IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received September 16, 2019, accepted October 24, 2019, date of publication October 29, 2019, date of current version November 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2949989

An Operation Planning Generation and Optimization Method for the New Intelligent 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 by the Doctoral Science Foundation-Funded Project of China under Grant 45832.

ABSTRACT With the rapid development of emerging technologies, the unmanned and intelligent degree of combat forces in the combat system of systems(SoS) has been continuously improved, making the operational equipment more diverse. How to better describe the multiple functions and forces hybrid combat SoS becomes a difficult problem. Changes in combat forces have transformed the operational style from single platform to cluster, which puts forward higher demands on the timeliness of the development of course of action(COA), which is the key step in the formulation of operational planning. It is of great military value to study the model of the hybrid new complex combat SoS, as well as the generation and optimization of the operational plan under the new combat model. This paper extends the FINC (Force, Intelligence, Networking, C2) model of heterogeneous networks, adds a variety of new intelligent unmanned combat force nodes, and proposes a new Force Intelligent Network C2 and Autonomous Model (FINCA^2) which represents more different types of function nodes and communication relationships between nodes. Based on this, this paper proposes six types of 12 kinds functionally simple, frequently used, and practically function chains (FC) for the new combat SoS. Through the Joint Mission Thread (JMT) method, the operational mission is decomposed to get specific operational actions, action relationships, and requirements list of function chains for the actions, and complete the matching of actions and resources. Based on the idea of *sigmoid* function in neural network, the multi-objective optimization algorithm NSGA-III is improved to find the Pareto frontier of the action order scheme that satisfies the task association relationship. Finally, the method proposed in this paper is verified by the case of a brigade-level combat SoS participating in border operations.

INDEX TERMS Intelligent combat SoS, FINCA^2, function chain, JMT, improved adaptive NSGA-III.

I. INTRODUCTION

The combat SoS is composed of operational entities with various functions. With the development of information technology, the types and functions of weapons and equipment are more diverse. For example, new intelligent weapons such as cruise missiles and single-handed drones are not only low-cost/high-mobility, but also with multiple functions [1], and the interaction between entities is more complicated and diverse. With the extensive use of new technologies, modern warfare presents the characteristics of accelerating combat rhythm, fierce confrontation between the enemy and me, and the interweaving of various uncertain factors. Therefore, the correct, timely and stable battlefield command faces more severe challenges. The outcome of the war no longer depends solely on the number of platforms and the functions of the individual platforms, but on the efficient operational plans formulated on the basis of rapid integration of battlefield resources.

Because of the high antagonism of combat operations, the traditional manual operation mode can not meet the requirements of rapid planning, so it is necessary to use information technology to improve the efficiency of planning and adjustment [2]. To integrate resources and improve efficiency of information means, first of all, we need to find an appropriate description of the combat SoS. The study of combat SoS based on complex networks has been studied for a long time [3], and the existing network considers the

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

heterogeneity of military network nodes, such as the FINC method [4]. But the node type and connection relationship are not very rich, and do not consider the new features brought by the multi-function node to the network of the combat SoS. Therefore, it needs to be improved and extended to a more diverse FINCA^2 network model.

The operational planning can be divided into two stages: combat action order generation and combat resource matching [5]. In the process of traditional combat resource matching, the existing research considers the single function of a single node, but for multiple functions, it often results in waste of resources and some tasks can not be satisfied. According to the OODA loop theory [6] and the F2T2EA theory of the kill chain [7], the operational plan can be divided into different steps. This paper analyzes the function requirements of the equipment in each step, combines with the idea of the network model [27]–[29], and combines different types of node connections into different types of function chains to meet the needs of each operational phase. The method based on function chain can reduce the connection and interaction between nodes, and is not easily affected by communication interference. It can also fully exert the capabilities of each node.

How to decompose the operational order to get a detailed order of operations and how to map the function chain to the operational order is a problem that must be solved. Using the method of JMT of the US Army for reference [8], the combat mission is decomposed into basic operational actions, and a list of basic function chain requirements for each operational action is obtained. There is usually a strict sequential logic relationship between the actions obtained by JMT decomposition, but not all operations have a strict order relationship. Under the premise of meeting the constraints of operational logic relationships, different operational schemes have an impact on operational time, the consumption of communication connections, and the instantaneous availability of drones. Planning and selecting a combat action plan is a typical multi-objective optimization problem.

The NSGA-III, an evolutionary multi-objective algorithm based on reference points, is selected as the basic optimization method. Compared with other algorithms, it has the advantages of fast convergence and maintaining population diversity [9]. According to the characteristics of the combat action generation problem, the algorithm coding method is improved to meet the timing logic constraints. In order to improve the effectiveness of the scheme, adaptive genetic operators are used to generate the descendant populations, and the crossover and mutation process of NSGA-III algorithm is improved through the neuron activation function, which can produce more and better populations.

The rest of the paper is structured as follows. The second part summarizes the work related to the network description of the combat SoS network, the generation of operational plans, and the multi-objective optimization method. The third section introduces the concept of the FINCA^2 method and proposes multiple function chains based on this. In the fourth section, the JMT method is described in detail how it can be applied to combat action decomposition and function chain requirements mapping. The fifth section mainly introduces the improvement process of the algorithm. The effectiveness and advancement of the proposed method is verified by the case of Section VI. Finally, the conclusions and future work are discussed.

II. BACKGROUND AND RELATED WORK

In order to establish a suitable network model of the new intelligent combat SoS, generate the operational planning based on this, and optimize the planning scheme, it is necessary to sort out and summarize the research status of the network modeling of combat SoS, the generation method of operational planning, and the multi-objective optimization method.

A. COMBAT SOS NETWORK

The combat SoS can be seen as a "super system" that interacts with multiple function entities to achieve common goals [10]. Due to the complexity of its function entities and the multiple interrelated flows of information between entities, modeling complex military systems is a challenging task [11]. The rapid development of network science has brought new hope. In fact, there are many studies on complex network representation military organizations [12]–[14].

Cares used the concept of complex networks and proposed the IACM model in the information age, which divided the forces into enemy targets, decision entities, sensor entities and influential entities, and described the flow of information between entities [15]. But the model does not consider other attributes of the entity, such as the behavior and capabilities of the entity. Robinson et al. extended it and proposed a directed graph model, which takes into account entity behavior and its ability to accomplish specific tasks [16]. Carley et al. proposed the PCANC model for military organizations [17], but the nodes and edges are uniform in these methods and are not distinguished according to specific research questions and backgrounds. Dekker studied the relationship between network topology and operational effectiveness, and builded a FINC model to support heterogeneous nodes describing the operational network, and based on this, analyzed the C4ISR System of air combat and land warfare [18]. As the FINC model did not consider the behavior and capabilities of the entity, Yang et al. subdivided the type of the edge and considered the weight of the node to extend the FINC method [19]. Based on the existing results of FINC and combined with the characteristics of the research, this paper expands the edges and types of FINC, proposes the FINCA^2 model, and then extracts the appropriate function chain model based on the idea of network motif.

B. BATTLE PLAN GENERATION METHOD

Battle plan generation can be divided into three categories: traditional battle plan generation, effect-based battle plan generation, and uncertain battle plan generation.

Battle planning can be divided into two phases: battle action order generation and battle resource matching. The battle action order generation is a key basic step. Therefore, in order to quickly develop a more effective battle plan, it is necessary to conduct research on the generation method of the battle action order. In the current research, traditional action order generation can be divided into two types: forward planning and reverse planning [20]. Forward planning is to directly find the solution to achieve the goal from the initial situation, but it is difficult to find a suitable solution in the face of the situation that has never occurred. Reverse planning mainly refers to the planning method based on the target decomposition, which decomposes the task into sub-tasks from top to bottom and finds out the action plan. Many qualitative and quantitative methods are used in the solution, such as state space planning, genetic algorithms, Bayesian networks, sequential logic, workflows, and cases [21].

With the development of military technology, the battle mode has changed from traditional linear operations to multi-dimensional and multi-objective coordinated nonlinear operations. In response to this transformation, the US military has proposed Effect Based Operation (EBO) [22], which has been successfully applied to several wars. The US military has developed CAESARII/EB, PYTHIA, CAT, SIAM and other systems [23]–[25] to support effect-based operational planning. EBO actually adds a non-linear requirement to the method of reverse planning and can express operational uncertainty. The uncertainties between tasks and effects are mainly described by Influence Net (IN) [26]. But there is also some controversy about the effect-based approach, which mainly considers that battle is full of uncertainties and that it is logically impossible to accurately predict the outcome of an action.

Uncertain battle plan generation is mainly aimed at the insufficiency of probabilistic network for uncertain events, and is improved by improving the adaptability of the plan and strengthening the robustness of the plan. To enhance the adaptability of the plan is to adjust the plan dynamically according to the battlefield situation, and to enhance the robustness means that the plan is not easily affected by the uncertainty. The method of strengthening robustness inevitably brings redundancy, which often results in poor real-time efficiency of the plan. The method of improving adaptability has higher requirements for battlefield perception and real-time adjustment.

The existing research on the generation of battle plans has achieved a lot of results, but the relationship between the action order and the operational effects in the existing methods is still not clear enough, and the ability of the action order to deal with the uncertainty of the battlefield is still not strong. R. Milo et al. proposed concept of motif based on the repetitive appearance of some important connection structures in the network [27]. The validity of motif has been verified [28], and has also been studied in military applications [29]. Based on this, we propose a method of action order generation based on function chain. Based on the function chain method, we can integrate multiple functions into one, form a strong mapping relationship with combat missions, and make up for the defects with uncertainties. In addition, different function chains, especially the autonomous capability of the cluster autonomous chain formed by the new intelligent weapons, can automatically make adjustments when the environment changes, and can make up for the deficiency of dealing with the uncertainty of battlefield.

C. MULTI-OBJECTIVE OPTIMIZATION METHOD AND IMPROVEMENT NSGA-III

As can be seen from the introduction of the battle plan generation method, many evolutionary algorithms have been applied to the research of action order generation. At the end of this paper, the problem is transformed into a multi-objective optimization problem. Many methods have been developed since the development of multi-objective optimization algorithms. Initially aimed at two objectives, such as NSGA-II [30] and SPEA2 [31] based on Pareto dominance, these methods are often not ideal for three or more objectives because the proportion of non-dominant solutions in the population is very high. In the actual application process, many researchers have improved the multi-objective evolutionary algorithm to improve the search ability of the evolutionary algorithm based on the Pareto dominant strategy to obtain a satisfactory solution. The improving methods mainly include an Indicator Based Evolutionary Algorithm (IBEA) that guides the search optimization process of a population through various performance indicators [32], and introducing a new dominant relationship improvement strategy, such as Grid-based Evolutionary Algorithm (GrEA) [33], which introduces the natural partition advantage of grid to enhance the pressure of optimal direction selection based on grid evolutionary computation. Preference-Inspired Co-Evolutionary Algorithm (PICEAs) avoids random search for high-dimensional and multi-objective optimization problems by guiding the optimization scheme to the Pareto frontier [34], [35]. These methods have achieved satisfactory solutions for specific problems, but there are still some shortcomings when they are widely used. For example, in the performance-based method, when the number of targets is large, the overhead of performance index calculation is huge. In grid-based evolutionary methods, the new dominance relationship is closely related to the problem itself. In the method based on preference information, the optimization efficiency is not high due to the randomness of preference information.

The NSGA-III is based on the target decomposition improvement strategy proposed by changing the selection mechanism in the algorithm framework of NSGA-II. Instead of the way NSGA-II selects the crowd with the same non-dominated level, NSGA-III generates a set of predefined and rule distribution reference points to maintain various candidate solutions to maintain the diversity of various candidate solutions, and ultimately enhance the convergence of the algorithm. NSGA-III has been proved to be superior to most multi-objective optimization methods in many multi-objective optimization benchmarks [36].

In order to improve the performance of NSGA-III and the ability to solve specific problems, many researchers have also improved NAGA-III. For example, when selecting a reference point in the NSGA-III algorithm, there will be a small number of individuals in the associated population, which leads to the loss of population diversity. Amin et al. modify the rules of the reference point, and the population selection rule is divided into two steps. The first step is to select one population individual for each reference point, and the second step is to follow the original method according to the remaining population number, so as to retain more and better elite groups [37]. Bi and Wang used the K-means clustering algorithm to divide the target space into several subspaces, and exchanged individuals between subspaces in the crossover phase, named NSGA-III-OSD [38]. Chen and Yuan improved the genetic strategy of the algorithm, using a dual-population coordinated archiving strategy, that is, the parent and the offspring participate in the genetic process at the same time, and used the ε-dominant mechanism and the constraint violation strategy to select more and better Pareto frontiers to solve the reservoir flood control dispatching problem [39].

As can be seen from the above description, the FINC method can describe the combat network of heterogeneous nodes, but for the new intelligent SoS, the nodes and edges of the FINC model can not meet the requirements. In view of this, we need to add a new type of intelligent node (Autonomous node), clarify the connection relationship between nodes, and several function chains are summarized on this basis and extracted as the basic unit of execution action. JMT refers to the end-to-end operational and technical description of operational activities and systems for the execution and completion of a joint mission. The JMT method is used to guide the decomposition of combat mission into several battle activities with temporal logic relationship, and to complete the mapping between operational activities and function chains. In order to improve the timeliness of mission completion and other indicators, it is necessary to improve the multi-objective optimization algorithm to find the order of actions that meet the needs.

III. HETEROGENEOUS COMBAT NETWORKMODEL

Dekker applies the traditional social network analysis (SNA) to the military field, and the FINC model proposed by Dekker, supports the description of heterogeneous nodes. But for new intelligent combat weapons, FINC nodes types and edges can not describe. Therefore, the FINCA^2 model is constructed by extending the FINC model, adding autonomous node (A) and corresponding edges to meet the actual needs of the combat SoS.

A. NETWORK MODEL OF COMBAT SOS

The FINC model is composed of different types of network nodes such as operations, intelligence, and command, as well as the side that describes the communication associations and communication capabilities between nodes. Based on this model, the unit coordination ability, network centrality and observe performance of combat organization can be analyzed. But for the cruise missile and the observe-attack integrated UAV in the new battle force, they not only have observe ability, but also can attack, and can make decisions independently, furthermore the UAV is a reusable resource. For this kind of hybrid multifunction node, the existing FINC model can not provide the type of node, so it is necessary to add the type of node. This kind of hybrid node is defined as autonomous node. Because this kind of node has the process of self-detection, self-decision and attack, and this process is expressed as self-cycling edge.

As shown in Fig.1, it is a simple example of a brigade-level combat organization with a mixture of people and unmanned. The rectangular nodes represent the force unit (F), the rounded rectangular nodes represent the intelligence unit (I), and the round nodes represent the command and control unit ($C2$), the diamond node represents the autonomous unit (A), the C2 nodes of different colors represent different levels, the A nodes of different colors represent different types, the connected edges represent the information transmission channel (N), and the weight on the edge represents the transmission delay on the information channel (in terms of time step), the information delay for the self-looping loop on the A-node describes the time that is from the node get observe information to its own decision.

The combat SoS can be expressed as FINCA^2 network, $G = (V,E)$, where $V = F \cup I \cup C \cup A1 \cup A2 = \{v_1, v_2, v_3, \ldots, v_n\}$ represents the set of nodes of the combat system, which can divided into sets of nodes of various types, F node set $F = \{v_1^F, v_2^F, v_3^F, \ldots, v_s^F\}$, I node set $I = \{v_1^I, v_2^I, v_3^I, \ldots, v_k^I\}$, $C2$ node set $C = \{v_1^{C2}, v_2^{C2}, v_3^{C2}, \ldots, v_l^{C2}\}$, $A1$ node set $A1 = \{v_1^{A1}, v_2^{A1}, v_3^{A1}, \ldots, v_p^{A1}\}$, $A2$ node set $A2 = \{v_1^{A2}, v_2^{A2}, v_3^{A2}, \ldots, v_q^{A2}\}$. The edge set between the function entities $E = \{e_1, e_2, e_3, \ldots, e_w\} \subseteq V \times V$, that is, the information communication method is as shown in the directed edge of Fig.2.

B. CONCEPT OF THE FC

According to the concepts of OODA, kill chain and motif, the function chain should be a basic capability unit with recurring features. Combined with the characteristics of the hybrid combat SoS, the following six types of twelve function chains with simple structure, frequent use and practical significance are proposed, and corresponding function chain models are established. Among them, F node is conventional tactical missile and other conventional fire systems, I node is satellite, radar and other observe systems, and C node is corresponding command and control system of each node. In general, $A1$ and $A2$ are new types of intelligent weapons.

(1) Observe chain, as shown in Figure 3, is connected by the command node (C) and the intelligence node (I). According to the requirements of the combat mission, the command node sends orders to the observe system, satellite or radar observe the target area, and sends the observe information

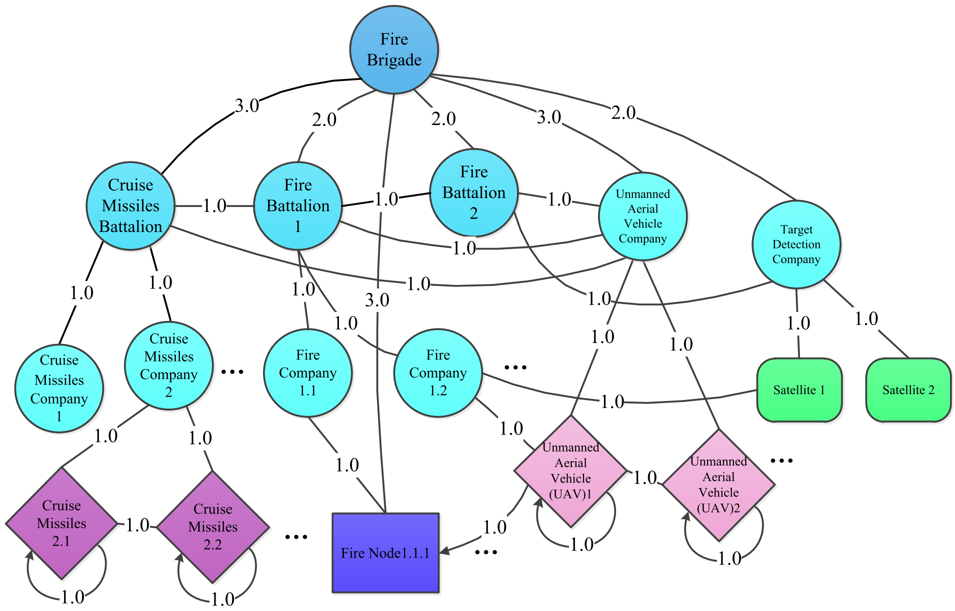

FIGURE 1. Example of FINCA^2 Model.

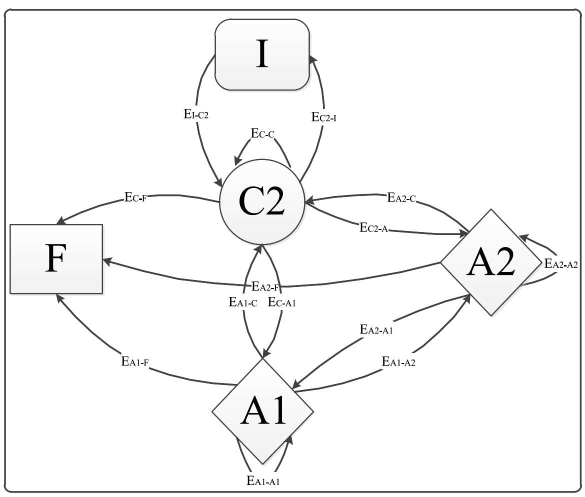

FIGURE 2. Complex military organization abstract representation model.

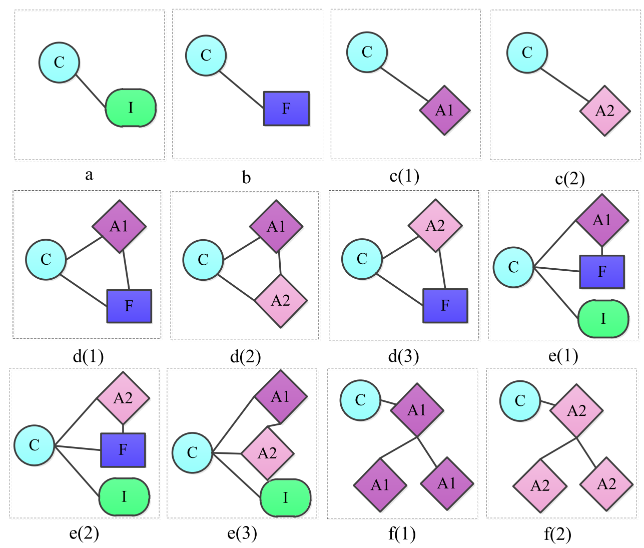

FIGURE 3. Basic function chain model.

back to the command node. The main function of this chain is observe, recorded as L1.

(2) Attack chain, as shown in Figure 3, is connected by the command node (C) and the firepower node (F). The command node sends attack commands to the fire node according to the task assigned by the superior or the information transmitted by the investigation system, and the fire node fires at the designated target and area. The main function of this chain is to attack, recorded as L2.

(3) Observe and attack chain, as shown in c(1) and c(2) in Figure 3, is connected by the command node (C) and the new smart node (A). The command node sends observe and attack commands to new intelligence node according to the task assigned by superior or the information transmitted by the observe system, aiming at moving or masking targets, etc. The new intelligence node gives observe and attack commands to the designated target and area. The main functions of this chain are observe and attack, which are respectively recorded as L3(I) and L3(II).

(4) Patrol and attack chain, as shown in d(1), d(2), and d(3) in Figure 3, is connected by the command node (C), the new autonomous node (A), and the firepower (F) or the new smart node (A). The command node aims at small size target or small cluster maneuvering target according to the information transmitted by the superior's task or observe system. The new intelligence node guides the fire node (or the new intelligence node) to observe and attack the designated target and area.

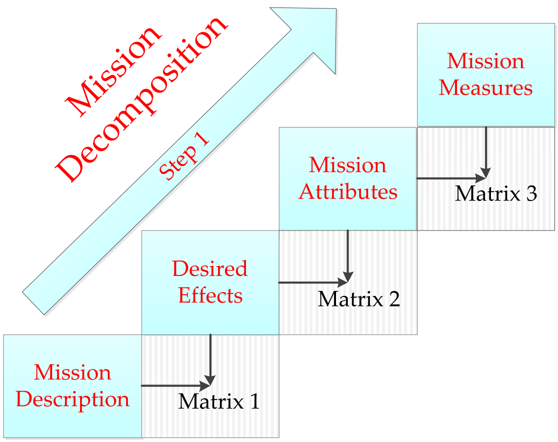

FIGURE 4. Mission level decomposition diagram.

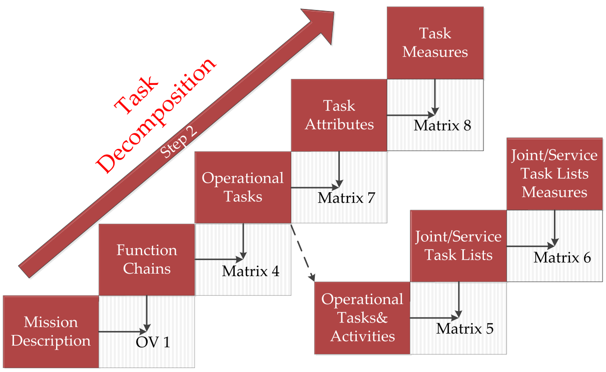

FIGURE 5. Mission level decomposition diagram Task-level measures decomposition diagram.

The main function of this chain is the multiple attacks under the observe guidance, which are recorded as L4(I), L4(II) and L4(III) respectively.

(5) Observe, attack and evaluation chain, as shown in e (1), e (2) and e (3) in Figure 3, is connected by the command node (C), the new autonomous node (A), and the firepower (F) or the new smart node (A). The intelligence node provides the structure of observe situation in real time. According to the task assigned by the superior or the information transmitted by the observe system, the command node aims at the key high value targets, or high-threat fixed-point targets and issues observe and attack orders to the new intelligence node. The new intelligence node guides the fire node (or the new intelligence node) to observe and attack the designated target and area, and then evaluates the attack effect according to the information returned by the new intelligence node and the intelligence node, so as to determine when to attack again. The main function of this chain is the multi-attack and effect evaluation under observe guidance, which are recorded as L5(I), L5(II) and L5(III) respectively.

(6) Cluster autonomous chain, as shown in f(1) and f(2) in Fig.3, is consist of several (3 or more) new autonomous nodes (A). According to the command of the command node, pilot (missile) leads slave aircraft (missile) to conduct observe and attack, and the slave (missile) is commanded by pilot (missile). The cluster autonomous chain is mainly for communication limited, large area shadowing targets. The main function of this chain is the cluster independent and flexible observe or attack, which are respectively recorded as L6(I) and L6(II).

The connection of the function chain is not always constant, and can be reorganized according to the needs of the combat process. Some connections can be reused multiple times to reduce the communication and time consumption caused by frequent changes in the connection. For example, links L3(I) and L3(I) can be reorganized to form L4(II) as needed, and then the remaining part of L4(III) after striking at the F node can be reconstructed with the original L2 link to form a new L4(III) chain.

IV. JMT-BASED COMBAT MISSION DECOMPOSITION AND FUNCTION CHAIN MAPPING

After the combat mission is given, it needs to be decomposed into specific operational activities, and the temporal logic relationship between activities and the matching relationship between activities and resources are given. The US military JMT itself is to complete the combat mission decomposition and end-to-end mapping the operational activities to the execution systems. JMT is not only the bridge connecting military missions, equipments and resources, but also provide an assessable and optimized technical method for the U.S. military to find joint operational capability. Using the process of JMT for reference, the combat mission is decomposed and the mapping of operational activities to function chains is established.

Similar to JMT, the input conditions for the decomposition process include: system parameters, function chain attributes, joint concepts (articles, organization, materials, facilities, etc.), operational environment, and threats. The output product are two kinds of views: the first is a generic data description view, and the second is a task decomposition diagram with temporal logical relationships and the mapping diagram of tasks and function chains.

This paper combining with the characteristics of combat concept, based on the JMT method, the index SoS of combat mission is decomposed, which mainly includes three steps: one is the decomposition of mission-level index; the other is the decomposition of tasks and activities-level indexs; the third is the decomposition of system/function chain-level indexs. The decomposition content is shown in Table 1. The third function chain index decomposition corresponds to the requirements of the function chain in this paper, that is, the list of the requirements of the function chain is given at the end.

STEP 1: MISSION-LEVEL DECOMPOSITION

These include: mission description, desired effect, mission attribute and mission measures. A mission is a task with a purpose that describes the actions to be taken and the reasons for the activity to be carried out; the effect describes the function chain behavior in the combat environment; the expected

TABLE 1. Function chain time coordination matrix.

Operational demand indexs	Mission measures	Mission completion effect	Combat mission requirements
	Task(activity) measures	Task performance or Work Efficiency	Combat task requirements
Functional demand indexs	SoS measures	Functional measurement of the SoS	SoS function requirements
	Function chain measures	Functional measurements of function chains	Functional requirements of function chains

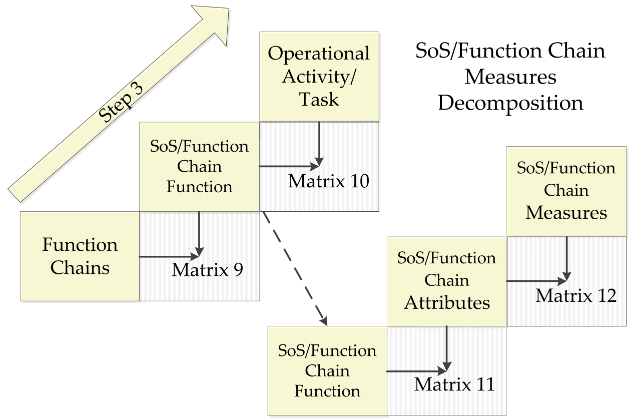

FIGURE 6. SoS/Function chain-level measureS decomposition graph.

effect is the condition to achieve the goal; the attribute is defined as an element or quantitative or qualitative feature of the action; and mission measures is used to measure mission effectiveness. The data used for evaluation is correctly described as utility measurements. Through the relationship matrix between each two items, the mission measurement indicators are finally obtained.

STEP 2: TASK-LEVEL DECOMPOSITION

These include the operational chains: operational chains are the key function chains used to accomplish missions; operational activities: a task is an action or activity, and the activity is equivalent to the task; universal joint task list (UTJL) is a written list of joint tasks; the UJTL provides indexs for each task; task attributes: similar to the mission attribute, it is an element or quantitative or qualitative feature of the action. At the task level, the focus is still on action and task measures. Each task have measurement attributes that is expected to be associated. Through the relationship matrix between each two items, the operational task measurement indexs and the temporal logic relationship between the tasks are finally obtained.

STEP 3: SoS/FUNCTION CHAIN-LEVEL DECOMPOSITION

SoS/function chain index decomposition is mainly to clarify the relationship between function chains and SoS/function chain functions, SoS /function chain functions and operational activities, SoS /function chain functions and attributes, SoS /function chain attributes and measures, as shown in Fig.6.

Function Chain / SoS Functions: Around support running operational chains, support completion of operational activities, the role or utility required of function chains / SoS.

Function Chain / SoS Attributes. A quantitative or qualitative feature of a function chain or SoS.

Function Chain/SoS Measures. Function chain or SoS measures are used to measure the effect of function chain or SoS, can be used to measure the effect, usually can be understood as the tactical and technical indicators of the SoS or function chain.

Through the relationship matrix between each two items, the SoS /function chain measurement indexs and the demand matrix of the operational tasks and function chains are finally obtained.

V. IMPROVED NSGA-III ALGORITHM

Although the task order is constrained by the temporal logic, but the scheme space for the execution order is still huge. Based on the multi-objective optimization method, it is feasible to select the Pareto frontier solution.

In order to improve the efficiency and quality of evolutionary algorithms, people try to combine various methods and data structures with evolutionary algorithms. Such algorithms have been very successful in practice and have become a potential and rapidly development research area [40], [41].

The aim of this paper is to generate operetion planning and optimization is accomplished through course of acion (COA), so encodes chromosomes directly in the form of task orders. Therefore, in order to ensure that the offspring generated during the genetic operation do not break the constraints of the temporal logic, the crossover and mutation methods need to be standardized.

In order to reduce the destructiveness of traditional crossover operators, the literature [42] proposed combining the principles of immunology in life sciences with genetic algorithms, based on immune operators (using domain knowledge) to repair and improve the results of crossover and mutation, improve the performance of the algorithm. Using the idea of the immune operator, in the process of crossover and mutation, the temporal logic constraint is regarded as an immune vaccine, and the offspring generated are selected to obtain the offspring that satisfy the constraint.

A. THE BASIS OF ALGORITHM IMPROVEMENT

The NSGA-III algorithm is consistent with the traditional genetic algorithm and NSGA-II, there is also a crossover and mutation operator. If we choose a constant crossover and mutation probability value, it is easy to fall into local optimum or premature. In order to overcome the shortcomings caused by the fixed parameters, adaptive dynamic adjustment parameters are usually adopted to avoid premature. In the

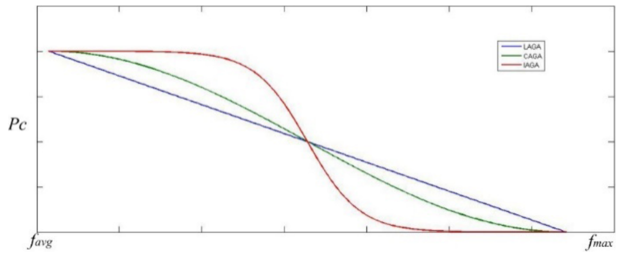

FIGURE 7. Comparison of IAGA CAGA LAGA adaptive adjustment curves.

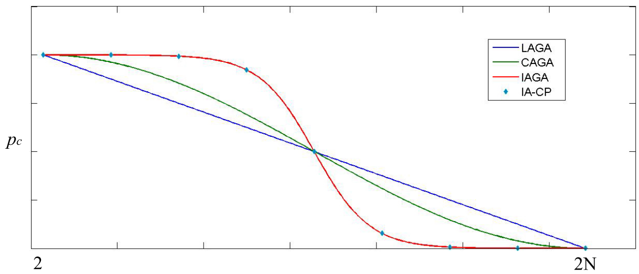

FIGURE 8. Schematic graph of crossover probability change.

research of traditional genetic algorithm, the literature [43] proposed the cosine adaptive genetic algorithm (CAGA), but the performance of linear adaptive GA (LAGA) is similar when the difference between f_{avg} and f_{max} is large.

Firstly, the adaptive adjustment curve of the crossover rate and the mutation rate should be slowly changed nearby the f_{avg}, thereby increasing the crossover rate and the mutation rate of the individuals whose fitness is close to the average fitness. Secondly, ensure that the adaptive adjustment curve does not tend to be linear when the difference between f_{avg} and f_{max} is large. Finally, to ensure that the better individuals in the population still have a certain crossover rate and mutation rate. At the same time, in order to retain the better individual as much as possible, the adaptive adjustment curve at f_{max} should be smoothed.

In neural networks, the most commonly used to constructor neuron activation functions is the *sigmoid* function, as shown in equation (1). This function shows a good balance between linear and nonlinear behavior [44], [45].

$$\varphi(v) = \frac{1}{1 + \exp(-av)} \quad (1)$$

As shown in Fig.7, the sigmoid function has a smoother top and bottom than the cosine function, it belongs to improved adaptive genetic algorithm (IAGA).

Based on the idea of the above-mentioned genetic adaptive operator, the adjustment of the crossover and mutation probability of NSGA-III is made. And based on the encoding method customized in this paper, the reasonable rules of crossover and mutation are determined.

B. ADAPTIVE CROSSOVER OPERATION

Pairing all chromosomes, the probability of crossover is p_c, and the probability of crossover is related to the non-dominated level, that is, the process obeys the *sigmoid* function, as shown in the improved adaptive crossover operator (IA-CP) point in Fig.8.

$$p_m = \frac{1}{1 + \exp\left(2A\,(l_1 + l_2)/_N\right)} \quad (2)$$

where A is a constant equal to 9.903438, l_1 and l_2 are the non-dominated levels which the two chromosomes are selected, and N is the largest non-dominated level.

The crossover method is completed by an improved order crossover operator method, which performs a conventional double-point crossover and then performs a modification of the patrol route that maintains the original relative access sequence. The specific crossover is as follows: First, randomly select two crossover points on the parent chromosome to determine the crossover area, for example, the two parent chromosomes are: parentl = 12|3456|789, parent2 = 13|5792|468, and the crossover area of parent2 and parent1 are exchange to obtain two new chromosomes: parentlc = 12xxxx789, parent2c = 13xxxx468, then other elements in parent1 and in parent2 insert into parentlc, parent2c in the order of the other chromosomes, the final two sub-chromosomes were obtained: childl = 123546789, child2 = 132579468. The sub-chromosomes obtained by such crossover methods can still satisfy the constraints of the temporal logic relationship.

C. ADAPTIVE MUTATION OPERATION

In this paper, adaptive mutation probability and customized mutation method are used for mutation operation. Firstly, the chromosome of Pp is selected from P with a probability of pm. Each chromosome of Pp mutates with a probability of p_m. Secondly, The mutation probability p_m composed of two parts, one part is determined by the non-dominated level of the chromosome, i.e. according to the change process of the IAGA according to the level, as shown in the improved adaptive mutate operator (IA-MP) point in Fig.10.

The other part is related to the algebra of evolution, Jiao Hong Yi et al. verified the validity of this change [46]. Therefore the probability of mutation is as follows:

$$p_m = \left(-\frac{1}{1+\exp\left(2Al/_N\right)} + 1\right) \cdot \left(p + (t-1) \times \frac{1-p}{t_m - 1}\right) \quad (3)$$

where A is a constant equal to 9.903438, l is the non-dominated level of the selected chromosome and N is the largest non-dominated level. The first half shows that the mutation probability of each level of chromosomes obeys the *sigmoid* function. In the second half, t_m is the largest evolution algebra, t is the current algebra, and p is a fixed value, such as 1/50. That is, it is related to the algebra of the iteration, and the mutation is accelerated later.

The task execution sequence obtained by the decomposition is constrained by the temporal logic relationship, and the chromosomes in this paper are encoded in the form of task

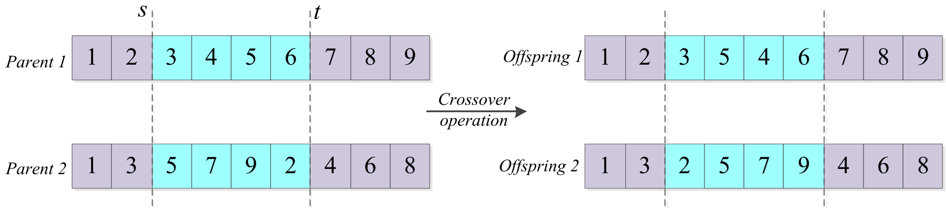

FIGURE 9. Example of crossover operation.

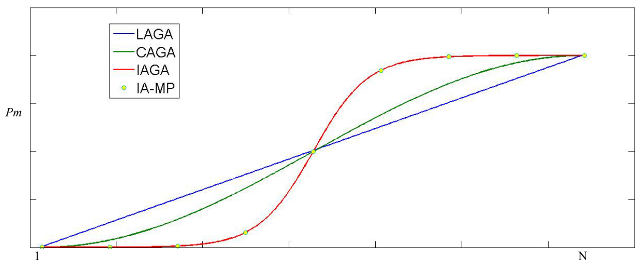

FIGURE 10. Probability graph of Level-Dependent mutation operation.

order. Therefore, the constraints are transformed into immune vaccines, i.e. new chromosomes are obtained by immune selection.

The specific steps are:

1. Select the previous task set *Ai* and the subsequent task set *Bi* for each task *i* by temporal logic.

2. Randomly select a point *j* from the selected chromosome *P*, and traverse the points other than *Aj* in *P*.

3. Determine in turn whether or not to exchange with *j*. such as selecting point *k* is selected, that is, after *j* and *k* are exchanged, whether *Aj* and *Bj* are still before and behind of *j*, and whether *Ak* and *Bk* are still before and behind of *k*. (vaccination process).

4. Get the set *Cj* that can be exchanged, and randomly select the points in *Cj* to exchange with *j*.

As shown in Fig.11, the order of tasks is T2-T3-T5-T6, T5-T7-T9, T1-T4-T8-T10. Assuming that T4 is randomly selected, and *A4* and *B4* are {T1} and {T8, T10}, traversing elements other than *A4* and *B4*, that is, T2, T3, T5, T6, T7 and T9 are sequentially judged. If T2 is selected, T2 cannot be exchanged because it is constrained by T2-T3. After traversing, *C4* is {T3, T5}, and T5 is randomly selected.

D. OBJECTIVE FUNCTION

After the mission is decomposed based on JMT, the function chain required for sub-task execution and the basic time for processing the task can be obtained. On the premise that the task order satisfies the constraints, different priority order will due to resource satisfaction (mainly reusable UAVs) result in waiting time, and because of link reuse or reconstruction result in different connection consumption. In this paper, the task's execution order is optimized by the total completion time of the task, the connection consumption caused by the link reconfiguration, and the real-time available UAVs.

1) TOTAL TIME THE MISSION IS COMPLETED

The mission completion time is the moment when the last action is completed.For a single action, the time spent can be divided into the time of execution and the time of preparation. The time of the preparation includes waiting time due to insufficient resources (UAVs), the construction time of the function chain and the collaborative time of multiple chains participation.

The construction time of function chains required for operational action is the maximum construction time in the function chain required for the action. The time of construction of the function chains L1 to L6(II) is different when the function chain of previous set of tasks can be satisfied or cannot be satisfied. If it can be satisfied, the construction time is (0,0,0,0,3,1,1,3,1,1,4,0) respectively, if not satisfied, it is (0,0,0,0,3,3,3,4,4,4,4,4).

In the network described by FINCA^2, the collaborative time of multiple chains participate includes two aspects, one is the delay of nodes for processing information, and the other is the delay of channels on the information channel. Therefore, when multiple chains participate, it is first necessary to determine the used operation chain, and calculate the collaborative time between each two chains, that is, give a time collaborative matrix. When multiple chains participate, only the maximum collaborative time in the collaborative matrix needs to be selected.

2) CONNECTION CONSUMPTION

In the FINCA^2 network, the function chain can be reorganized and reconstructed, and in this process, the connection consumption will be generated. In the actual use process, the subsequent tasks will be used as much as possible the reusable link part of the pre-order task to reduce the reconnection. When F and A nodes do not perform tasks, they always maintain the connection with their superior nodes. The I and C nodes always maintain the connection with the superior node, that is, when they establish connections with other nodes, the connection with their superior nodes will not be disconnected. The function chains L1, L2, L3(I), and L3(II) are simple connection structures of the I, F, and A nodes with their superior nodes, so the change in the number of uses does not result in the consumption of reconnection. For the first task performed, the connection change amount, i.e. the connection consumption is:

$$n_c = l_i \cdot (0, 0, 0, 0, 3, 3, 3, 4, 4, 4, 4, 4)^T \tag{4}$$

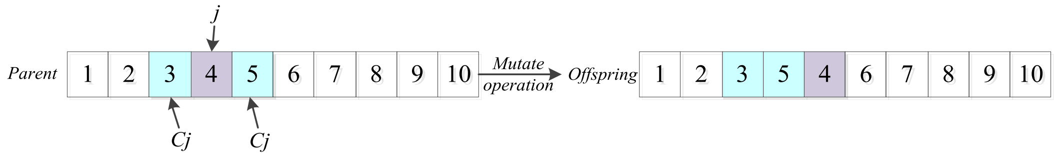

FIGURE 11. **Schematic graph of the mutation operation.**

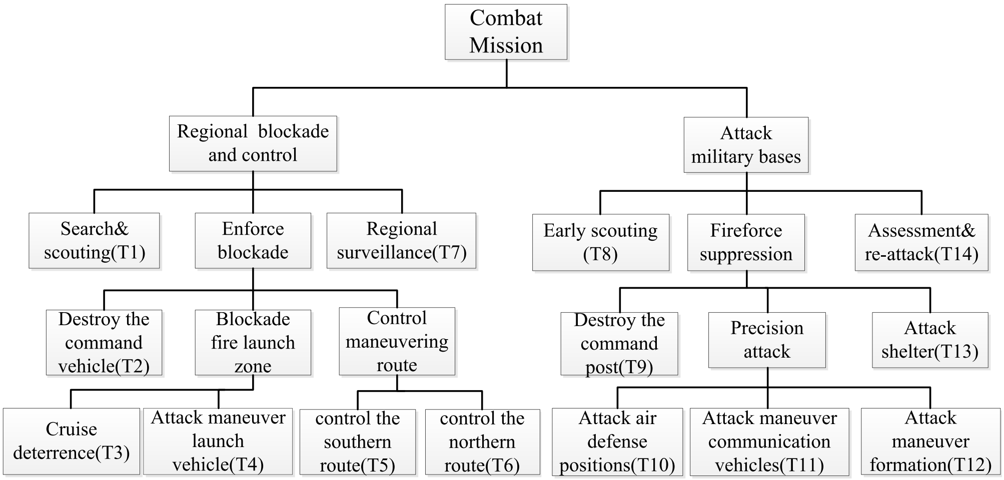

FIGURE 12. **Task decomposition graph.**

where, l_i is the function chain requirement vector of the initial task. For the connection consumption of subsequent tasks, it is divided into two situations: when the task is completed, because of the resources such as the UAVs are not satisfied, no new task starts, and when the task is completed, the resource can be satisfied, there are new task starts. The first type, when there is a task completion, no task starts, the node will be disconnected from the temporarily formed link, and connected with the superior C node, the connection consumption is:

$$n_d(t_j) = \boldsymbol{l}(t_j) \cdot (0, 0, 0, 0, 0, 2, 2, 1, 3, 3, 0, 4)^T \tag{5}$$

where, t_j is the time when there is no task to start after there is task completed, $l(t_j)$ is the total function chain requirement of the tasks completed at the time t_j, and $n_d(t_j)$ is the connection consumption generated at the time t_j. The second type is that when a task is completed, some new task starts. At this point, need to calculate whether there are some existing connections that can be reused. The connection consumption is calculated as:

$$\begin{aligned} n_r(t_k) = & (\boldsymbol{l_s}(t_k) - \boldsymbol{r}(t_k)) \cdot (0, 0, 0, 0, 3, 3, 3, 4, 4, 4, 4, 4)^T \\ & + (\boldsymbol{l_e}(t_k) - \boldsymbol{r}(t_k)) \cdot (0, 0, 0, 0, 0, 2, 2, 1, 3, 3, 0, 4)^T \\ & + r(t_k) \cdot (0, 0, 0, 0, 3, 1, 1, 3, 1, 1, 4, 0)^T \end{aligned} \tag{6}$$

Among them, $r(t_k) = \frac{|\boldsymbol{l_s}(t_k)+\boldsymbol{l_e}(t_k)|-|\boldsymbol{l_s}(t_k)-\boldsymbol{l_e}(t_k)|}{2}$ is the function chain vector that can be reused for the previous and subsequent tasks, t_k is the time at which the task starts after there is task completed, $l_s(t_k)$ is the function chain vector required for the task starting at time t_k, and $l_e(t_k)$ is the function chain vector used by the task completed at time t_k, $n_r(t_k)$ is the connection consumption generated at time t_k. It can be seen that the connection consumption n is:

$$n = \sum_{t_j \in t_e \setminus t_s} n_d(t_j) + \sum_{t_k \in t_e \cap t_s} n_r(t_k) + n_c \tag{7}$$

Here, t_s is a set of times when all the tasks start, and t_e is a set of times when there is task completed.

3) AVERAGE USAGE OF UAVS

The A nodes in the function chains L3(II), L4(II), L4(III), L5(II), L5(III) and L6(II) may all be UAVs, while UAVs, I nodes, and F nodes are different with the A-nodes such as the cruise missile, it can be re-used. During the execution of task, the more UAVs that are not involved in the task, the more function chains available for other tasks, the more the task can be done successfully.

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

$$s(i) = \sum_{\substack{t_s(k) \le T(i) \\ t_e(k) \ge T(i+1)}} l_k \cdot (0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 3)^T \tag{8}$$

where, l_k is the function chain vector used by the task being executed from time $T(i)$ to $T(i+1)$. The average number of

TABLE 2. Function chain time coordination matrix.

Time coordination	L1	L2	L3 (I)	L3 (II)	L4 (I)	L4 (II)	L4 (III)	L5 (I)	L5 (II)	L5 (III)	L6 (I)	L6 (II)
L1	3	5	5	5	6	6	6	6	6	6	7	7
L2	5	3	5	5	4	6	4	4	4	6	7	7
L3(I)	5	5	3	5	6	4	6	6	6	6	5	7
L3(II)	5	5	5	3	6	4	6	6	6	4	7	5
L4(I)	6	4	6	6	4	6	4	4	4	6	7	7
L4(II)	6	6	4	4	6	4	6	6	6	4	7	5
L4(III)	6	4	6	6	4	6	4	4	4	6	7	7
L5(I)	6	4	6	6	4	6	4	4	4	6	7	7
L5(II)	6	4	6	6	4	6	4	4	4	6	7	7
L5(III)	6	6	6	4	6	4	6	6	6	4	7	5
L6(I)	7	7	5	7	7	7	7	7	7	7	5	7
L6(II)	7	7	7	5	7	5	7	7	7	5	7	5

TABLE 3. Function chain and time needs list.

Requirement list	L1	L2	L3 (I)	L3 (II)	L4 (I)	L4 (II)	L4 (III)	L5 (I)	L5 (II)	L5 (III)	L6 (I)	L6 (II)	Time /min
T1	1	0	0	3	0	2	1	0	0	0	0	0	6
T2	0	2	2	0	0	1	1	0	1	0	0	0	7
T3	0	0	0	0	0	0	0	0	0	0	1	1	12
T4	0	0	2	2	1	1	2	0	0	0	0	0	4
T5	1	0	0	0	0	0	0	0	0	0	0	2	12
T6	1	0	0	0	0	0	0	0	0	0	0	2	12
T7	1	0	0	2	0	0	0	0	0	0	0	0	6
T8	1	0	0	3	0	0	0	0	0	0	0	0	5
T9	0	2	1	0	0	0	2	2	0	1	1	0	7
T10	0	2	0	0	1	2	0	2	0	0	1	0	5
T11	0	0	2	0	0	2	0	0	0	0	0	0	4
T12	1	0	2	1	1	1	1	0	0	0	1	0	8
T13	0	0	2	0	0	3	1	0	1	2	0	1	10
T14	1	0	0	2	0	0	0	1	2	2	0	0	6

UAVs used is:

$$s = \frac{1}{n(T)} \sum_{i=1}^{n(T)} s(i) \tag{9}$$

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

VI. CASE STUDY

Assume that there is a need to complete a border combat mission. Due to the terrain is complex and facilities is concealed, and the troops can go to the front line are restricted. So, the target is difficult to find, and the terrain has an impact on communication. Hybrid manned and unmanned combat SoS has become the dominant force to break down the constraints. The mission consists of two major parts. One is the elimination of the force and the blockade the area where exist dispute between the enemy and US; the second is to destroy the enemy's military base located near the dispute area. The hybrid combat SoS has 25 UAVs, the common missile(F) and cruise missile(A) are 30 and 50 rounds respectively. The mission are decompose into 14 subtasks as shown in Fig.12 below.

Among the 14 subtasks in the above figure, some subtasks have a priority relationship. The priority relationships include T1-T2-T4, T7-T5-T6, T8-T9-T10-T12, which include strict order relationships T1-T2, T8-T9. According to the FINCA^2 structure of the combat SoS, in order to simplify the calculation, we set the time delay of each node and edge processing information to 1 unit, and calculate the time collaborative matrix between the function chains, as shown in Table 2.

After the decomposition process in Section 3, the function chain requirement matrix of the task can be obtained as follows. For the I-node, the satellite requirements, the number 1 only represents the need to use, not the specific number.

Under the premise of satisfying the above three constraint relationships, there are still $C_{14}^{3} \cdot C_{11}^{3} \cdot C_{8}^{4} \cdot 4! = 100900800$

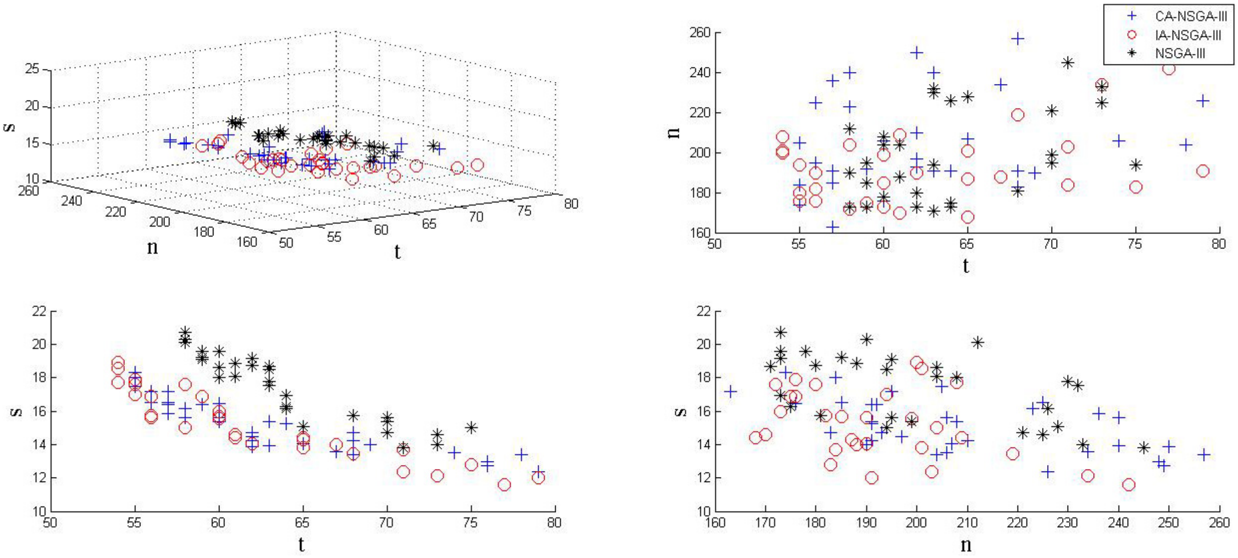

FIGURE 13. Comparative result.

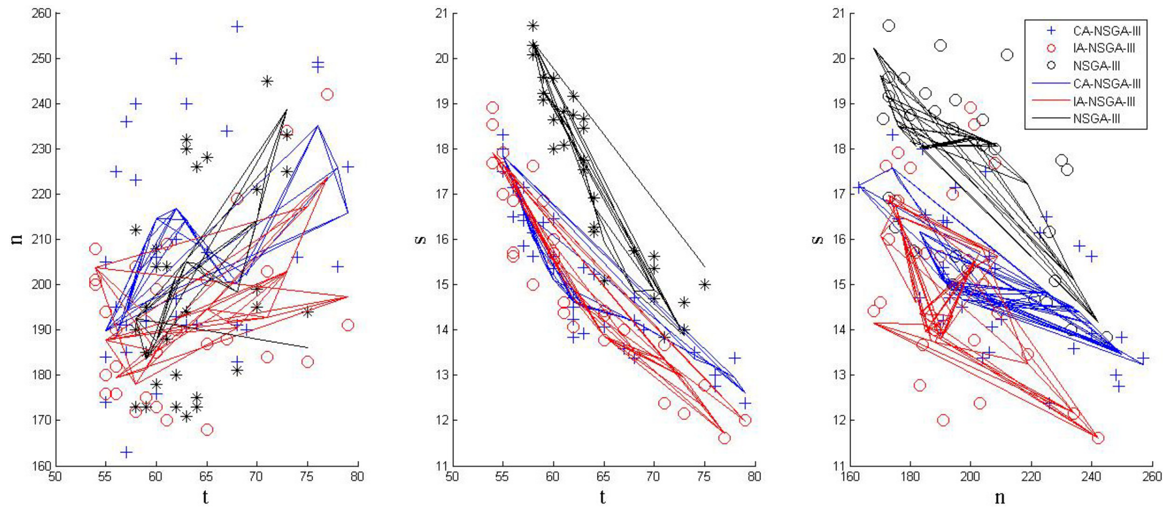

FIGURE 14. Comparative curves fitting result.

schemes that can meet the order requirements of the tasks. For the above three objective functions, it is unrealistic to choose a better scheme through traversal. Select NSGA-III as the basic algorithm, and perform the cosine improvement and adapitive improvement introduced above for the crossover and mutation operations. The three algorithms perform 50 iterations on Matlab to verify the advanced of the improved algorithm. Fig.13 and 14 below are comparative results, in which the black star is the result of the original algorithm (NSGA-III), the blue cross is the result of the cosine improvement (CA-NSGA-III), and the red circle is the result of the adaptive improvement (IA-NSGA-III). The upper left corner of Figure 13 is a three-dimensional result graph of three objective functions, and the other are two-dimensional contrast graphs of two targets. It can be seen that all three methods can obtain a set of non-dominated solutions. It can be seen from the bottom left corner of Fig.13 that there is a significant negative correlation between the time of completion of the tasks and the number of UAVs used in the three methods. It can be concluded that the more resources can reused, the more helps the task to complete quickly. The three figures in Fig.14 are two-dimensional results of two targets and curve fitting of the results. As can be seen from the three figures, the blue curve is relatively below the black curve and the red curve is at the bottom. Therefore, the cosine improvement method is superior to the original method, and the improved adaptive algorithm is superior to the cosine improvement method. As can be seen from the final graph of Fig.14, the three groups of curves are more clearly separated from each other. In other words, the improved method has a more significant effect on the calculation results of time and reused resources. It can be seen from the comparison of the results of the three methods that the adaptive improvement method can obtain more and better Pareto frontier solutions, the cosine improvement method is second, and the original method is next to the cosine improvement method.

VII. CONCLUSION

In view of the characteristics of the new intelligent SoS, this paper extends the FINC model to the new intelligent C2 network model FINCA^2, which meets the requirements of the connection between nodes and nodes of the combat SoS. Based on this, a variety of function chains with simple structure and frequent use are extracted to reduce the consumption caused by connection switching and communication interference.

Based on this, JMT decomposition is used to obtain task orders with temporal logic and function chain mapping. According to the characteristics of the problem, the NSGA-III algorithm, which has better convergence and guarantees the diversity of population, is customized and improved to optimize the operational order to obtain the required operational plan. We compare the linear crossover and mutation process, the cosine improved process and the adaptive improved process, Validation of the effectiveness of the method with operational cases and the adaptive improvement algorithm can be used to obtain more frontier solutions.

Our original intention is to provide a set of methods to generate and optimize the operational plan for the new combat SoS. For other combat scenarios, such as anti-submarine warfare, air combat, etc., the equipment types are different, the network descriptions may need to be properly improved, and the number and type of function chains also need to change accordingly. But the method and the process of solving the whole operational plan are common.

In our future work, on the one hand, we need to compare with more multi-objective algorithms and explore different methods to improve the algorithm. On the other hand, the description of the combat network can be expanded more abundantly.

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] A. Boukhtouta, A. Bedrouni, J. Berger, F. Bouak, and A. Guitouni, "A survey of military planning systems," in *Proc. 9th ICCRTS Int. Command Control Res. Technol. Symp.*, 2004, pp. 5–7.

[3] J. R. Cares and J. Q. Dickmann Jr., Eds. *Operations Research for Unmanned Systems*. New York, NY, USA: Wiley, 2016.

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

[5] B. Srivastava and S. Kambhampati, "Scaling up planning by teasing out resource scheduling," in *Proc. Eur. Conf. Planning*, vol. 1809. Berlin, Germany: Springer, 1999, pp. 172–186.

[6] J. Boyd, *A Discourse on Winning and Losing*. Montgomery, AL, USA: Air Univ. Press, 2018.

[7] R. Fogleman, "Targeting—Air force doctrine document 2-1.9," Air Force Doctrine Center, Maxwell Air Force Base, Montgomery, AL, USA, Tech. Rep. USAF-AFDD-2-1.9, 2006.

[8] C. Behre, "DoD enterprise architecture conference applied joint mission thread," USJFCOM, Norfolk, VA, USA, Tech. Rep.

[9] 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, Apr. 2013.

[10] R. Johnson, "Dynamic complexity in system of systems," Internal Rep. Eng. Technol. Adv. Syst., Boeing Company, Chicago, IL, USA, Tech. Rep., 2007.

[11] M. Zhang, L. Ma, and M. Ji, "Networked system-of-systems combat OODA command circulation time measuring modeling and experiment," *J. Command Control*, vol. 1, no. 1, pp. 50–55, 2015.

[12] P. Di, T. Hu, B. Hu, and J. Zhang, "Research on invulnerability of combat net model based on complex networks," *J. Syst. Simul.*, vol. 23, no. 1, pp. 56–60, 2011.

[13] Y. Qi, Z. Liu, H. Chen, and J. Han, "Distributed combat system of systems network modeling," *J. Netw.*, vol. 8, no. 8, pp. 1906–1912, 2013.

[14] J. R. Cares and J. Q. Dickmann, *Operations Research for Unmanned Systems*. New York, NY, USA: Wiley, 2016.

[15] J. R. Cares, "An information age combat model," Alidade, Inc., Newport, PR, USA, Tech. Rep. TPD-01-C-003, 2004.

[16] S. B. Robinson, Y. Li, J. N. Nixon, and D. N. Mavris, "Model development of large-scale DoD system-of-systems," unpublished.

[17] A. Mahmoud *et al.*, "Modeling community containment for pandemic influenza: A letter report," Committee Model. Community Containment Pandemic Influenza, Inst. Med. (HDM), Charlottesville, VA, USA, Lett. Rep., 2006.

[18] A. H. Dekker, "Measuring the agility of networked military forces," *J. Battlefeild Technol.*, vol. 9, no. 1, pp. 19–24, 2006.

[19] G. Yang, W. Zhang, B. Xiu, Z. Liu, and J. Huang, "Key potential-oriented criticality analysis for complex military organization based on FINC-E model," *Comput, Comput. Math. Org. Theory*, vol. 20, no. 3, pp. 278–301, 2014.

[20] Z.-J. Du, "Modeling and resolving method of course of action planning under antagonism," (in Chinese), Ph.D. dissertation, Nat. Univ. Defense Technol., Changsha, China, 2012.

[21] L. Falzon, "Using Bayesian network analysis to support centre of gravity analysis in military planning," *Eur. J. Oper. Res.*, vol. 170, no. 2, pp. 629–643, 2006.

[22] R. J. Elder, Jr., "Effects-based operations: A command philosophy," *Air Space Power J.*, vol. 21, no. 1, p. 10, 2007.

[23] L. W. Wagenhals and A. H. Levis, "Modeling support of effects-based operations in war games," in *Proc. Command Control Res. Technol. Symp.*, 2002.

[24] P. W. Pachowicz, L. W. Wagenhals, A. H. Levis, and J. Pham, "Building and analyzing timed influence net models with Internet-enabled pythia," *Proc. SPIE*, vol. 6564, May 2007, Art. no. 65640C.

[25] J. F. Lemmer and D. E. Gossink, "Recursive noisy OR—A rule for estimating complex probabilistic interactions," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 34, no. 6, pp. 2252–2261, Dec. 2004.

[26] J. A. Rosen and W. L. Smith, "*Influence net* modeling with causal strengths: An evolutionary approach," in *Proc. Command Control Res. Technol. Symp.*, vol. 1996, pp. 25–28.

[27] 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, 2002.

[28] 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.

[29] Y. Lee and T. Lee, "Network-based metric for measuring combat effectiveness," *Defence Sci. J.*, vol. 64, no. 2, pp. 115–122, 2014.

[30] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[31] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multi objective optimization," in *Proc. Evol. Methods Design Optim. Control Appl. Ind. Problems*, vol. 103, 2001, pp. 95–100.

[32] 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.

[33] 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.

[34] C. M. Fonseca and P. J. Fleming, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 28, no. 1, pp. 26–37, Jan. 1998.

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

[36] 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.

[37] A. Ibrahim, S. Rahnamayan, K. Deb, and M. V. Martin, "EliteNSGA-III: An improved evolutionary many-objective optimization algorithm," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 973–982.

[38] X. Bi and C. Wang, "An improved NSGA-III algorithm based on elimination operator for many-objective optimization," *Memetic Comput.*, vol. 9, no. 4, pp. 361–383, Dec. 2017.

[39] C. Chen, Y. Yuan, and X. Yuan, "An improved NSGA-III algorithm for reservoir flood control operation," *Water Resour. Manage.*, vol. 31, no. 14, pp. 4469–4483, 2017.

[40] H. Majeed and C. Ryan, "A less destructive, context-aware crossover operator for GP," *Genetic Programming* (Lecture Notes in Computer Science), vol. 3905. Berlin, Germany: Springer-Verlag, 2006, pp. 36–48.

[41] X. Yao and Y. Xu, "Recent advances in evolutionary computation," *J. Comput. Sci. Technol.*, vol. 21, no. 1, pp. 1–18, 2006.

[42] L. Jiao and L. Wang, "A novel genetic algorithm based on immunity," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 30, no. 5, pp. 552–561, Sep. 2000.

[43] S. Shi, Q. F. Li, and X. H. Wang, "Design optimization of brushless direct current motor based on adaptive genetic algorithm," (in Chinese), *J. Xi'an Jiaotong Univ.*, vol. 36, no. 12, pp. 1215–1218, 2002.

[44] H. Y. Kuang, J. Jin, and Y. Su, "Improving crossover and mutation for adaptive genetic algorithm," *Comput. Eng. Appl.*, vol. 42, no. 12, pp. 93–96, 2006.

[45] A. Menon, K. Mehrotra, C. K. Mohan, S. Ranka, "Characterization of a class of sigmoid functions with applications to neural networks," *Neural Netw.*, vol. 9, pp. 819–835, Jul. 1996.

[46] 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 Gener. Comput. Syst.*, vol. 88, pp. 571–585, Nov. 2018.