ represents the dependencies between tasks [42]. As shown in Fig. 4, there are 12 tasks in the figure, and the directed edge represents the sequence relationship between the parent task and the child task, that is, the child task cannot be executed before the parent task. We consider the order relationship between tasks as a special priority relationship with strict time order.

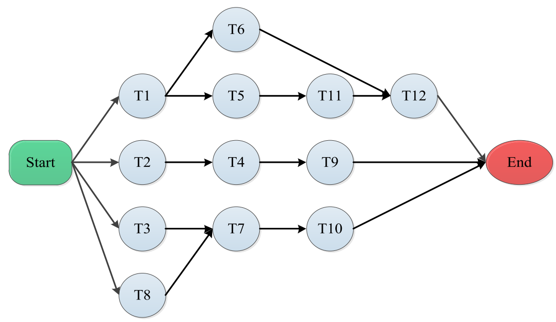

FIGURE 4. Example of task graph.

B. FUNCTION CHAIN DYNAMIC RECONFIGURATION

When a task is completed, UAV resources are released and checked whether the existing usable UAV can meet the resource requirements of the next task. If satisfied, the next task start. If not, the next task will keep wait until there is a task completed and the function chain resources are released. In this process, all the function chains that need to be used in the task need to be considered. In the cooperative operation process, each node keeps connected with the superior node, and establishes connections with other nodes according to needs, such as the UAVs cooperation begins command by UAV command vehicle. And then changes to direct communication with the firepower battalion, as shown in Fig.5(a); the autonomous node and other force nodes temporarily formed cooperation, as shown in Fig.5(b). These process will result in disconnecting and reconnecting between nodes, that is, the communication and time consumption, this is very important relative to the rapidly changing battlefield.

For F-node and A-node, one C-node will contain several F-nodes or A-nodes. In the process of function chain scheduling, when there is a task completion, If the resource is not satisfied and no new task starts, the I node is disconnected from the temporary node and autonomos connected to the superior node, and the connection consumption vector is (0,0,0,2,2,0,4,4). And if there are tasks start after the end of the task, in order to reduce the connection consumption, and the time and communication consumption caused by frequently switching C nodes, try to use the existing connection

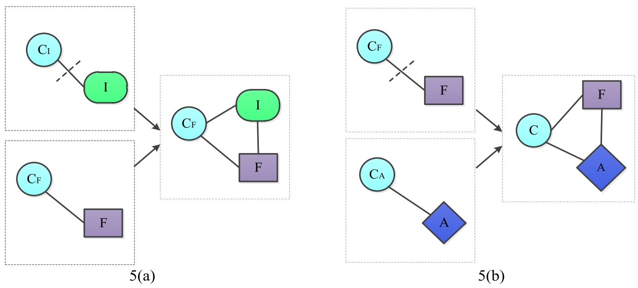

FIGURE 5. Function chain reconfiguration in the process of cooperative operations.

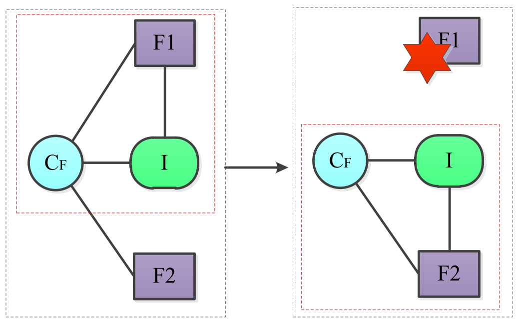

FIGURE 6. Link reuse.

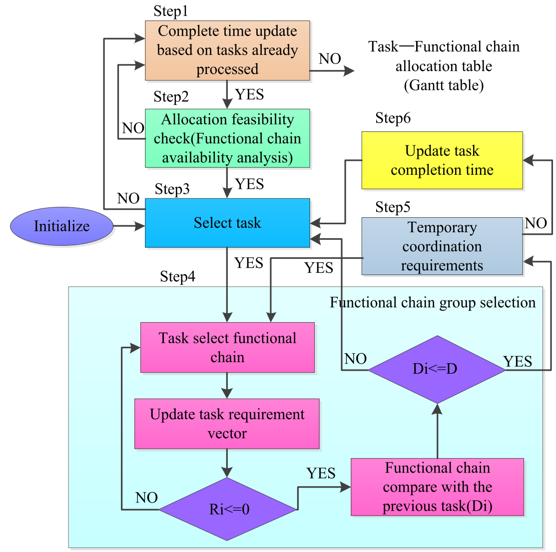

FIGURE 7. MPLDCS algorithm flow.

as far as possible. For use new chain L1, L2(I), L2(II), there is no link reassembly, and link L3(I) is as shown in Fig.6. Using the C_F-I that has been constructed before, the connection consumption becomes 1, the link L3(II), L4(I) and L4(II) are similar. Link L3 (III) needs to be reconfigured every time it is used, and the connection consumption is 3 every time. That is, the vector of connection consumption is (0,0,0,1,1,3,1,1).

Therefore, after the completed tasks and the started tasks are matched, the function chain of the subsequent tasks cannot be satisfied in the pre-order task, needs to be reorganized, and the vector of link changes is (0,0,0,3,3,3,5,5); the connection consumption vector of the functional chains that are not needed in the following tasks is (0,0,0,2,2,0,4,4); the connection consumption vector of the functional chains that can be used in the following tasks is (0,0,0,1,1,3,1,1).

C. FUNCTION CHAIN SCHEDULING METHOD

After the mission decomposition using the method in the ''Task Relation Description'' section, we get the type and number of function chains required for these tasks. In this part, we study how to organize existing combat nodes to form a function chain for a mission planning scheme to achieve better combat effectiveness.

Because the matching of function chains and task requirements requires the measurement of multi-dimensional variables, and the task requires the collaborative processing of different function chains, the search and solution for this problem is a complex scheduling search problem. The multi-dimensional dynamic list scheduling method (MDLS) is often used to solve such problems [23], [15]. The MDLS algorithm searches the resource-task scheduling schemes in two key steps: (1) selecting processable tasks from the optional task set; and (2) selecting the best combination of function chains to perform tasks that need to be processed. Yang [45] pointed out that although MDLS algorithm solves the problem of task planning, from the search process of its solution, the MDLS algorithm does not find the best allocation scheme for solving the planning problem. In some specific cases, the results are even flawed. It mainly includes the search strategy easily leads to limited results and the priority function setting is unreasonable, and proposes a multi-priority list dynamic scheme (MPLDS) algorithm. In this paper, aiming at the allocation operational nodes to functional chains, based on the idea of MPLDS algorithm and its applicability improvement, a multi-priority list dynamic reconfiguration scheme (MPLDCS) algorithm is proposed, which defines the task selection function chains, the function chain selection task and the priority function of the task to be processed in the optional task set.

To overcome the defect of MDLS algorithm, the distance between the function chain capability vector and the task requirement vector is defined, and multiple priority lists of resource provisioning and time saving of each function chain are established to determine the function chain of the task selection. Then using the connection consumption formula calculate the demand difference of the function chain between the selected task and the previous task, and compared with the pre-specified threshold value. Through the dynamic reconstruction process in the previous section ''Function Chain Dynamic Reconfiguration'', Choosing tasks that are as close as possible to the previous task's functional chain requirements can reduce connection consumption.

The algorithm flow chart is as follows:

The main work of the algorithm includes the following four parts:

(1) Analysis of the ready-to-use function chain based on node status;

(2) Select the task to be processed from the task graph GT according to priority;

(3) Select the best combination of function chains to processing task, and compare it with the function chain group of the previous task;

(4) During task execution, checks whether there is a temporary coordination requirement.

Variable definitions of the algorithm process:

Ready is a set of tasks that can be processed at the current time;

Vector $R = [R_1, R_2, R_3, R_4]$ is the set of remaining resources of the initial combat SoS. R_1, R_2, R_3, R_4 are the number of available F, I, C, and A nodes, and N is the connection rule of each node.

$$Q = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 0 \\ 0 & 2 & 1 & 1 \end{bmatrix}$$

is the function chain connection matrix, q is the corresponding function chain requirement vector, $q \cdot Q$ which satisfies the constraint of the network connection N, ie $q \cdot Q \in N$.

FREE is a function chain set that can be acquired by processing tasks at the current time, and *Ps* is a function chain set of the combat SoS, $FREE \subset Ps$;

OUT(i) is the direct follow-up task set of task i,;

nOut(i) is the number of direct follow-up tasks for task i, $nOut(i) = |\ OUT(i)\ |$;

IN(i) is the direct previous task set of task i,;

nIn(i) is the number of direct previous tasks for task i, $In(i) = |\ IN(i)\ |$;

M is the set of tasks that have been processed currently;

L is the type of capability of the function chains;

β_{ij} is the *j-th* ability value of the function chain p_i;

$l(m)$ represents the function chain p_m last processed task (if the function chain has not processed any tasks then $l(m) = 0$);

$q(i)$ represents a function chain group for assigning processing task i;

$FT = < f_1, f_2, \ldots, f_m >$, indicating the completion time of each task to be processed currently;

$F_G(f)$ represents the task group whose deadline is f, and $G(F_G)$ is all function chain for processing the task group;

$D(t_i)$ represents the time to process task i;

S_i is the start processing time of task i;

Di is the function chain requirement difference between the current task and the previous task;

D is the set threshold of requirement difference;

tp_{im} is the priority of task t_i selects function chain p_m;

$n = 0$ is the number of connection changes initialized to 0;

The detailed process of the algorithm is as follows:

Algorithm 1

Initialize $Ready = \{i|nIn(i) = 0\}$, $FREE = Ps$, $|M| = 0$.
Step 1: Update the completion time of the tasks in M (skip this step in the initial stage)
    Set $f = \min_{f_t \in FT}(f_t)$;
    $FT \leftarrow FT \backslash \{f\}$;
    $FREE \leftarrow FREE \cup G(F_G(f))$;
    for each $i \in F_G$;
     for each $j \in OUT(i)$;
        $nIN(j) \leftarrow nIN(j) - 1$;
       if $nIN(j) = 0$;
          $Ready \leftarrow Ready \cup \{j\}$;
       end if;
     end for;
    end for;
Step 2: Analysis satisfaction of resource
    if $\forall i \in Ready \exists l : \sum_{m \in Free} \beta_{ml} \leq [q \cdot Q]_{il}$, $l = 1, 2, \cdots, L$;
     then *GO TO Step 1;*
      else *GO TO Step 3;*
     end if;
Step 3: Task selection
     if $READY = \phi$;
     GO TO Step 1;
     end if;
     Set
   $Ready' = \{i \in Ready| \sum_{m \in Free} \beta_{ml} \geq [q \cdot Q]_{il}$,
   $l = 1, 2, \cdots, L\}$;
     Select $i = \arg\min_{j \in Ready'}\{pr(j)\}$;
     $Ready \leftarrow Ready \backslash \{i\}$;
Step 4: Function chain group selection
        $q(i) = \phi$;
      do until $R_i = 0$;
        $n = \arg\max_{m \in Free}\{tp_{im}\}$;
         $Free \leftarrow Free \backslash \{m\}$;
     for $l = 1$ *to* L;
        if $[q \cdot Q]_{il} \geq \beta_{nl}$;
          then $[q \cdot Q]_{il} = [q \cdot Q]_{il} - \beta_{nl}$;
          else $[q \cdot Q]_{il} = 0$;
          end if;
       End for;
          $q(i) \leftarrow q(i) \cup \{n\}$;
       end do;
Step 5: Function chain requirement difference with the previous task
        if $Di <= D$
         GO TO Step 6;
        else *GO TO Step 4;*
Step 6: Analysis of the requirement for cooperation
        If $i = null$;
          GO TO Step 7;
        else *GO TO Step 4;*

Algorithm 1 *(Continued.)*

Step 7: Task time update

$$s_i = \max(f, \max_{m \in Gp(i)} \{s_{l(m)} + D(t_{l(m)})\});$$

$$f = s_i + D(t_i);$$

if $f \notin FT$;

$FT \leftarrow FT \cup \{f\}$;

end if;

Step 8: Update connection consumption

$n = n + n(before, now)$

Output n.

After all tasks are completed $t = max(FT)$;

Output t.

V. OPTIMIZATION OF MISSION PLANNING SCHEME BASED ON IMPROVED NSGA-III

A. OPTIMIZATION VARIABLES AND OBJECTIVE FUNCTION

If the execution order of the task is determined, it is used as the input of the MPLDCS method to calculate the value of the objective function. Therefore, the priority of tasks execution is the key of the schedule strategy. Taking the task execution order as an optimized variable needs to meet the constraints of task graph.

Let the completion time of the mission, the time and communication consumption of the connection changes, and the average usage of UAV composed the objective function.

1) MISSION COMPLETION TIME

Mission completion time $t = max(FT)$

The time spent on tasks can be divided into the time of the task execution and the time of the preparation phase. The time of the preparation phase includes the waiting time due to the shortage of resources (unmanned aerial vehicles), and the time of constructing function chains and multi-chain coordination participation time.

The time to build function chains is the maximum time built in the task that is about to start. The time for constructing the function chain L1 to L4(II) is divided into the previous group of tasks that can be satisfied and cannot be satisfied. If it is satisfied, the construction vector time is (0,0,0,2,2,3,2,2), else is (0,0,0,3,3,3,5,5).

In the network described by FINCA, the time of the combat loop includes two aspects, one is processing information at each node (*delay of nodes*), and the other is transmission delay on the information channel (*delay of channels*). The time of chains coordination includes the time consumption of each node and each edge to process and transmit information, that is, when multiple chains participate, it is necessary to determine the operational chain that need to be used. Therefore, it is necessary to first determine the cooperation time between each two chains, that is, to give a time coordination matrix. When multiple chains participate, only need to select the maximum cooperation time in the cooperation matrix.

2) CONNECTION CONSUMPTION

Because of each node keeps its connection with the superior node when it is not performing tasks, the function chains L1, L2 (I) and L2 (II) are simple connection structures between I, F and A nodes with their superior nodes, so there is no connection consumption when the number changes. The first task connection change amount is calculated as $n_c = l_i \cdot (0, 0, 0, 3, 3, 3, 5, 5)^T$, l_i is the chain requirement vector of the task, and calculate the connection consumption of the subsequent task divided into two cases: the first type, when there are tasks completed, because the remaining UAVs cannot meet the requirements of the remainder tasks, no task starts, the node will disconnect from the temporary link and connect with its superior node. The connection consumption is:

$$n_d(t_j) = \mathbf{l}(t_j) \cdot (0, 0, 0, 2, 2, 0, 4, 4)^T \tag{6}$$

T_j is the moment when there is no task to start after the some task is completed. $\boldsymbol{l}(t_j)$ is the total function chain requirement for tasks that are completed at t_j times, $n_d(t_j)$ *is* the connection consumption generated at t_j.

The second type is that when there are tasks completed, one or more new tasks begin. At this point, considering that there may be some existing connections reused, the connection consumption is:

$$\begin{aligned} n_r(t_k) = &(\boldsymbol{l_s}(t_k) - \boldsymbol{r}(t_k)) \cdot (0, 0, 0, 3, 3, 3, 5, 5)^T \\ &+ (\boldsymbol{l_e}(t_k) - \boldsymbol{r}(t_k)) \cdot (0, 0, 0, 2, 2, 0, 4, 4)^T \\ &+ r(t_k) \cdot (0, 0, 0, 1, 1, 3, 1, 1)^T \end{aligned} \tag{7}$$

where, $r(t_k) = \frac{|\boldsymbol{l_s}(t_k)+\boldsymbol{l_e}(t_k)|-|\boldsymbol{l_s}(t_k)-\boldsymbol{l_e}(t_k)|}{2}$ is the reusable function chain vector for previous and subsequent tasks. t_k is the moment when a task is completed and there some task begins. $\boldsymbol{l}_s(t_k)$ is the function chain requirement vector needed for the tasks starting at t_k time, and $\boldsymbol{l}_e(t_k)$ is the function chain requirement vector needed for the tasks ending at t_k time, $n_r(t_k)$ is the connection consumption generated at t_k time. It can be seen that the connection consumption n:

$$n = \sum_{t_j \in t_e \backslash t_s} n_d(t_j) + \sum_{t_k \in t_e \cap t_s} n_r(t_k) + n_c \tag{8}$$

where, t_s is a set of times when there is task start and t_e is a set of times when there is task end.

3) AVERAGE USAGE OF UAVs

The function chains L1, L3(I), L3(III), L4(I) and L4(II) all contain UAVs (I nodes), Unlike F and A nodes, I nodes can be reused. During the execution of the task, the number of remaining UAVs is more can make more functional chains available for later tasks, and the task can be completed more smoothly.

$T = t_s \cup t_e$ is the set of times when the UAVs connection changes (For the convenience of calculation, the time in the set T is arranged in ascending order.). The number of UAVs used from time $T(i)$ to $T(i+1)$ is recorded as $s(i)$, so there is:

$$s(i) = \sum_{\substack{t_s(k)\leq T(i)\\ t_e(k)\geq T(i+1)}} l_k \cdot (1, 0, 0, 1, 1, 0, 2, 2)^T \quad (9)$$

The average number of UAVs used is:

$$s = \frac{1}{n(T)} \sum_{i=1}^{n(T)} s(i) \quad (10)$$

where, $n(T)$ is the maximum value of the time when the UAV connection changes.

B. IMPROVED NSGA-III ALGORITHM

After the mission is decomposed into multiple subtasks, the tasks can be transform order under the premise of satisfying the priority order. For any given task sequence, it can be scheduled according to the MPLDCS method described above. This large amount of feasible scheme space will bring great difficulties to the selection of appropriate scheme. Based on the multi-objective optimization method, it is feasible to select the Pareto front solution.

1) IMPROVED ALGORITHM FRAMEWORK

NSGA-III and NSGA-II have the same algorithm framework, and are also composed of population initialization, reference point generation, and so on. In this paper, the encoding mode of the chromosome is constrained by the task optimization execution order of task graph.

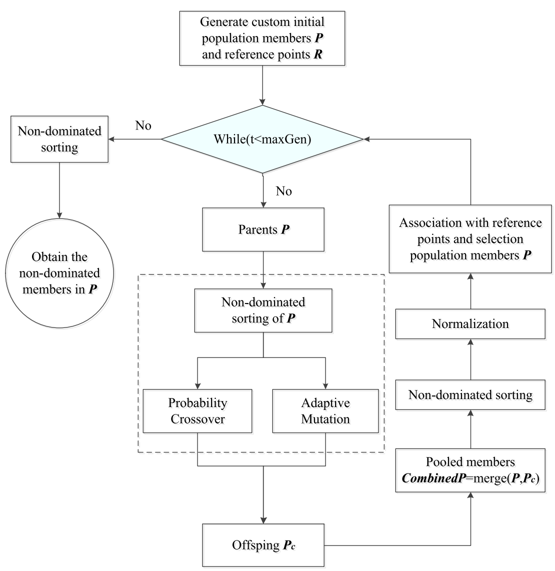

FIGURE 8. Improved algorithm framework.

In order to apply the NSGA-III algorithm to solve the problem, a custom genetic operator needs to be used in the algorithm to ensure the independent variable in genetic variation can keep relative logical order between tasks. The specific algorithm process is shown in Fig.8.

2) LINEAR CROSSOVER OPERATOR

All chromosomes were paired, the probability of crossing is p_c. The crossover probability is related to the non-dominated level, and the level is front, the crossover probability is higher. Define p_c as follows:

$$p_c = \frac{1}{l_1 + l_2 - 2} \quad (11)$$

Since the chromosomes in this paper are encoded in the form of task orders, that is, the two chromosomes of the offspring after the crossover also need to satisfy the relative order. Improve the Order Crossover (OX) method to complete the crossover process. In the first step, a fragment of st $(2 \leq st \leq N_T - 1)$ genes is randomly selected from one parent, and select the same fragment in another parent. The second step is to keep the unselected genes in the two parent chromosomes unchanged, the selected gene segments are reordered according to the order of the other task. This crossover approach ensures that the offspring chromosomes also satisfy the task order constraints. As shown in Fig 9.

3) ADAPTIVE MUTATION OPERATOR

Because of the generated task order, that is, the chromosome needs to meet the task order requirements, this paper uses a customized adaptive mutation operator to perform the mutation operation. First, the chromosome of Pp is selected from P with a probability of pm. Each chromosome of Pp mutates with a probability of p_m.

$$p_m = \left(1 - \frac{1}{2^{l-1}}\right) \cdot \left(p + (t-1) \times \frac{1-p}{t_m - 1}\right) \quad (12)$$

This paper considers that in each evolutionary process, the non-dominant level of chromosome is lower, there should be a higher probability of mutation. In addition, Jiao Hong Yi *et al.* proposed that the probability of mutation is related to the generation of evolution, and the validity is verified [44]. Therefore, a method for calculating the mutation probability of combining the non-dominant level and the evolutionary generation is proposed. Where l is the non-dominated level of the selected chromosome. In the first half, it can be seen that the value of p_m is smaller when the level is front. In the second half, t_m is the maximum evolution generation, t is the current generation, and p is a fixed value, such as 1/50. That is, it is related to the generation of the iteration, and the probability of mutation is higher when generation is greater.

Previous and subsequent tasks of each task i are selected by priority chain to composition set Ai. Then, select a point that can be exchanged in task i, marked as pi. Mutation operator uses two-point exchange to traverse elements except Ai in Pc, and then successively judge whether it is subject to priority constraint after exchange with pi, and obtains the set Bi that can be exchanged, and finally randomly select the points in Bi exchange with pi. The main mutation process is shown in algorithm 2.

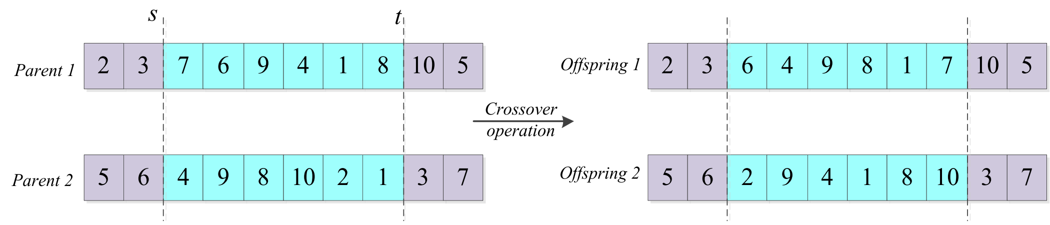

FIGURE 9. Example of crossover operation.

Algorithm 2 Function Mutation-op (*P, Pc,pm*)

Input: Initial population members, ***P*** *of size* ***N****, mutation percentage,* ***pm****, mutation occurring probability,* p_m*, priority order chains,* ***Tor****, related task set,* ***Ai*** $(i \in (1, N_T))$.
Output: Offspring $\boldsymbol{P_p}$

```
1 Pc← popselection(P, pm);
2 t = 1;
3  while t ≦ N-pc
4    k← random (0,1);
5    if k>pm
6     Pp(t)← chromosome (t);
7     k = k + 1;
8     else pi← rand(random((O,NT)));
9      if pi is not exchangeable
10       Go to line 7;
11       else C← Pp(t)\Ai;
12       Bi← exchangeableset(C, Tor, pi);
13       q← rand(random((O,\Bi\)));
14       Pp(t) ←exchange (chromosome(t),pi, B(q));
15      end
16    end
17 end
```

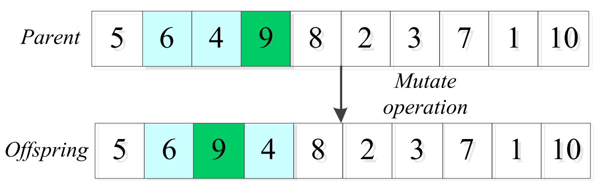

FIGURE 10. Mutation operation.

As shown in Fig.10, the existing task order constrains T4-T3-T1-T10, T5-T9-T8, and T6-T8-T2-T7. We random choose task 9, we randomly select task 9. First, we judge that the A9 set for T9 is {T5, T8}, and traverse elements other than the set of A9, such as T2, because T2 is constrained by T8-T2, so it cannot be exchanged. After judged that {6, 4} can be exchanged with 9 and randomly selected 4.

VI. CASE STUDY

Suppose there is a hybrid operating SoS that accomplishes a military mission, including the direct destruction of a military base and the regional blockade of a region. The SoS has 18 UAVs (I), 40 conventional missiles (F) and 48 loitering Missile (A). Fig.11 is an example of an area for a operational mission.

The mission objectives are decomposed into13 sub-tasks as shown in the right side of Fig.11. Some sub-tasks have a priority and order relationship. The priority relationship include T1-T2-T4-T3, T7-T8-T10, T13-T11-T12, and the strict order relationship are T1-T2, T7-T8. According to the FINCA structure of the combat SoS, the time coordination matrix between the function chains is calculated, as shown in Table 1.

TABLE 1. Function chain time coordination matrix.

	L1	L2(I)	L2(II)	L3(I)	L3(II)	L3(III)	L4(I)	L4(II)
L1	3	5	5	6	6	6	6	6
L2(I)	5	3	5	4	6	4	4	6
L2(II)	5	5	3	6	4	6	6	4
L3(I)	6	4	6	4	6	4	4	6
L3(II)	6	6	4	6	4	6	6	4
L3(III)	6	4	6	4	6	4	4	6
L4(I)	6	4	6	4	6	4	4	6
L4(II)	6	6	4	6	4	6	6	4

TABLE 2. Function chain and execution time requirement list.

Task	L1	L2 (I)	L2 (II)	L3 (I)	L3 (II)	L3 (III)	L4 (I)	L4 (II)	Time /min
T1	3	0	0	0	0	0	0	0	5
T2	1	2	0	3	2	0	1	2	7
T3	0	2	0	0	2	1	0	0	5
T4	1	0	2	1	2	1	0	0	4
T5	2	0	2	1	3	0	0	2	10
T6	2	1	1	0	0	0	3	2	6
T7	3	0	0	0	0	2	0	0	6
T8	0	2	2	0	1	2	0	0	7
T9	1	0	2	0	4	0	0	0	8
T10	0	2	0	3	0	3	0	0	4
T11	2	0	0	0	0	0	2	2	12
T12	2	0	0	0	0	0	2	2	12
T13	2	0	0	2	0	0	0	0	6

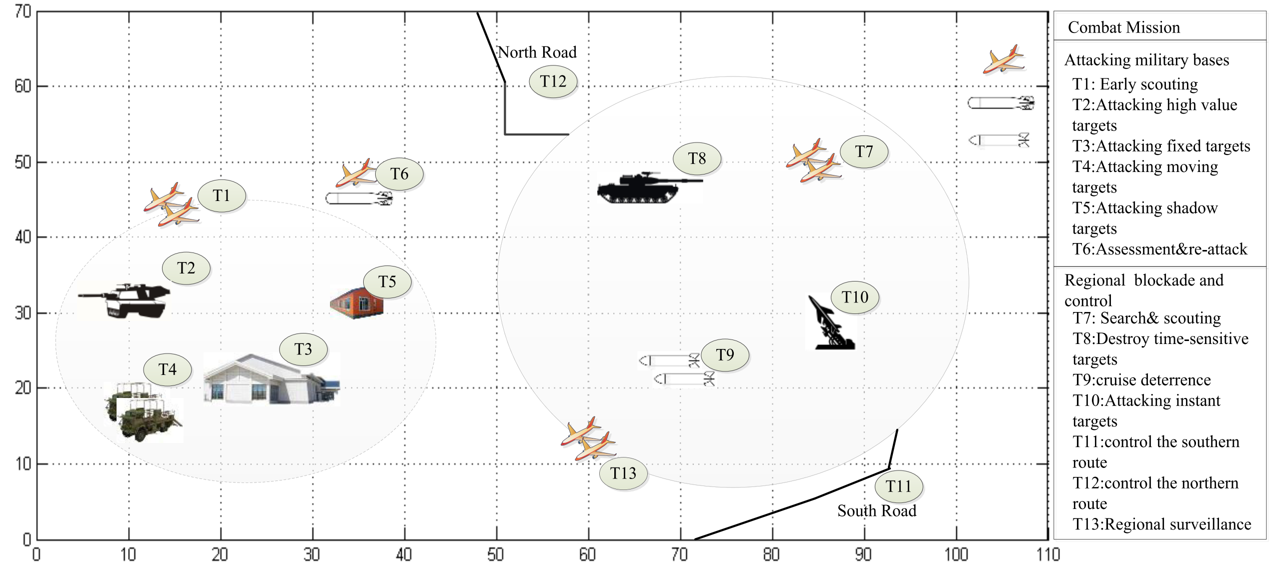

FIGURE 11. Example of combat scenario.

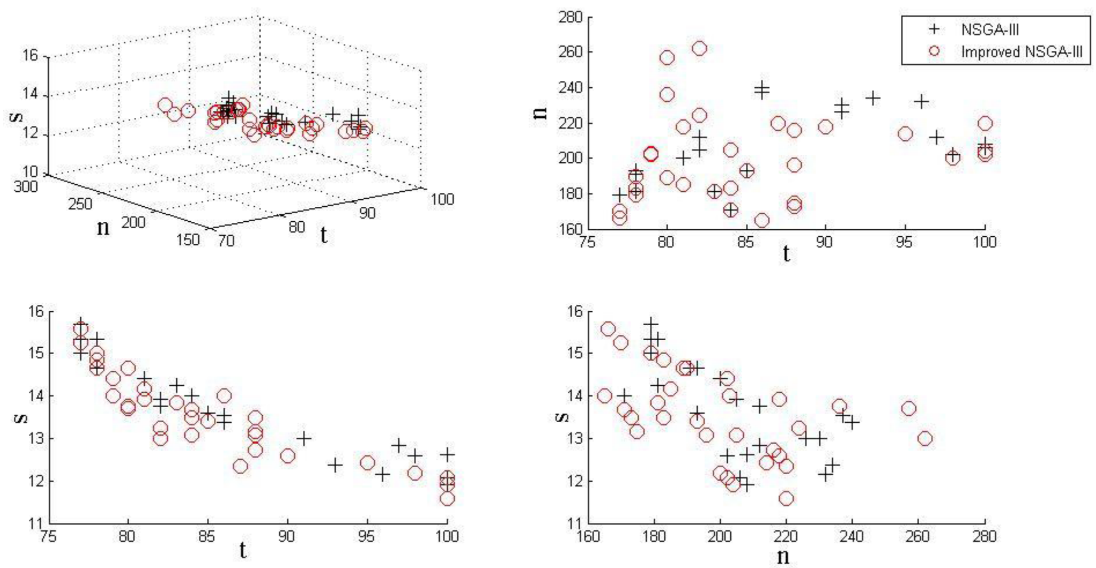

FIGURE 12. Comparative result.

Through the operational capability requirements and functional chain constraints, the task function chain requirement list is obtained, and the task execution time requirement list is analyzed according to the task difficulty and resource conditions, as shown in Table 2.

In the case of only satisfying task priority constraints, there are a total of $C_{13}^4 \cdot C_9^3 \cdot C_6^3 \cdot 3! = 7207200$ solutions that satisfy the task order. Each task order can use the scheduling method described in the section "Function Chain Scheduling Method" for task planning. If traversal the workload is very large. We use the improved NSGA-III algorithm to perform 50 iterations simulation experiments on Matlab, and compare them with algorithm that do not improved crossover and mutation to verify the effectiveness of the method and the advanced nature of the improvement. The results are shown in Fig.12 and Fig.13 below, in which the circle dot represents the improved algorithm and the cross pattern represents the original algorithm. Fig.12 is 3-dimensional population diagram and 2-dimensional results of the two targets respectively. For both methods, a set of non-dominated solutions can be obtained. As can be seen from the lower left-hand corner of Fig.12, the time to complete the mission is negatively correlated with the number of UAV used, which means that the number of reusable resources has a significant impact on

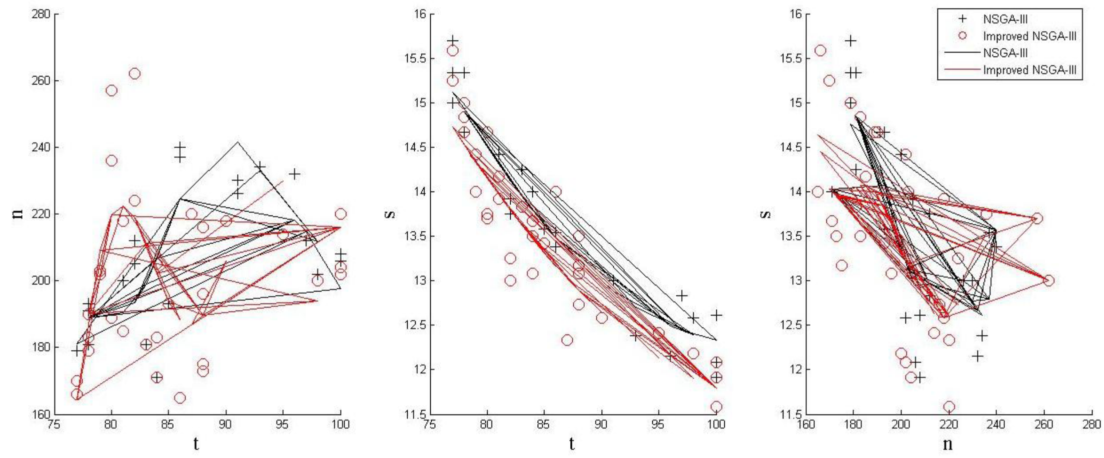

FIGURE 13. Comparative curves fitting result.

mission completion. Fig.13 shows the results of two of the three targets, where the curves are the fitting of the respective results. From the three graphs in Fig.13, it can be seen that the red (improved algorithm) fitting curve is obviously lower, that is, the improved algorithm has a better pareto frontier, especially the middle graph, with almost no overlapping parts, so if decision maker pay more attention to the time overhead and reused resource availability, the improved algorithm has obvious advantages. In general, the improved algorithm can provide more and more advantageous options.

VII. CONCLUSION

Aiming at the characteristics of increasingly complex of combat SoS and higher intelligence of weapon equipment, this paper extends the FINC method to describe complex combat SoS, proposes a function chain-based mission planning method, and perform tasks with the function chain as the basic unit. According to the feature that the function chain is composed of multiple nodes and can reconfigurable, the original scheduling method is improved, so that after the task order is determined, MPLDCS can make reasonable planning. Therefore, the order of tasks is critical to the results of planning, and the NSGA-III method with better convergence and guarantees population diversity is selected and improved to obtain solutions that satisfying multiple objectives.

Due to time constraints, there are still some shortcomings in our research. For example, in order to simplify the calculation, the UAV is defined as having observe function only. In fact, the UAV application with the integrated function of observe and attack has been widely used, which requires define a variety of different *A* nodes. In addition, different methods can be explored to improve the algorithm.

In our future work, the network description method needs to be further improved according to the actual situation, as well as exploring the improvement of the algorithm by using other algorithms for reference.

REFERENCES

[1] L. Bing, “Research on cloud ammuition system theory and method,” Ph.D. dissertation, Dept. Mechatronical Eng., Beijing Inst. Technol., Beijing, China, 2014.

[2] P. Li and H. Duan, “A potential game approach to multiple UAV cooperative search and surveillance,” *Aerosp. Sci. Technol.*, vol. 68, pp. 403–415, Sep. 2017.

[3] H. Leung, “An integrated decision support system based on the human OODA loop,” in *Proc. IEEE 17th Int. Conf. Cogn. Inform. Cogn. Comput. (ICCICC)*, Jul. 2018, p. 1.

[4] K. Deb and H. Jain, “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints,” *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[5] J. Ron, “Dynamic complexity in system of systems,” Internal Rep. Eng. Technol. Adv. Syst., Boeing Company, Chicago, IL, USA, 2007.

[6] D. S. Alberts *et al.*, “Understanding information age warfare,” Assistant Secretary Defense (C3I/Command Control Res. Program), Washington, DC, USA, Tech. Rep., 2001.

[7] H. U. Xiao-Feng, “War complexity and war gaming & simulation in the information age,” *J. Syst. Simul.*, vol. 12, pp. 3572–3580, 2006.

[8] J. R. Cares, *Distributed Networked Operations: The Foundations of Network Centric Warfare*. Bloomington, IN, USA: iUniverse, 2006.

[9] K. M. Carley and K. David, *A Typology for C2 Measures*. Pittsburgh, PA, USA: Carnegie-Mellon Univ., 1999.

[10] A. Mahmoud *et al.*, “Modeling community containment for pandemic influenza,” Committee Model. Community Containment Pandemic Influenza, Inst. Med., Health Med. Division, Washington, DC, USA, Lett. Rep., 2006.

[11] A. H. Dekker, “Applying social network analysis concepts to military C4ISR architectures,” *Connections*, vol. 24, no. 3, pp. 93–103, 2002.

[12] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, “The vehicle routing problem: State of the art classification and review,” *Comput. Ind. Eng.*, vol. 99, pp. 300–313, Sep. 2016.

[13] C. Shao, X. Wang, M. Shahidehpour, X. Wang, and B. Wang, “An MILP-based optimal power flow in multicarrier energy systems,” *IEEE Trans. Sustain. Energy*, vol. 8, no. 1, pp. 239–248, Jan. 2017.

[14] K. E. Nygard, P. R. Chandler, and M. Pachter, “Dynamic network flow optimization models for air vehicle resource allocation,” in *Proc. Amer. Control Conf.*, vol. 3, Jun. 2001, pp. 1853–1858.

[15] Q. Ning, G. Tao, B. Chen, Y. Lei, H. Yan, and C. Zhao, “Multi-UAVs trajectory and mission cooperative planning based on the Markov model,” *Phys. Commun.*, vol. 35, Aug. 2019, Art. no. 100717.

[16] R. J. Szczerba, P. Galkowski, I. S. Glicktein, and N. Ternullo, “Robust algorithm for real-time route planning,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 3, pp. 869–878, Jul. 2000.

[17] M. B. Dias, R. Zlot, N. Kalra, and A. Stentz, ''Market-based multi-robot coordination: A survey and analysis,'' *Proc. IEEE*, vol. 94, no. 7, pp. 1257–1270, Jul. 2006.

[18] L. VigJulie and A. Adams, *Market-Based Multi-robot Coalition Formation*. Osaka, Japan: Springer, 2006, pp. 227–236.

[19] Z. Changwen, M. Ding, and C. Zhou, ''Real-time route planning for unmanned air vehicle with an evolutionary algorithm,'' *Int. J. Pattern Recognit. Artif. Intell.*, vol. 17, no. 1, pp. 63–81, 2008.

[20] M. Dorigo and T. Stützle, ''Ant colony optimization: Overview and recent advances,'' in *Handbook of Metaheuristics*. Cham, Switzerland: Springer, 2019. pp. 311–351.

[21] N. Nigam and I. Kroo, ''Control and design of multiple unmanned air vehicles for a persistent surveillance task,'' in *Proc. 12th AIAA/ISSMO Multidisciplinary Anal. Optim. Conf.*, 2008, p. 5913.

[22] N. Nigam and I. Kroo, ''Persistent surveillance using multiple unmanned air vehicles,'' in *Proc. IEEE Aerosp. Conf.*, Mar. 2008, pp. 1–4.

[23] E. Saad, J. Vian, G. Clark, and S. Bieniawski, ''Vehicle swarm rapid prototyping testbed,'' in *Proc. AIAA Infotech Aerosp. Conf. AIAA Unmanned Unlimited Conf.*, Jun. 2009, pp. 1–9.

[24] H. Van Dyke Parunak, S. A. Brueckner, and J. Sauter, ''Digital pheromones for coordination of unmanned vehicles,'' in *Environments for Multi-Agent Systems*. Berlin, Germany: Springer, 2004, pp. 246–263.

[25] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhangd, ''Multiobjective evolutionary algorithms: A survey of the state of the art,'' *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 32–49, Mar. 2011.

[26] Deb, Kalyanmoy, ''A fast and elitist multiobjective genetic algorithm: NSGA-II,'' *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Aug. 2002.

[27] E. Zitzler, M. Laumanns, and L. Thiele, ''SPEA2: Improving the strength Pareto evolutionary algorithm,'' Comput. Eng. Netw. Lab., Dept. Elect. Eng., Swiss Federal Inst. Technol. (ETH), Zürich, Switzerland, TIK-Rep. 103, 2001.

[28] E. Zitzler and S. Künzli, ''Indicator-based selection in multiobjective search,'' in *Proc. Int. Conf. Parallel Problem Solving Nature*, vol. 3242. Berlin, Germany: Springer, 2004, pp. 832–842.

[29] S. Yang, M. Li, X. Liu, and J. Zheng, ''A grid-based evolutionary algorithm for many-objective optimization,'' *IEEE Trans. Evol. Comput.*, vol. 17, no. 5, pp. 721–736, Oct. 2013.

[30] S. Ying, L. Li, Z. Wang, W. Li, and W. Wang, ''An improved decomposition-based multiobjective evolutionary algorithm with a better balance of convergence and diversity,'' *Appl. Soft Comput.*, vol. 57, pp. 627–641, Aug. 2017.

[31] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina, ''A preference-based evolutionary algorithm for multi-objective optimization,'' *Evol. Comput.*, vol. 17, no. 3, pp. 411–436, Sep. 2009.

[32] H. Jain and K. Deb, ''An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach,'' *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 602–622, Aug. 2014.

[33] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, ''Network motifs: Simple building blocks of complex networks,'' *Science* , vol. 298, no. 5594 pp. 824–827, Oct. 2002.

[34] G. Pan, J. Tang, and F. Guo, ''Analysis of co-associated transcription factors via ordered adjacency differences on motif distribution,'' *Sci. Rep.*, vol. 7, Feb. 2017, Art. no. 43597.

[35] Y. Lee and T. Lee, ''Network-based metric for measuring combat effectiveness,'' *Defence Sci. J.*, vol. 64, no. 2, pp. 115–122, 2014.

[36] L. Deyi, ''Network Warfare and Complex Network,'' *Chin. Military*, vol. 19, no. 3, pp. 111–119, 2006.

[37] D. Zheng and G. Ergün, ''Coupled growing networ,'' *Adv. Complex Syst.*, vol. 6, no. 4, pp. 507–514, 2003.

[38] S. Haider and A. H. Levis, ''An approximation technique for belief revision in timed influence nets,'' *George Mason Univ Fairfax va Syst. Archit. Lab*, vol. 27, no. 2, pp. 6–7, 2007.

[39] E. D. Sacerdoti, ''Planning in a hierarchy of abstraction spaces,'' *Artif. Intell.*, vol. 5, no. 2, pp. 115–135, 1974.

[40] C. Qi, D. Wang, M.-A. Héctor, P. Zhao, and H. Wang, ''Hierarchical task network planning with resources and temporal constraints,'' *Knowl.-Based Syst.*, vol. 133, pp. 17–32, Oct. 2017.

[41] Z. Aiqing and M. Zeyao, ''A new scheduling algorithm for digraph-based parallel computing,'' *Chin. J. Comput.*, vol. 32, no. 11, pp. 2178–2186, 2009.

[42] L. Kankan, ''Research on task scheduling method in heterogeneous multi-core environment,'' M.S. thesis, School Comput. Sci. Technol., Hangzhou Univ. Electron. Sci. Technol., Hangzhou, China, 2009.

[43] R. Fogleman, *Targeting—Air Force Doctrine Document 2-1.9, Air Force Doctrine Center*, Air Force BA, USA: Maxwell, 2006.

[44] J.-H. Yi, S. Deb, J. Dong, A. H. Alavi, and G.-G. Wang, ''An improved NSGA-III algorithm with adaptive mutation operator for Big Data optimization problems,'' *Future Generation Computer Systems*, vol. 88, pp. 571–585, Nov. 2018.

[45] Y. Dongsheng, ''Effective measurement and design of C2 organization,'' Ph.D. dissertation, Dept. College Inf. Syst. Manage., Nat. Univ. Defense Technol., Changsha, China, 2004.