Dynamic Multi-objective AWPSO in DT-Assisted UAV Cooperative Task Assignment

Min Deng, Zhiqiang Yao, *Senior Member, IEEE*, Xingwang Li, *Senior Member, IEEE*, Han Wang, *Senior Member, IEEE*, Arumugam Nallanathan, *Fellow, IEEE*, and Zeyang Zhang

***Abstract*—In recent years, more and more attention has been paid to the unmanned aerial vehicle (UAV) cooperative task assignment. In order to complete the task with the lowest cost, some researchers use multi-objective optimization to solve the assignment problem. But few of them consider the complex dynamic scenarios. In this article, the time-varying resource supply and demands are provided by established digital twins (DTs) of UAVs and targets, thereby enabling accurate decision guidance for dynamic task assignment. It takes the scheduling cost, path cost, risk cost and total task time cost as the optimization objectives. To solve this model, an improved dynamic multi-objective adaptive weighted particle swarm Optimization algorithm (DMOAWPSO) is proposed. In the initialization stage, a heuristic method is used to increase the effectiveness of the solution. Besides, the adaptive mutation and subgroup methods are adopted to improve the diversity of the solution. Then, effective environment change detection and response strategies are designed to adapt to dynamic scenarios. Finally, the evaluation metrics are calculated in different instances. Compared with the popular and classic dynamic multi-objective algorithms, the simulation results verify that the proposed algorithm is effective and can cope with the environment changes better in solving the task assignment problem.**

***Index Terms*—dynamic multi-objective optimization, UAV, digital twin, task assignment, evolutionary algorithm**

I. Introduction

UNMANNED aerial vehicle(UAV) plays an increasingly important role in various fields, especially in the military field. For example, situational reconnaissance [1] and target surveillance [2] can be completed by UAVs, owing to their unique advantages of massive connections, high spectral efficiency and flexibility [3]. However, with the complexity of the mission environment and requirements increasing, a single UAV with limited capabilities often can not meet the mission requirements [4]. Therefore, multiple UAVs are often formed as a team to cooperate utilizing the autonomous control technology. In this case, the data interaction between UAVs and command center can be excuted by adopting reliable and efficient wireless communication network technology, such as device-to-device (D2D) technology, to enhance the transmission capacity, reduce the transmission power, and improve the spectrum utilization [5], [6]. UAV collaborative task assignment is to assign different subtasks and the corresponding sequences to each UAV to meet the mission requirements within their capabilities. In this way, some heterogeneous UAVs can be assigned to perform multiple tasks against certain given targets with limited time. The aim of the assigning problem is to achieve optimal efficiency by taking the minimum system cost [7].

Different from the traditional formation control, the task assignment of UAV is a complex combinatorial optimization problem, which needs to consider the task priority, coordination, time constraint and flight trajectory. In the actual environment, it is difficult to describe the problem properly, especially for the complex dynamic environment, such as changes in the number and position of UAVs and targets, or changes in their own parameters. Essentially, the task assignment problem is also a nondeterministic polynomial hard (NP-hard) problem [8]. So the size of the problem (the number of UAVs, targets and tasks) affects the complexity of the solution process.

Most of the research on the problem of UAV cooperative task assignment is limited in static environment, which often ignores the influence of the position change, scale change and parameter change of both resources and targets. In addition, due to the complexity of UAV flight path and unpredictability of flight target in reality, the existing UAV field situation is mapped to the virtual space for analysis, namely digital twin (DT) technology [9]. DT provides an excellent solution for intelligent resource assignment in UAV by creating a real-time digital simulation model of physical entities, which is of great significance to the intelligent development of UAV system [10].

Since the problem scale affects the complexity of the solution process, this paper mainly focuses on the case of small scale and typical constraints. Considering of the dynamic changes, a dynamic multi-objective optimization model is established with constraints of resource demand, time, path and risk coefficient. Meanwhile, this paper proposes to use DT to reflect the dynamic characteristics of the resource demand and supply of the UAV and target from physical entities. The objective is to find the optimal target assignment scheme with minimum scheduling cost, total time cost, path cost and risk

This work is funded by the National Key R&D Program of China with 2020YFA0713502, and the National Natural Science Foundation of China (Grant No.12071398), and the Natural Science Foundation of Hunan Province, China (Grant No.2019JJ50620). *(Corresponding author: Xingwang Li and Han Wang.)*

M. Deng, Z. Yao and Z. Zhang are with the School of Automatic and Electronic Information, Xiangtan University, Hunan National Center for Applied Mathematics, Key Laboratory of Intelligent Computing and Information Processing of Ministry of Education, Xiangtan, 411105, China (e-mail: iemdeng@xtu.edu.cn, yaozhiqiang@xtu.edu.cn, blackcatovo@qq.com). X. Li is with the School of Physics and Electronic Information Engineering, Henan Polytechnic University, Jiaozuo 454000, China (e-mail: lixingwangbupt@gmail.com). H. Wang is with the College of Physical Science and Engineering, Yichun University, Yichun, 336000, China (e-mail: hanwang1214@126.com). A. Nallanathan is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K. (e-mail: a.nallanathan@qmul.ac.uk)

cost for UAVs. Then, an improved dynamic multi-objective adaptive weighting particle swarm optimization (DMOAW-PSO) algorithm is proposed to solve the model. In this algorithm, the heuristic initialization, adaptive mutation and subgroup methods are used to improve the effectiveness of the solution. Moreover, improved environment change detection and response strategy are designed to deal with the dynamic change.

The rest of this paper is scheduled as follows. Section II introduces the previous related work. Section III describes the system model. Section IV presents the improved algorithm. Section V provides simulation results and analyzes the algorithm performance. Section VI concludes this paper with discussions on future work.

II. Related Works

A. Scenarios and Model

There have been several relatively in-depth studies on the task assignment of UAVs. How to assign resources and tasks needs to consider different optimization goals and corresponding constraints in different scenarios. The research of static environment has been very mature. Cui *et al.* [11] decomposed complex tasks into sub-tasks suitable for a single UAV. Then they described the task assignment problem from the aspects of task reward, terrain cost and loss cost. The constraints concluded heterogeneous UAV load constraints and mission cost constraints. Ye *et al.* [12] mainly studied the cooperative jamming decision-making of UAVs in complex electromagnetic environments. The model evaluated the jamming effects by considering benefits of power suppression, frequency alignment, jamming coverage space and jamming pattern, and costs of the probability cost of resource discovery. For the study of dynamic environment, the task reassignment of UAV in a time sensitive and dynamic environment was studied in [13]. Different time windows of new tasks, different locations of new tasks, the continuous emergence of new tasks and the continuous emergence of search and rescue scenarios of different scales were considered from four perspectives. Peng *et al.* [14] aimed to improve the problems of poor real-time performance and low solution quality of dynamic task assignment of heterogeneous UAV groups in uncertain environment. It designed a dynamic task assignment model based on multi-objective, multi task, heterogeneous multi machine platform. It constrained the target cost performance and task execution time window. Most of the current research relies on certain simple hypothesis, lacking real reflection of dynamic environment changes. The distribution relationships are limited to one-to-one or one-to-many distribution. Most of them use the evaluation factor method when establishing the object of the model, so it is difficult to unify the dimensions.

DT provides an excellent solution for intelligent resource allocation in task assignment by creating a real-time digital simulation model of physical entities. In [10], DT of task assignment is established to capture the time-varying resource supply and demands, so that unified resource scheduling and assignment can be performed. An architecture of Digital Twin-driven Smart ShopFloor (DTSF) to realize dynamic resource allocation optimization (DRAO) is proposed in [15].Schroeder *et al.* [16] proposed the use of AutomationML to exchange data between the digital twin and other systems in the future manufacturing. Jee *et al.* [17] proposed the architecture of Digital Twin-driven Smart ShopFloor (DTSF) to realize dynamic resource allocation optimization (DRAO). Digital twin can be applied to the maintenance, diagnosis and prognostics of products or equipments. Therefore, the model architecture of this paper also adds the part of digital twin driver.

As a new concept, DT is expected to be widely used in many fields. In terms of vertical security problem of maglev train [18], DT can help reflect the unknown external disturbances to ensure the vertical security. In terms of the next-generation communication, although the existing several technological advances are expected to help the forecasted demand on the 5G environment, these technologies have not been fully tested to benchmark performance [19]. There are many challenges that need to be resolved. A 3-hop non-orthogonal multiple access (NOMA) UAV-aided green communication network framework was proposed in [20], in which DT can be adopted to provide the realistic imperfect successive interference cancelation evaluation instead of assumption. Similarly, DT will be possibly applied in the modeling of cellular users, D2D transmitters, and relay nodes [21] in D2D communication and the channel model and estimation in reconfigurable intelligent surface-assisted orthogonal frequency division multiplexing (OFDM) wireless communications [22].

B. Algorithm Method

As an NP-hard problem, no polynomial time algorithm has been found to solve the task assignment model of UAVs at this stage. However, some researchers have tried to solve it for small and medium scales. Recent solution methods can be divided into two categories: traditional mathematical methods and intelligent optimization methods. Traditional mathematical methods includle constraint programming method [23], mixed linear programming method [24], graph theory method [25] and so on. These methods can obtain the optimal solution of the proposed problem with high computational complexity but long computing time. They are ineffective in solving medium-scale and high-complexity problems. While intelligent optimization algorithms, such as genetic algorithm [26], particle swarm algorithm [27], simulated annealing method [28], game theory [29], can greatly reduce the amount of calculation through iterative evolution and other forms. The optimal or sub-optimal solution can be obtained in an acceptable time period. But the solution space will increase exponentially with the increase of the scale and optimization objectives. However, the modern intelligent optimization algorithms have been relatively perfect to adapt to specific problems to improve the solution performance. So they have become the mainstream method of UAV cooperative task assignment. Besides, the deep belief network was utilized to learn the control system in [30]. Such deep learning methods can be applied to learn and deal with the uncertainties encountered in practice effectively.

For intelligent algorithms to solve the UAV task assignment problem, many researchers use weighted summation to transform multiple optimization objectives into a single objective

optimization problem, which simplifies the problem and makes it easier to calculate. However, single objective optimization has the following drawbacks: (1) It is difficult to determine the allocation of weights. Generally, there must be some contradictions between different indicator functions. It is difficult to balance these indicators by choosing appropriate weights; (2) The stability of the problem is affected. The dimensions of each evaluation indicator are often inconsistent. So the superposition of indicators with different orders of magnitude can deteriorate the stability of the model; (3) The final solution of single objective optimization contains less information. Due to the superposition of each indicator, we can only obtain a unique single objective optimization solution. Besides, we cannot see the advantages and disadvantages of different indicators from the solution.

In recent years, some researchers have begun to use multi-objective optimization algorithms to solve such problems. Wang *et al.* [31] constructed a mathematical model of heterogeneous UAVs and time-series coupled tasks. Through the proposed improved multi-objective quantum behavioral particle swarm optimization algorithm, four objects of completion time, target reward, UAV damage and total range were optimized. Atencia *et al.* [32] proposed a new weighted stochastic multi-objective evolutionary algorithm to solve the UAV mission planning problem. Seven optimization objectives were optimized. The convergence speed of the proposed algorithm was verified through multiple experiments. Deb *et al.* [33] suggested a nondominated sorting genetic algorithm II (NSGA-II), which can maintain a better spread of solutions and converge better in the obtained nondominated front.

However, the above researches only consider the application of multi-objective algorithms in static environments, without considering the selection of algorithms in dynamic environments. Sun *et al.* [34] proposed an improved multi-objective ant lion optimization (IMOALO) algorithm to solve the UAV network communication problems. The algorithm used chaos-opposition based learning solution initialization and hybrid solution update operators to solve the problem. It has better performance than some other benchmark approaches. Yu *et al.* [35] proposed an improved multi-objective optimization algorithm to solve the resource allocation problem in radio access technologies (RATs). The algorithm used the weighted Tchebycheff method to optimize the problem, and numerical results demonstrate that the proposed algorithm yields fast convergence, high system energy efficiency, and flexible energy efficiency tradeoff.

The key to dynamic multi-objective algorithms is the environmental change detection and response. The main requirement of environmental change detection is to quickly and accurately detect the changes of environmental parameters at the current moment compared to the previous moment. The current methods include re-evaluation [36], distribution estimation of objective function value data [37], steady state detection method [38] and so on. For these detection methods, the first method selects a small part of the population individuals in two environments before and after the change. Its detection accuracy cannot be well guaranteed. The latter two methods have high computational costs. And the detection accuracy is not stable enough. The existing environmental change response mechanisms include diversity introduction mechanism [36], diversity maintenance mechanism [39], prediction mechanism [40], memory mechanism [41] and so on. For different types of environmental changes, choosing the appropriate response mechanism can better improve the quality of the solution.

As discussed above, compared with the recent works, this paper will study the assignment problem in dynamic multi-objective scenarios. Dynamic DTs will be used to collect real-time changes of UAVs and targets parameters. At the same time, more real and complex many-to-many distribution relationship will be considered. Thus, this paper establishes a multi-objective optimization model to avoid the influence of dimension difference and weight selection on optimization objectives. Then an improved DMOAWPSO algorithm is proposed. The main improvements compared to the adaptive weighting particle swarm optimization (AWPSO) algorithm [42] are as follows:

1) A heuristic population initialization method is adopted to better cover the solution space and provide better initial solution to meet various constraints. It can guide the subsequent population evolution and speed up the convergence.
2) An adaptive mutation method is proposed, which contains three different mutation operators in total. It can help the population find a better solution with a greater mutation probability when the environment changes.
3) Multi-subgroup optimization is used to increase the diversity of the optimal solution set. The number of different subgroups is allocated according to the number of objective functions.
4) An improved environmental change detection and environmental change response method is proposed to realize dynamic multi-objective scheduling optimization. The DTs of UAVs and targets can provide the real time parameters by connecting the virtual simulation model and physical enties. It detects and classifies the change according to the measurement of the change of environment and the relative movement of the center of mass. Then, different response methods are adopted under corresponding change categories, so that the population can maintain better diversity and convergence in the new environment.

III. System Model

A. Scene Construction

This paper considers a three-dimensional space environment where UAVs and targets are distributed in two areas at a certain distance initially. Assume that there are M UAVs resources $(u_i, i = 1, ..., M)$ and N targets $(t_j, j = 1, ..., N)$. The parameter information of the targets is assumed to have been obtained by satellite reconnaissance. Among the resources, the foregoing m_1 UAVs can use their carried jamming weapon types k_1 and k_2 to jam the targets. While the remaining $m_2 = M - m_1$ UAVs can use their carried attack weapon types k_3 and k_4 to attack the targets. Each UAV can perform multiple tasks in sequence. So some of the UAVs will be in

the waiting stage or unassigned with tasks. But each target must be assigned with at least one jamming weapon resource and one attack weapon resource. Each target has different requirements for weapon resources. Thus, the corresponding assignment relationship between the resources and targets should be many-to-many. When all targets are attacked, the total task is completed. Table I and Table II show the detailed attributes of UAV and target [31].

TABLE I UAV ATTRIBUTE PARAMETER

Attribute	*Parameter*
UAV number	N^U
Position	X_i^U
Detection radius	R_i^{detect}
Capability type	P_i
Speed	v_i
Weapon resources carried	L_i^k
Maximum flight duration	T_i^{max}
Resource value	V_i^U
Maximum range	dis_i^{max}
Fuel quantity	$fuel_i$

TABLE II TARGET ATTRIBUTE PARAMETER

Attribute	*Parameter*
Target number	N^T
Position	X_j^T
Weapon resources required	L_j^k
Target value	V_j^T
Is it known	Ikn_j

In order to complete the task, UAVs need to cooperative to execute the mission against all the targets. The execution process can be divided into three stages: flight stage, waiting stage and execution stage. In the flight stage, the UAV flies from its current location to the target location. In the waiting stage, after the UAV reaches the target location, it has to wait for the arrival of other cooperative UAVs. While in the execution stage, the UAV performs its assigned relevant tasks against the targets. In this paper, the jamming task time is set to be 10 minutes, and the attack task time is 5 minutes. The jamming UAV first performs 5 minutes of jamming, and the subsequent 5-minute jamming tasks are performed simultaneously with the attack tasks of other attack UAVs [43]. Fig. 1 shows the distribution relationship between UAV, target and task.

The task assignment structure of UAVs based on DT consists of two parts: the actual space and the simulation space. The UAVs interact with the command center during the whole flight process. The command center obtains various performance parameters of the UAVs and the targets. The parameters data is put into the simulation space for analysis and integration through existing algorithms. After the assignment scheme is obtained, the final strategy is fed back to the actual space for implementation. The connection of these two spaces depends on DT to interact. Due to the dynamic changes of the environment, the DTs of the UAVs and targets need to collect the corresponding changes and then transfer them to the simulation space. The simulation space dynamically adjusts the strategy according to the DT information, so that the scheme of the actual space can be updated quickly and efficiently. Fig. 2 shows the simulation actual interaction based on DT.

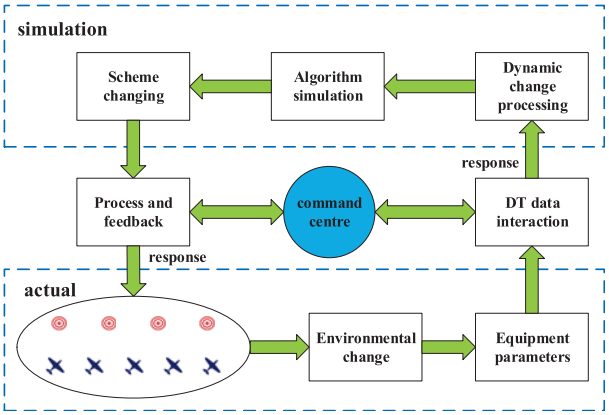


Fig. 2. Simulation actual interaction based on DT.

The system constructed in this paper consists of UAVs, targets and DT. As for the digital layer, the digital representation of physical entities (UAVs and targets) provides the operational dynamics of how the task assignment system operates, and the parameter data listed in Table I and Table II can be transmitted so as to calculate the assignment strategy. In particular, when environmental information or the parameters of the target change, the DT captures the change in the system, such as the emergence of new targets, the change of target location and so on, and synchronously calculates the assignment strategy. DT will eventually update a general strategy to the entity layer, guiding the UAVs to execute corresponding tasks during the subsequent movement. When the new target is discovered, DT will get the new target information quickly and give the appropriate assignment scheme in time. The establishment of DT can help make intelligent and predictive resource assignment decisions with little communication overhead.

B. Task Assignment Model

We use DT technology to conduct data interaction between actual space and simulation space, which can be expressed as

$$DT = \left\{ \left(U, \tilde{U} \right), \left(T, \tilde{T} \right) \right\}, \tag{1}$$

where U and T represent the physical entities of UAV and target, $\tilde{U}$ and $\tilde{T}$ represent the virtual copies of UAV and target copied by DT. Because data interaction through DT exists certain delay deviation Δt in time, the impact of delay Δt needs to be considered in the subsequent processing of the task assignment modeling.

Since the UAV resources are limited, and the task is expected be completed with minimum time and cost, the task

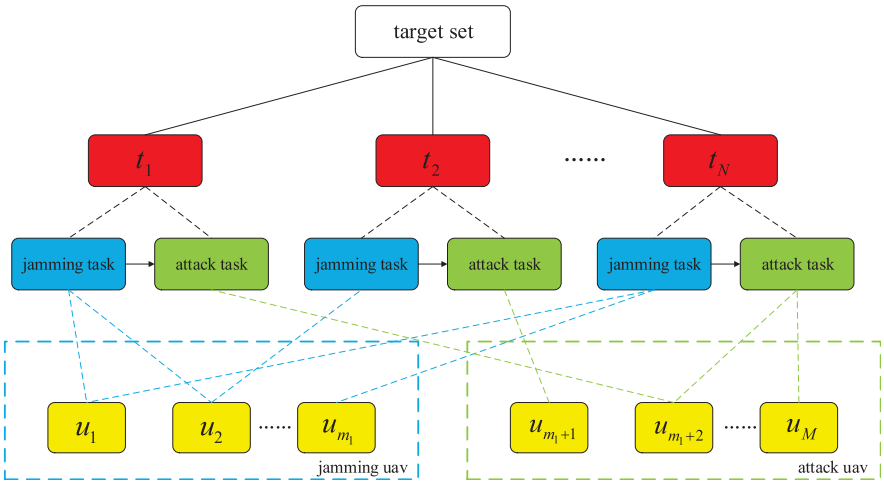


Fig. 1. UAV, target and task assignment diagram.

assignment should be optimized. Under the above constraints, in this part, four cost functions, that is the scheduling cost, path cost, risk cost and total task time cost are optimized to find a series of Pareto optimal solutions. The calculation of each cost function is as follows.

a) Scheduling Cost: The scheduling cost of UAVs during operation can be equivalent to the cost consumption of total operation time of different UAVs, which can be defined as

$$F_{sdu} = \sum_{i=1}^{M}\sum_{j=1}^{N} x_{i,j} \cdot V_i^U \cdot \left(T_{i,j}^{fly} + T_{i,j}^{wait} + T_{i,j}^{work}\right), \quad (2)$$

where $x_{i,j}$ represents the decision variable. If UAV i is assigned to target j, then $x_{i,j} = 1$. Otherwise, $x_{i,j} = 0$. $T_{i,j}^{fly}$, $T_{i,j}^{wait}$ and $T_{i,j}^{work}$ represent the flight time, waiting time and execution time of UAV i assigned to target j to perform tasks respectively.

b) Path Cost: Different assignment schemes will form different flight paths of UAVs. The longer the path is, the greater cost of the corresponding UAV becomes. The path cost can be defined as

$$F_{path} = \sum_{i=1}^{M}\sum_{j=1}^{N} x_{i,j} \cdot D_{i,j}, \quad (3)$$

where $D_{i,j}$ represents the length of the flight path from the UAV i to the target j, $D_{i,j} = ||X_j^T - X_i^U||$. It's obvious that the flight path length of the UAV is mainly reflected in the distance traveled within the flight time $T_{i,j}^{fly}$, $T_{i,j}^{fly} = D_{i,j}/v_i$. This paper assumes that the possible extra flight path length produced while the UAV is at waiting and execution stages is short and negligible [32].

c) Risk Cost: As the UAV's mission time increases, the UAV's remaining fuel will decrease. If the remained fuel gets fewer, the UAV will be at greater risk when performing the mission. At the same time, if the UAV waits around the target for a longer time, the possibility of exposure may increase, and it will bear a greater risk from target threat. Thus, the risk cost is defined as

$$F_{risk} = \sum_{i=1}^{M}\sum_{j=1}^{N} x_{i,j} \cdot (Fu_{i,j} + Wt_{i,j}), \quad (4)$$

where $Fu_{i,j} = \begin{cases} 10 \times 1.5^{\frac{fuel_i^{\max} - fu_{i,j}}{fu_{i,j}}}, & 0 < \frac{fu_{i,j}}{fuel_i^{\max}} \le 0.2 \\ 5 \times 1.33^{\frac{fuel_i^{\max}}{fu_{i,j}}}, & 0.2 < \frac{fu_{i,j}}{fuel_i^{\max}} \le 0.3 \\ 1 \times 2.15^{\frac{fuel_i^{\max}}{fu_{i,j}}}, & 0.3 < \frac{fu_{i,j}}{fuel_i^{\max}} \le 0.4 \\ 0, \quad \text{others} \end{cases}$

is the risk from the decrease of remaining fuel when the UAV i is performing task against the target j, and $Wt_{i,j} = \begin{cases} T_{i,j}^{wait} - T_{i,j}^{work}, & 3T_{i,j}^{work} \le T_{i,j}^{wait} \\ 5 \times \left(3T_{i,j}^{work} - T_{i,j}^{wait}\right), & 2T_{i,j}^{work} \le T_{i,j}^{wait} < 3T_{i,j}^{work} \\ T_{i,j}^{wait}, & T_{i,j}^{work} \le T_{i,j}^{wait} < 2T_{i,j}^{work} \\ 0, & 0 \le T_{i,j}^{wait} < T_{i,j}^{work} \end{cases}$

is the risk generated when the UAV i is at the waiting stage against the target j. $fu_{i,j}$ means the remaining fuel of UAV i when performing the task of target j. $fuel_i^{max}$ represents the maximum fuel value of UAV i.

d) Total Task Time Cost: As mentioned earlier, the UAVs assignment optimization is to guarantee that all tasks can be allocated with resources and executed as quickly as possible. Different assignment schemes and execution sequences will affect the completion time of each task, resulting in changes in the final total time consumption, which is defined as

$$F_{time} = Time_{j^{last}}^{end} + \Delta t, \quad (5)$$

where Δt represents the delay deviation caused by DT [10]. The task execution order of all targets is sorted according to the value of the j-th target's V_j^T in descending order. The larger V_j^T is, the earlier the UAV will be assigned to the target. The subscript j^{last} represents the sequence number of the target whose value is the smallest. In complex dynamic scenarios, j^{last} should be updated according to the reordering result if a new target is found. $Time_{j^{last}}^{end}$ indicates the completion time of the attack task of the j^{last}-th target.

To some extent, the flight path length of UAVs can be converted into flight time. But path cost does not need to consider the value of UAVs. Besides, the existence of waiting time makes flight time and path length not equivalent. Under the prerequisite that all the tasks must be completed, the above cost functions to be optimized will conflict with each other. For example, if we want a smaller path cost, we can make the UAV choose the nearby distributed target. But the waiting time for other cooperative UAVs may be longer, thus increasing the risk cost. If we want to spend less time on the total task to complete the last task as soon as possible, we should use more UAVs to participate in the assignment. But it may increase the scheduling cost. If we need to find a better path, it may increase the path cost. At the same time, the phenomenon of no UAV participating in the assignment must be avoided because the constraint that the target task must be executed needs to be satisfied. Therefore, this paper establishes the assignment model as a multi-objective optimization model to refrain from weighting these cost functions. The multi-objective function of UAVs assignment optimization problem is expressed as

$$\min\left(F_{sdu}, F_{path}, F_{risk}, F_{time}\right) \tag{6}$$

$$s.t. \sum_{i=1}^{m_1} x_{i,j} \geq 1, \sum_{i=m_1+1}^{M} x_{i,j} \geq 1 \tag{6-a}$$

$$\sum_{i=1}^{M} x_{i,j} \cdot resSu_i^k \geq res\mathrm{Re}_j^k \tag{6-b}$$

$$\sum_{j=1}^{N} x_{i,j} \cdot dis_{i,j} \leq dis_i^{\max} \tag{6-c}$$

$$\sum_{j=1}^{N} x_{i,j} \cdot time_{i,j} \leq time_i^{\max} \tag{6-d}$$

$$\sum_{i=1}^{M} x_{i,j} \cdot (Fu_{i,j} + Wt_{i,j}) \leq risk^{\max} \tag{6-e}$$

$$x_{i,j} \in \{0,1\}, \forall (i,j) \in M \times N. \tag{6-f}$$

In the formula, F_{sdu}, F_{path}, F_{risk} and F_{time} are jointly optimized to minimize the costs as much as possible. Constraint (6-a) means that for any target j, there must be at least one jamming UAV and one attack UAV assigned to it. In constraint (6-b), $resSu_i^k$ means the number of weapon resources k that UAV i can contribute, and $resRe_j^k$ means the number of weapon resources k required by the target j, $k \in \{k_1, k_2, k_3, k_4\}$. For any target j, the number of weapon resources k contained in UAV team for a feasible scheme must not be less than the demand of target j for weapon resources k. Constraint (6-c) means that the sum of the flying distance of UAV i to the assigned targets must be within its range dis_i^{max}. Constraint (6-d) means that the sum of the flying time of UAV i to the assigned targets must be within its maximum flying time $time_i^{max}$. And constraint (6-e) means that the sum of the risk coefficients of the UAV team assigned to each target j cannot be higher than the maximum risk coefficient $risk^{max}$ during the task execution. This constraint mainly limits that a UAV shouldn't participate in too many target tasks, and the proportion of waiting time should be reduced as much as possible.

C. Dynamic Scenarios

The actual environment is changing rapidly, and the amount of information presented at different time varies greatly. It makes the problem model and solution algorithm more complex and more difficult to obtain the desired optimal solution. Dynamic research mostly focuses on the continuous movement of targets and the discovery of obstacles during the flight of UAVs in unknown environments. The UAVs are assumed to be able to detect the emergence of new targets within the action radius R_i^{detect} during operation. At the same time, dynamic changes such as the change of target weapon resource requirements and the change of the number of targets in the satellite detection area are also considered.

a) Change in Target Quantity: In a given environment, apart from the already known targets, there may be some undetected targets that can be found by UAVs in the process of task execution. New targets may also fly into the current area. So before the assignment scheme is obtained, the number of targets may increase. If the target j changes from unknown to known, we can note as

$$IKn_j = 0 \Rightarrow IKn_j = 1, \tag{7}$$

when a new target is found, it needs to be allocated with UAV resources immediately without affecting the original assignment scheme.

b) Change of Target Weapon Resource Demand: The target weapon resource demand is assumed to be known through previous satellite reconnaissance and evaluation [7]. But in the process of assignment, the target's demand for different types of weapon resources may increase or decrease. If the weapon resource requirements of target j are: one jamming weapons k_1, one jamming weapons k_2 and one attack weapons k_4, we can note as

$$N_j^{k_1} = 1, N_j^{k_2} = 1, N_j^{k_3} = 0, N_j^{k_4} = 1, \tag{8}$$

if target j's demand for attack weapon resource k_3 increases and its demand for jamming weapon resource k_2 decreases, we can note as

$$\begin{matrix} N_j^{k_3} = 0 \Rightarrow N_j^{k_3} = 1 \\ N_j^{k_2} = 1 \Rightarrow N_j^{k_2} = 0 \end{matrix}, \tag{9}$$

no matter how the weapon resource demand of the target changes, the assignment scheme of the target must meet the constraint (6-b). Therefore, when such a change occurs, the assignment scheme needs to be adjusted.

c) Change of target coordinates: Compared with the above two changes, the changes of the coordinates of the targets occur more frequently. In the real environment, the targets may not always stay in the same position. If the coordinates of the target change, the calculation of each cost function also changes. If the coordinates change a lot (its calculation method is reflected in Section IV), the assignment scheme of the target needs to be re-evaluated. Other generated assignment schemes will need to make corresponding adjustments.

IV. Algorithm Design

To solve the multi-objective optimization model (6), this paper proposes an improved dynamic multi-objective adaptive weighted particle swarm algorithm. In order to adjust to the dynamic complex scenarios, the response strategy to environmental change in different situations are given in this part.

A. Overall Algorithm Description

The main flow chart of the algorithm proposed in this paper is shown in Fig. 3.

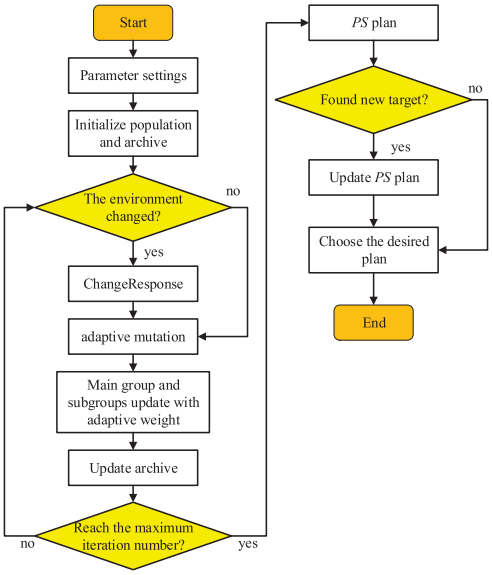


Fig. 3. Flowchart of improved algorithm.

Once the environment is changed, it performs the change response. Then it mutates adaptively and update the group and archive until the *PS* (Pareto-optimal Set) plan is obtained. After the iteration of the algorithm is completed, if a new target is discovered during the execution process of the current scheme, two types of UAVs will be assigned with new tasks against it preferentially: those that haven't been assigned with tasks, and those that are in the waiting stage but can complete the newly assigned task before the current task starts. The detailed operation steps are designed in the following parts.

B. Encoding

In order to facilitate the subsequent calculation of the algorithm and reduce the redundancy of data, this paper encodes the assignment relationship between UAVs resources and targets as shown in Fig. 4.

In Fig. 4, the scheme that UAVs assigned to all the N targets is exhibited as an individual consisting of $M \times N$ codes, where code 0 means the UAV is not assigned to the target, and code 1 means the opposite. This coding method can reflect the many-to-many assignment relationship better. The population is recorded as pop, and the individual in the population is recorded as pop_y.

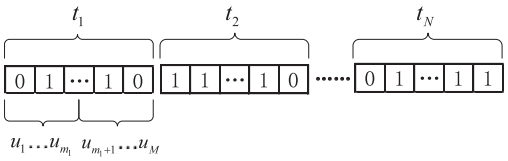


Fig. 4. Chromosome coding.

C. Heuristic Initialization

In initialization phase, most individuals use the chaos initialization [44] to cover a wider solution space. For one part of the rest individuals, the assignment schemes are determined according to the distance between the UAV and the target. In this way, the UAV is assigned to the nearest target, and the maximum number of UAVs allocated to the same target is constrained. For the other part of the rest individuals, all targets are randomly assigned one jamming UAV and one attack UAV. If the scheme does not meet the constraint of weapon resources, the UAV is randomly added to meet the constraint. The above methods can make the algorithm generate multiple effective optimal solutions quickly to guide the subsequent evolution of the population and accelerate the convergence speed of the population.

D. Adaptive Mutation

The intelligent optimization method is easy to fall into the local optimum in the process of evolution. Therefore, this paper proposes an adaptive mutation method to increase the diversity of the Pareto solution set and prevent the algorithm from falling into the local optimum convergence. The mutation algorithm mainly includes three mutation operators:

1) *Mutation Operator 1:* This mutation operator is mainly aimed at the bit-flip mutation between the same pop_y [45]. pop_y is randomly divided into two parts P_A and P_B. And then part P_A is randomly divided into 3 segments. The 3 segments are randomly sorted and combined. Data in part P_B is randomly mutated, 0 to 1 or 1 to 0.
2) *Mutation Operator 2:* Randomly select two targets in the current pop_y, and exchange their corresponding assignment schemes;
3) *Mutation Operator 3:* Calculate the number of UAVs allocated to each target in the current pop_y. Then find out the target assigned with the largest number of UAVs. If the number of UAVs assigned to it is greater than the threshold of UAV number, one UAV is randomly deleted from the target assignment plan.

Whenever a mutation operation is required, one of the three operators is randomly selected with the same probability. That's because compared with other selection probabilities, it can improve the diversity and optimum convergence performance of the adaptive mutation further from the point of uncertainty and comentropy. For the operation of the mutation method, part of the flow chart is shown in the Fig. 5.

In Fig. 5, the input individuls in the popolation chooses whether to mutate according to probability pm, and then

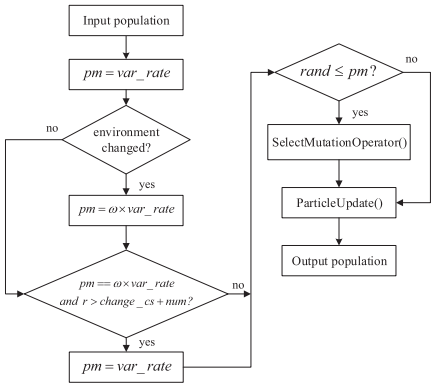


Fig. 5. Adaptive mutation selection.

outputs a new population. *var_rate* represents the preset mutation probability pm. ω represents the change coefficient of the mutation rate after the environment changes. In this paper, $\omega = 2$. r represents the current number of iterations. *change_cs* represents the iteration number before the environmental changes, and *num* represents the number of iterations using the new mutation rate after environmental changes. $num = 10$ is taken in this paper. *SelectMutationOperator()* represents to select one of the three mutation operators with the same probability and to mutate. *ParticleUpdate()* represents the update function of particle position and velocity. Through this adaptive mutation selection, the population pop can traverse more solution spaces in the new environment and accelerate the convergence speed.

E. Use of Subgroups

In this part, a method of joint optimization of multiple subgroups is designed. On the basis of reference [46], this paper divides the population pop into the main group pop^{main} and four subgroups $pop^{subgroup}$. The main group pop^{main} searches for the Pareto optimal solution according to the normal algorithm flow. And each subgroup $pop^{subgroup}$ optimizes a single objective function to find the optimal solution of a single objective. However, in the proposed algorithm, the subgroup $pop^{subgroup}$ and the main group pop^{main} share an external archive A, which stores the non-dominated solution in the main group pop^{main} and the optimal solution in the subgroup $pop^{subgroup}$. In each iterative update, the subgroup selects the optimal individual pop_y of its corresponding objective function from the external archive A as the guide at first. Then, the updated optimal individual pop_x within the subgroup takes the place of the dominant solution in the external archive A. If the number of solutions stored in the external archive exceeds the given size, the farthest priority selection method [47] is used to delete the spare solution. The use of subgroups can increase the diversity of the final Pareto optimal solution set, especially the boundary solution to each objective function.

F. Dynamic Processing Scheme

For the dynamic multi-objective optimization problem, the solution algorithm should be able to deal with different scenarios. The following two points need to be guaranteed as far as possible [48]:

1) If the environment changes, the algorithm must ensure that it can sensitively detect the changes of the environment and effectively respond to the environment changes.
2) If the environment does not change, the algorithm should track the Pareto front of the current environment as quickly as possible.

This section mainly discusses the processing of dynamic environment: environmental change detection and environmental change response.

a) Environmental Change Detection: This paper proposes to generate several detection individuals pop_c by heuristic initialization according to the current environment after the end of each generation and before the start of the next generation. They are used to detect whether their objective function values have changed in the subsequent environment. Then, it generates several new detection individuals $pop_c^{'}$ according to the subsequent environment. These individuals $pop_c^{'}$ are put into the old environment for detection. Through such detection, it can accurately reflect whether the environment has changed with a smaller computational cost.

As shown in Fig. 2, when the environment changes are detected, DTs obtain the details of the changes and interacts with the simulation space in time. Through the above-mentioned environmental change detection measures, we can detect changes in the environment in time, and take a series of measures to ensure the diversity and convergence of population pop in the new environment.

b) Environmental Change Response: Different dynamic situations have different effects on the population pop. If the target's location changes significantly, some assignment schemes may need to be adjusted or even reconstructed. The discovery of new targets will also greatly affect some of the existing assignment schemes. Therefore, this paper proposes to take the following response measures according to the situations and severity of environmental changes.

Firstly, the severity of environmental change can be defined as [40]:

$$\delta(t) = \frac{1}{F}\frac{1}{K}\sum_{u=1}^{F}\sum_{v=1}^{K}\left|\Delta f_u(sol_v) - \mu_u(g)\right|, \qquad (10)$$

where $\Delta f_u(sol_v) = \frac{f_u(sol_v,g) - f_u(sol_v,g-1)}{p_u(g) - l_u(g)}$ and $\mu_u(g) = \frac{1}{K}\sum_{v=1}^{K}\Delta f_u(sol_v)$, $F = 4$ represents the number of objective functions, K represents the number of individuals in the population pop, $f_u(sol_v, g)$ represents the u-th objective function value of the v-th individual in the g-th environment, $p_u(g)$ and $l_u(g)$ represent the maximum and minimum values of the u-th objective function in the g-th environment.

Then, the centroid of the Pareto front set obtained at time t can be calculated as [38]:

$$C_t = \frac{1}{|PS_t|} \sum_{i=1}^{|PS_t|} PS_{ti}, \tag{11}$$

where $|PS_t|$ represents the number of solutions corresponding to the Pareto-optimal set of the population pop at time t, and $|PS_{ti}|$ represents the function value corresponding to the i-th individual in the frontier set at time t.

Assuming that the environment at the time $t+1$ has changed compared to the environment at the time t, the degree of change of the centroid after change is

$$S_t = \|C_{t+1} - C_t\|. \tag{12}$$

In fact, it is equal to the Euclidean distance between them.

Next, the lowest point composed of the maximum vavlues of each objective in the Pareto-optimal set in the new environment can be found as [49]

$$z_u^{mad} = \max F_u(x^*), x^* \in PS, \tag{13}$$

where $F_u(x^*)$ represents the function value of the u-th cost function. Compare S_t with the distance betwen z_u^{mad} and C_t, the new change intensity $str(t)$ can be obtained as

$$str(t) = \frac{S_t}{\|C_t - z_u^{mad}\|}. \tag{14}$$

It indicates the degree to which the moving distance of the centroid relative to the moving distance in the worst case of the change.

Finally, The response strategy to environmental change in different situations is shown in Algorithm 1.

In this algorithm, the assignment scheme is initialized by *single_init()* only for newly emerging targets. *farthest_selection()* indicates the farthest priority selection method [36]. *heuristic_initialization()* indicates the heuristic initialization method. And *pop_selection()* means to add the solution set in archive A to the population in the new environment at first. Then, if the number of schemes in archive A is insufficient, it is filled with randomly selected optimal historical individuals of the population. *popgbest_selection()* means to extract the optimal historical individuals of the population. *judge_domination()* means to judge whether there is a dominant solution in the new archive after adding the non-dominanted solution of the population. If it exists, delete the dominant solution.

Algorithm 1 ChangeResponse()

Input: pop (population), A (archive), K (population size);
Output: pop' (new population), A' (new archive);

if *found a new target* **then**
  $sigpop_1$ = single_init(K);
  $sigpop_2$ = single_init($3K/4$);
  pop_1' = farthest_selection($pop, K/4, sigpop_2/4$);
  pop_2' = heuristic_initialization($K/4$);
  pop_3' = pop_selection ($pop, A, K/2, sigpop_2/2$);
  pop_4' = popgbest_selection ($pop, sigpop_1$);
  pop' = $[pop_1', pop_2', pop_3', pop_4']$;
  pop' = farthest_selection(pop', K);
  A' = judge_domination(pop');
else if $\delta(t) < str(t)$ **then**
  pop_1' = farthest_selection($pop, 3K/5$);
  pop_2' = heuristic_initialization($K/5$);
  pop_3' = mutations ($pop, K/5$);
  pop' = $[pop_1', pop_2', pop_3']$;
  A' = judge_domination(pop');
else if $\delta(t) \geq str(t)$ **then**
  pop_1'= pop_selection ($pop, A, K/2$);
  pop_2' = farthest_selection($pop, K/4$);
  pop_3' = heuristic_initialization($K/4$);
  pop' = $[pop_1', pop_2', pop_3']$;
  A' = judge_domination(pop');
end if
return pop', A';

G. Complexity and Scalability Analysis

In this part, we analyzed the complexity and scalability of the proposed algorithm in detail. As shown in Fig. 3, the proposed framework consists of the following three parts:

1) Environmental change detection and response: environmental change detection is to give some special solutions for calculation, which is negligible compared with the environmental change response. In most cases, the response to environmental change requires the transformation of every individual in the population, so the computational complexity can be recorded as $O(N^2)$.
2) The main body of AWPSO-based algorithm: the computational complexity is $O(MN^2)$. Specifically, M represents the number of objective functions that is much smaller than N [50].
3) The verification and detection of the final scheme: it is necessary to evaluate each of the solution in the resulting Pareto solution set, so the computational complexity is $O(N)$.

According to the above analysis, the computational complexity of the proposed algorithm is $O(N^2)$.

Then, the complexity of the proposed algorithm is compared with other three algorithms, that is, dynamic NSGA-II (DNSGAII) [33], steady-state and generational evolution algorithm (SGEA) [38], AWPSO. While the AWPSO algorithm is originally a single-objective optimization algorithm. So we add dynamic change detection and response as well as multi-objective processing to adapt to the scenario, named DMOAWPSO. It can be seen from Table III that, the complexity of the proposed algorithm is the same as that of the other three algorithms compared.

As for the scalability, the proposed algorithm has strong scalability and can be used in combination with other algorithms and techniques. It consists of several operation steps that can be extracted to other artificial intelligence algorithm. Besides, the proposed algorithm can be used to solve most dynamic multi-objective problems, such as vehicle network communication and job shop scheduling.

TABLE III COMPARISON OF COMPLEXITY OF DIFFERENT ALGORITHMS

	DNSGA-II	***SGEA***	***DMOAWPSO***	***proposed***
Algorithm complexity	$O\left(MN^2\right)$	$O\left(MN^2\right)$	$O\left(MN^2\right)$	$O\left(MN^2\right)$

V. EXPERIMENTAL ANALYSIS

In this section, the feasibility, effectiveness and superiority of the proposed algorithm compared with other algorithms are verified by simulation experiments. There are many kinds of evaluation indexes for dynamic multi-objective optimization algorithms:(1) Accurate measurement index: GD (Generational Distance), SR (Success Ratio) and SC (two Set Coverage) [51], etc. (2) Diversity index: S (Spacing), PD (Pure Diversity) [52] and CS (Coverage Scope), etc. (3) Comprehensive index: IGD (Inverted Generational Distance), HV (Hypervolume) [53] and HVR (Hypervolume Rate), etc. The number of indexes is relatively large. It is impossible for all indexes to be used in the experiment. This paper comprehensively analyzes the performance of the algorithm in terms of convergence, distribution and diversity through four evaluation metrics: the Number of Pareto Solutions (NPS) [54], HV, PD and SC. These four evaluation indicators selected above can balance accuracy, diversity, and comprehensiveness, comprehensively reflecting the superiority of the proposed algorithm in practical applications.

Assume that the simulation scenario is in a three dimensional space with size of 500km×500km×10km. There are 7 targets and 12 UAVs, including 4 known targets and 3 newly emerging targets. The relevant parameters of the UAVs and targets can be set similarly to [31], [32]. The number of environmental changes is 8, including 5 changes of target position, 2 changes in the requirement of weapon resources for the target and 1 change in the discovery of a new target. The delay deviation value caused by DT Δt can be referred to [10]. Both of the population *pop* K and the maximum number of iteration are set as 100.

A. *Effectiveness Testing on the Changes on the DMOAWPSO*

This part conducts the comparison experiments of different changes that can be made to the orignal DMOAWPSO. Among them, the changes adopted in the proposed algorithm can be found in Section II and IV. Then, the effectiveness of our proposed processing steps is reflected in the Table IV and Table V.

In Table IV, the items in the first column from top to bottom are the original DMOAWPSO algorithm, DMOAWPSO with heuristic initialization, DMOAWPSO with subgroup, DMOAWPSO with heuristic initialization and subgroup, DMOAWPSO with adaptive mutation as well as heuristic initialization and subgroup, respectively. The last item represents our proposed method. From the results of the three evaluation indicators, we can see that the optimization effect of the proposed method is the most obvious. All of the three indicators have been improved.

Table V mainly focuses on the performance comparison of different combination of dynamic schemes. The items in the first column from top to bottom are: DMOAWPSO with re-evaluation detection and complete initialization response, DMOAWPSO with steady-state mode detection and response, DMOAWPSO with the steps in the last item in Table IV and steady-state mode detection and response, the proposed algorithm. From the results of the three evaluation indicators, it can be seen that the proposed algorithm has the best performance compared to other combination schemes.

For the mutation part, one of the three mutation operators needs to be selected randomly with equal probability. To verify the effectiveness of the equal selection probability, its performance is compared with other mutation operations with different selection probability of the three mutation operators shown in Table VI.

The items in the first column in Table VI represent the selection probabilities of the three mutation operators. Specifically, the first row and the last row represent the comparison of the three evaluation indexes when the mutation operator is not used and the three mutation operators are selected with the equal probability, respectively. It can be seen that both the NPS and the PD are best when selecting the mutation operator with the equal probability. The HV is the best when the variation operators are selected with probabilities represented by the third row. But in this instance, the improvement of other indicators is not as good as selecting the mutation operators with the equal probability. Moreover, the difference of HV between instances is not much. So, this paper adopts the equal probability to select three mutation operators. It means that the selection probability of each of the three mutation operators is 1/3.

B. *Simulation Results of Proposed Algorithm*

The initial simulation scenario is shown in Fig. 6

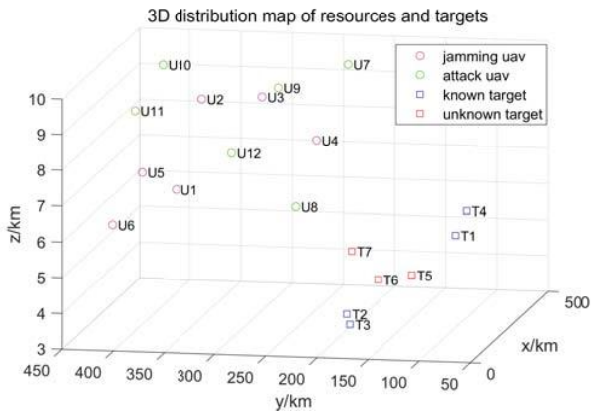


Fig. 6. Initial distribution of UAVs and targets.

There are 4 known targets and 3 unknown targets in the initial state. These unknown targets may be found in the

TABLE IV COMPARISON OF DIFFERENT STEP COMBINATIONS

Different improvements	*NPS*	*HV*	*PD*
DMOAWPSO	7.3667	0.2463	6.8787E+07
DMOAWPSO_init	7.7667	0.2762	5.0883E+07
DMOAWPSO_subgroup	8.7000	0.2797	6.9047E+07
DMOAWPSO_init_subgroup	8.6000	0.2936	7.2691E+07
DMOAWPSO_init_subgroup_advari	**9.4000**	**0.3011**	**9.3251E+07**

TABLE V COMPARISON OF DIFFERENT STEP COMBINATIONS

Different improvements	*NPS*	*HV*	*PD*
DMOAWPSO_randinitdy	7.3667	0.2463	6.8787E+07
DMOAWPSO_SGEAdy	10.2333	0.3017	7.2035E+07
stepimprove_SGEAdy	12.2000	0.3330	1.1934E+08
stepimprove_dyimprove	**13.1667**	**0.3413**	**1.3105E+08**

TABLE VI COMPARISON OF MUTATION WITH DIFFERENT SELECTION PROBABILITY OF OPERATORS

Mutation operator	*NPS*	*HV*	*PD*
0,0,0	8.3667	0.2863	8.8787E+07
1,0,0	10.9000	0.3234	1.0250e+08
0,1,0	11.9000	**0.3528**	1.2238e+08
0,0,1	12.6000	0.3256	1.1560e+08
0.5,0.25,0.25	12.2000	0.3489	1.0607e+08
0.25,0.5,0.25	13.0000	0.3493	9.9903e+07
0.25,0.25,0.5	13.3000	0.3401	1.0276e+08
0.33,0.33,0.33	**13.3000**	0.3467	**1.3034e+08**

subsequent dynamic process or finally in the flight process of the UAV. After completing the simulation of the maximum number of iterations, the obtained optimal solution set is analyzed to determine whether each solution has the ability to discover unknown targets. If any unknown target is found, it needs to be assigned UAVs immediately.

Because there are 4 objectives to be optimized in this paper, the parallel coordinate visualization [55] is adopted to display the final Pareto frontier. In order to compare the advantages and disadvantages of each scheme in terms of four cost metrics intuitively, this paper uses the z-score method to process the data. The final Pareto solution set can be represented as show in Fig. 7.

From Fig. 7, it can be seen that different schemes have different advantages and disadvantages on each cost. The final assignment scheme can be determined according to the requirement of practical scenarios. If we need lower scheduling and path costs, scheme 12 is the most suitable one. While if we are more concerned about risk costs and total task time, scheme 1 or 5 can be chosen. The performance of the schemes in the middle part are relatively average.

Taking Scheme 4 in Fig. 7 for an example. Fig. 8 shows the orign scheme after the simulation iteration. It shows the UAV assignment plan in the upper subfigure (a), in which the dotted line and the solid line represent the motion trajectories of the attack UAV and jamming UAV respectively. The lines of the same color represent the motion trajectory of the same

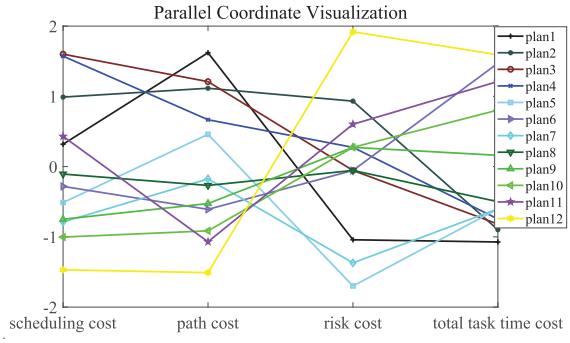


Fig. 7. Parallel coordinate visualization of final frontier solution set.

UAV. For the mission schedule in the lower subfigure (b), the green virtual line represents the mission completion time of each target. For instance, the green five-star indicates that the UAV 9 discovered a new target 7 at the moment indiated by the red vertical line during its flight time. Fig. 9 shows the change of UAV assignment plan and schedule after the first unknown target is found.

From Fig. 9, after the UAV 9 discovers the unknown target 7, the system assigns the UAV 2, UAV 5 and UAV 7 to perform the tasks of the target 7. Among them, UAV 5 and UAV 7 are idle UAVs thaSt have not participated in the assigned tasks before. They start to fly and execute the

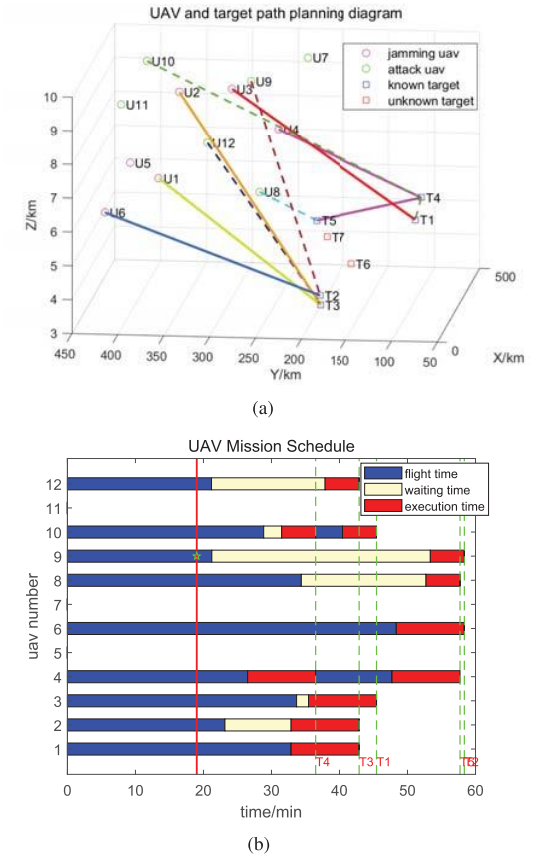


Fig. 8. Assignment plan and schedule when UAV is not moving.

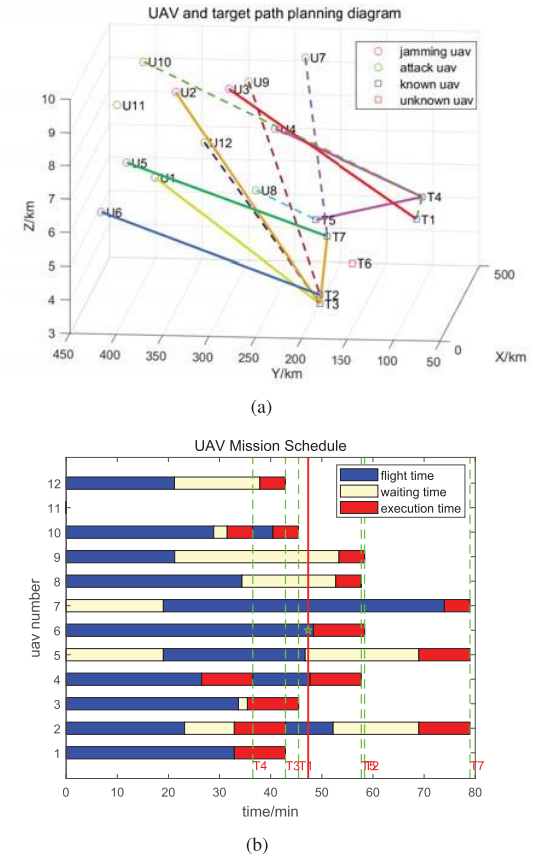


Fig. 9. Assignment plan and schedule after the first unknown target is discovered.

mission after the discovery. UAV 2 has to execute task of target 7 after completing all the current assigned tasks. At the same time, UAV 6 also finds the second unknown target 6 during the subsequent flight process (the time indicated by the red vertical line in the subfigure 8(b)). Fig. 10 shows the changed assignment plan and schedule of UAVs after the second unknown target is found.

As shown in Fig. 10, after the unknown target 6 is found, the system sends the UAV 5 and UAV 10 to perform the task of the target 6. It is worth noting that UAV 5 is just at idle waiting time when performing the current task. So it can execute the subsequent task by using the idle time effectively. In this case, it can reduce the risk cost and the total time of the task. By this time, after all the targets are assigned with UAVs, the assignment scheme in Fig. 10 is the final form.

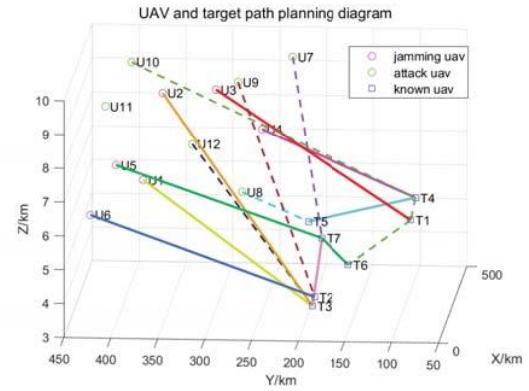
C. Comparison of Different Algorithms Under Multiple Metrics

In order to evaluate the performance of the proposed algorithm, this paper compares it with the other three classical and popular algorithms: DNSGA-II, SGEA, DMOAWPSO. However, their original application scenarios are different, making it unsuitable to the scenario constructed in this paper. The modifications of the comparison algorithms are needed.

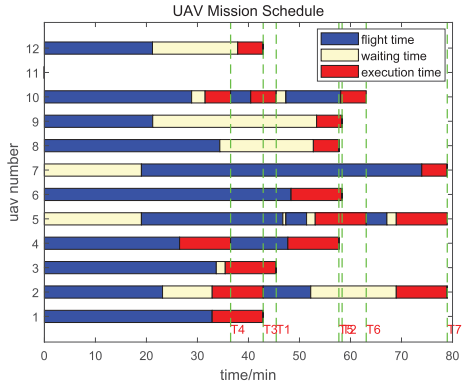
Among them, the NSGA-II algorithm is a multi-objective optimization algorithm. We add the dynamic change detection and dynamic change response to adapt to the scenario assumed in this paper. While the AWPSO algorithm is originally a single-objective optimization algorithm. So we add multi-objective processing, dynamic change detection and dynamic change response methods to adapt to the scenario.

The variation curves of HV metric of four different algorithms during the iteration are provided in Fig. 11. The results are the average of 30 times experiments for each algorithm.

It can be seen that the proposed algorithm has a faster convergence speed. That's because the application of heuristic initialization method makes it have a higher starting HV. At the same time, the proposed algorithm has faster recovery ability and more stable response to dynamic changes. For DMOAWPSO algorithm, it is obvious that it can not adapt to the new environment quickly after the dynamic change of the environment. That's mainly because it loses the diversity of the previous environment and the search scope in the new environment is limited. For SGEA algorithm, it also has good stability in the face of dynamic changes in the environment. But it falls into local optimization to a large extent, and there is no good method to jump out of local optimization. As for DNSGA-II algorithm, because of its slow



(a)



(b)

Fig. 10. Final assignment plan and schedule.

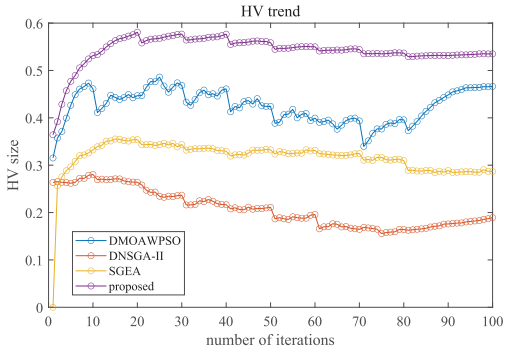


Fig. 11. Average HV change curve of different algorithms.

convergence speed, it needs more iterations to achieve better results. Therefore, when the environment changes dynamically, its response strategy will affect the previous results, further making its diversity lost and convergence worse.

In addition to the comparison of different algorithms by the comprehensive measurement metric HV under the above scenario, this paper further designs 9 groups of different test instances to compare the NPS, HV, PD and SC metrics of each algorithm. The results are the average of 30 times experiments for each algorithm. The test results of different metrics are shown in Tables VII to X.

Take the first test instance 5_5_5_0_0 for an example. The meaning of the digits in the test instance is that it consists of 5 jamming UAVs, 5 attack UAVs, and 5 targets, among which there is 0 newly emerging targets, and 0 changes. By that analogy, we can recognize the difference of different test instances.

It can be seen from Table VII that the proposed algorithm achieves the maximum NPS value among all the test instances. Because the problem considered in this paper is discrete, and the final solution set may contain a large number of repeated solutions, it is particularly important to consider the number of non-repeated Pareto solution sets. Higher NPS metric can improve the quality of decision-making [48]. Overall, the difference of NPS in each test instance is not very large. But the larger the scale is, the smaller the metric NPS will be. Due to the complex constraints, it will be more difficult to find a new effective Pareto optimal solutions in a larger-scale scheduling.

Table VIII shows that the proposed algorithm can achieve the maximum HV value in all test instances. In the test instances with fewer changes and smaller scale, the proposed algorithm can obtain larger HV. That's because smaller scale and especially fewer targets will make the cost reduced accordingly. It is worth noting that in the test instance 4_8_8_3_10, the HV metric is obviously small. The possible reason is that compared with the number of the targets, the jamming weapon resources are relatively too few to perform jamming task in both the jamming and the subsequent attack mission stages. Each jamming UAV needs to be assigned to different targets as much as possible, which leads to the greatly increased risk cost of the jamming UAV, thus affecting the HV metric.

We can see from Table IX that the proposed algorithm can obtain the maximum PD value in most test instances. But for the first group of test instance 5_5_5_0_0 in a static environment, it is obvious that the congestion calculation and elite retention strategy in DNSGA-II algorithm and SGEA algorithm can provide better diversity. At the same time, in the test instances 8_8_10_3_10 and 10_10_10_4_10 with large scale and many changes, the diversity of DNSGA-II algorithm and SGEA algorithm is relatively better than the proposed algorithm. In addition to the different strategies adopted by the two algorithms mentioned above, the possible reason is that in a larger scale environment, complex constraints are more stringent, which limits the generation of better solutions. This is similar to the phenomenon in Table III. While DNSGA-II algorithm and SGEA algorithm retain more suboptimal solutions in each iteration process. In these frequent changing environment, the proposed response strategy does not cover enough optimization space. Its memory based retention strategy will lose a certain diversity. But this stragety significantly improves the convergence speed.

Table X shows the two set coverage SC between two of the four algorithms. It can intuitively reflect the domination of the solution set obtained by one algorithm over the solution set obtained by another algorithm. Its calculation formula is as follows [51]

TABLE VII MEAN AND STANDARD DEVIATION VALUES OF NPS METRIC OBTAINED BY FOUR ALGORITHMS

Instance	DMOAWPSO		DNSGA-II		SGEA		Proposed	
	avg	*std*	*avg*	*std*	*avg*	*std*	*avg*	*std*
5_5_5_0_0	10.2333	3.9800	4.9667	2.3122	8.2667	2.9235	**12.5667**	**1.6955**
5_5_6_1_5	12.8667	5.2439	4.8333	1.9313	3.0667	1.3880	**18.1333**	**3.8483**
5_5_7_2_5	6.8667	3.0708	2.4667	1.1366	2.8667	1.1666	**10.1667**	**3.9136**
6_6_7_3_8	8.7333	2.9353	2.9667	1.5643	2.7000	1.1788	**13.1667**	**4.1778**
7_7_7_2_8	8.6000	3.3383	3.3667	1.9911	2.8333	1.4162	**13.2667**	**4.0593**
8_8_10_3_10	7.3667	4.7885	3.9667	1.4735	4.3333	1.6259	**10.7333**	**5.2976**
10_10_10_4_10	7.0667	4.2825	3.4667	1.2243	2.8333	1.1769	**10.9667**	**4.9999**
4_8_8_3_10	7.5333	3.5402	3.1667	1.6626	3.5333	1.7167	**13.5667**	**3.4709**
8_4_8_3_10	8.0667	3.6287	2.7000	1.5790	4.5667	2.4870	**10.1667**	**4.6984**
Average	8.5926	3.8676	3.5445	1.6527	3.8889	1.6755	**12.5259**	**4.0179**

TABLE VIII MEAN AND STANDARD DEVIATION VALUES OF HV METRIC OBTAINED BY FOUR ALGORITHMS

Instance	DMOAWPSO		DNSGA-II		SGEA		Proposed	
	avg	*std*	*avg*	*std*	*avg*	*std*	*avg*	*std*
5_5_5_0_0	0.4730	0.0514	0.2732	0.0700	0.4087	0.0312	**0.5188**	**0.0205**
5_5_6_1_5	0.4536	0.0447	0.2300	0.0465	0.3178	0.0568	**0.5341**	**0.0349**
5_5_7_2_5	0.2356	0.0297	0.0882	0.0447	0.1626	0.0341	**0.2891**	**0.0219**
6_6_7_3_8	0.2832	0.0454	0.1075	0.0459	0.1708	0.0363	**0.3413**	**0.0255**
7_7_7_2_8	0.2434	0.0376	0.0894	0.0398	0.1232	0.0323	**0.3126**	**0.0348**
8_8_10_3_10	0.1466	0.0530	0.0071	0.0114	0.0522	0.0209	**0.2184**	**0.0423**
10_10_10_4_10	0.1676	0.0469	0.0265	0.0242	0.1000	0.0327	**0.2540**	**0.0462**
4_8_8_3_10	0.1136	0.0319	0.0247	0.0172	0.0679	0.0255	**0.1957**	**0.0336**
8_4_8_3_10	0.2163	0.0522	0.0634	0.0279	0.1444	0.0322	**0.2769**	**0.0340**
Average	0.2592	0.0436	0.1011	0.0364	0.1720	0.0336	**0.3268**	**0.0326**

TABLE IX MEAN AND STANDARD DEVIATION VALUES OF PD METRIC OBTAINED BY FOUR ALGORITHMS

Instance	DMOAWPSO		DNSGA-II		SGEA		Proposed	
	avg	*std*	*avg*	*std*	*avg*	*std*	*avg*	*std*
5_5_5_0_0	4.5768E+07	1.9540E+07	8.2460E+07	4.6441E+07	**1.2557E+08**	**4.6195E+07**	4.7251E+07	2.6209E+07
5_5_6_1_5	5.8475E+07	3.8912E+07	7.9775E+07	4.5234E+07	5.3882E+07	4.7228E+07	**8.9094E+07**	**4.1671E+07**
5_5_7_2_5	4.7943E+07	3.2430E+07	4.1639E+07	3.7298E+07	6.1939E+07	4.6229E+07	**7.7265E+07**	**3.8400E+07**
6_6_7_3_8	6.5223E+07	3.6328E+07	6.0329E+07	4.9757E+07	5.0847E+07	3.8994E+07	**1.3105E+08**	**4.2748E+07**
7_7_7_2_8	6.9937E+07	4.1060E+07	6.2431E+07	5.9149E+07	6.6418E+07	4.7553E+07	**1.0428E+08**	**5.8581E+07**
8_8_10_3_10	5.7888E+07	4.7935E+07	9.0396E+07	5.3134E+07	**1.5812E+08**	**1.0170E+08**	1.0111E+08	5.1122E+07
10_10_10_4_10	6.2506E+07	5.3940E+07	**1.0709E+08**	**7.3004E+07**	8.1458E+07	6.0468E+07	9.3790E+07	6.3963E+07
4_8_8_3_10	7.4331E+07	4.6146E+07	9.3093E+07	9.2035E+07	1.0854E+08	8.3137E+07	**1.4590E+08**	**7.9726E+07**
8_4_8_3_10	6.7171E+07	4.2849E+07	5.9053E+07	6.4040E+07	9.8800E+07	7.3302E+07	**1.1380E+08**	**7.1836E+07**
Average	6.1027E+07	3.9904E+07	7.5141E+07	5.7788E+07	8.9508E+07	6.0534E+07	**1.0039E+08**	**5.2695E+07**

$$C(A,B) = \frac{|b \in B | \exists a \in A, a \prec b|}{|B|} \tag{15}$$

$$SC(A,B) = \frac{C(A,B)}{C(A,B) + C(B,A)}, \tag{16}$$

where A, B represents the frontier set obtained by two different algorithms, $C(A,B)$ represents the proportion of individuals in set B dominated by individuals in set A, $SC(A,B)$ represents the percentage of individuals in set B dominated by individuals in set A in the total dominated proportion of the two algorithms. The closer the SC metric to 1, the better the former algorithm is. While the closer the SC metric to 0, the better the latter algorithm is. As we can see from Table VI, the proposed algorithm is far better than DNSGA-II algorithm and SGEA algorithm, with an average SC of more than 0.9. Similarly, the DMOAWPSO algorithm is far better than DNSGA-II algorithm and slightly better than SGEA algorithm. At the same time, the proposed algorithm is also better than the DMOAWPSO algorithm. With the increase of scale and change frequency, the overall trend of SC metric is also increasing. This shows that the proposed algorithm can track the changing frontier more effectively

TABLE X COMPARISON OF SC COVERAGE OF TWO EPISODES

Instance	Proposed vs DNSGA-II	Proposed vs SGEA	Proposed vs DMOAWPSO	DMOAWPSO vs DNSGA-III	DMOAWPSO vs SGEA	SGEA vs DNSGA-II
5_5_5_0_0	0.9843	0.9472	0.5863	0.9110	0.7580	0.6311
5_5_6_1_5	1.0000	0.9112	0.6172	0.9974	0.6725	0.6866
5_5_7_2_5	1.0000	0.9414	0.5993	0.9856	0.6932	0.6565
6_6_7_3_8	1.0000	0.9564	0.5216	0.9813	0.7871	0.6157
7_7_7_2_8	0.9567	0.9365	0.6363	0.8776	0.8269	0.5133
8_8_10_3_10	1.0000	1.0000	0.6392	0.9955	0.8701	0.8125
10_10_10_4_10	1.0000	0.9705	0.7418	0.9258	0.7066	0.7798
4_8_8_3_10	1.0000	0.9460	0.7316	0.8828	0.6664	0.7310
8_4_8_3_10	1.0000	0.9271	0.6507	0.8897	0.7597	0.6255
Average	0.9934	0.9485	0.6360	0.9385	0.7489	0.6724

than the DMOAWPSO algorithm in complex and changeable environments. The quality of the solution set is higher.

VI. CONCLUSION

In order to allocate jamming and attack UAVs resources to different targets with lowest cost in dynamic confrontation scenarios, this paper applies dynamic DT to establish a multi-objective task assignment optimization model. Aiming at the above problem, an improved dynamic multi-objective adaptive weighted particle swarm optimization algorithm is proposed. The proposed environmental change detection and response strategy can ensure the convergence and diversity of the population in the new environment. The proposed algorithm is compared with three classic and popular algorithms. The NPS, HV, PD and SC results verify that the proposed algorithm has better convergence, diversity and higher decision quality in most test instances.

The model and algorithm proposed in this paper can be used not only in the field of task assignment, but also in the field of the next-generation communication, air-assisted vehicle network, epidemic prevention and control, path planning and so on. Besides, the proposed algorithm has strong scalability and can be used in combination with other algorithms and techniques, such as clustering algorithm, deep learning algorithm and so on. In future work, researches on more exact optimization indicators and constraints processing with DT are worth to focus on to adapt to more complex dynamic scenarios. At the same time, the issues of data transmission delay and effective transmission rate cannot be ignored [56].

REFERENCES

[1] J. Zhang, and J. Xing, "Cooperative task assignment of multi-UAV system," *Chin J. Aeronaut.*, vol. 33, no. 11, pp. 2825-2827, Mar. 2020.

[2] N. Dilshad, J. Hwang, J. Song, and N. Sung, "Applications and challenges in video surveillance via drone: a brief survey," *2020 Int. Conf. Inform. Commun. Technol. Converg. (ICTC)*, 2020, pp. 728-732.

[3] Q. Wang, X. Li, S. Bhatia, Y. Liu, L. T. Alex, S. A. Khowaja, and V. G. Menon, "UAV-enabled non-orthogonal multiple access networks for ground-air-ground communications," *IEEE Trans. Green. Commun. Netw.*, vol. 6, no. 3, pp. 1340-1354, Sept. 2022.

[4] Q. Deng, J. Yu, and N. Wang, "Cooperative task assignment of multiple heterogeneous unmanned aerial vehicles using a modified genetic algorithm with multi-type genes," *Chin J. Aeronaut.*, vol. 26, no. 5, pp. 1238-1250, Oct. 2013.

[5] Y. Yang, Y. Zhang, L. Dai, J. Li, S. Mumtaz, and J. Rodriguez, "Transmission capacity analysis of relay-assisted device-to-device overlay/underlay communication," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 380-389, Feb. 2017.

[6] M. K. Afzal, Y. B. Zikria, S. Mumtaz, A. Rayes, A. Al-Dulaimi, and M. Guizani, "Unlocking 5G spectrum potential for intelligent IoT: opportunities, challenges, and solutions," *IEEE Commun. Mag.*, vol. 56, no. 10, pp. 92-93, Oct. 2018.

[7] H. Duan, J. Zhao, Y. Deng, Y. Shi, and X. Ding, "Dynamic discrete pigeon-inspired optimization for multi-UAV cooperative search-attack mission planning," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 1, pp. 706-720, Feb. 2021.

[8] Y. Chen, D. Yang, and J. Yu, "Multi-UAV task assignment with parameter and time-sensitive uncertainties using modified two-part wolf pack search algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 6, pp. 2853-2872, Dec. 2018.

[9] Z. Lv, D. Chen, H. Feng, H. Zhu, and H. Lv, "Digital twins in unmanned aerial vehicles for rapid medical resource delivery in epidemics," *IEEE Trans. Intell. Transp. Syst.*, pp. 1-9, Sept. 2021.

[10] W. Sun, P. Wang, N. Xu, G. Wang, and Y. Zhang, "Dynamic digital twin and distributed incentives for resource allocation in aerial-assisted internet of vehicles," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5839-5852, Apr, 2022.

[11] Y. Cui, W. Dong, D. Hu, and H. Liu, "The application of improved harmony search algorithm to multi-UAV task assignment," *Electronics*, vol. 11, no. 8, pp. 1043-1171, Apr. 2022.

[12] F. Ye, C. Fei, and L. Gao, "Multiobjective cognitive cooperative jamming decision-making method based on Tabu search-artificial bee colony algorithm," *Intl J. Aerospace. Eng.*, vol. 2018, pp. 8-18, Dec. 2018.

[13] J. Chen, X. Qing, F. Ye, K. Xiao, K. You, and Q. Sun, "Consensus-based bundle algorithm with local replanning for heterogeneous multi-UAV system in the time-sensitive and dynamic environment," *J. Supercomput.*, vol. 78, no. 2, pp. 1712-1740, Feb. 2022.

[14] Q. Peng, H. Wu, and N. Li, "Modeling and solving the dynamic task allocation problem of heterogeneous UAV swarm in unknown environment," *Complexity*, vol. 2022, pp. 5-19, Apr. 2022.

[15] H. Zhang, G. Zhang and Q. Yan, "Dynamic resource allocation optimization for digital twin-driven smart shopfloor," *2018 IEEE 15th Int. Conf. Netw., Sensing. Control. (ICNSC), Zhuhai, China*, May. 2018, pp. 1-5.

[16] G. N. Schroeder, C. Steinmetz, C. E. Pereira, and D. B. Espindola, "Digital twin data modeling with automationml and a communication methodology for data exchange," *IFAC-PapersOnLine*, vol. 49, no. 30, pp. 12-17, Nov. 2016.

[17] A. Jee and S. Prakriya, "Performance of Energy and Spectrally Efficient AF Relay-Aided Incremental CDRT NOMA Based IoT Network with Imperfect SIC for Smart Cities," *IEEE Internet Things J.*, pp. 1-17, Dec. 2022.

[18] Y. Sun, J. Xu, H. Wu, G. Lin and S. Mumtaz, "Deep Learning Based Semi-Supervised Control for Vertical Security of Maglev Vehicle With Guaranteed Bounded Airgap," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7,pp. 4431-4442, Jan. 2021.

[19] M. K. Afzal, Y. B. Zikria, S. Mumtaz, A. Rayes, A. Al-Dulaimi and M. Guizani, "Unlocking 5G Spectrum Potential for Intelligent IoT: Oppor-

tunities, Challenges, and Solutions," *IEEE Commun. Mag.*, vol. 56, no. 10, pp. 92-93, Oct. 2018.

[20] Q. Wang, X. Li, S. Bhatia, Y. Liu, L. T. Alex, S. A. Khowaja, and V. G. Menon, "UAV-Enabled Non-Orthogonal Multiple Access Networks for Ground-Air-Ground Communications," *IEEE Trans. Green. Commun. Netw.*, vol. 6, no. 3, pp. 1340-1354, Sept. 2022.

[21] Y. Yang, Y. Zhang, L. Dai, J. Li, S. Mumtaz and J. Rodriguez, "Transmission Capacity Analysis of Relay-Assisted Device-to-Device Overlay/Underlay Communication," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 380-389, Feb. 2017.

[22] S. Lin, B. Zheng, G. C. Alexandropoulos, M. Wen, and F. Chen, "Adaptive Transmission for Reconfigurable Intelligent Surface-Assisted OFDM Wireless Communications," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2653-2665, Nov. 2020.

[23] C. Ekelin, and J. Jonsson, "Solving embedded system scheduling problems using constraint programming," *IEEE Real-Time Syst. Sympos.*, 2000, pp. 27-45.

[24] M. Radmanesh, M. Kumar, A. Nemati, and M. Sarim, "Dynamic optimal UAV trajectory planning in the national airspace system via mixed integer linear programming," *Proc. Inst. Mech. Eng., Part G: J. Aerosp. Eng.*, vol. 230, no. 9, pp. 1668-1682, Sept. 2015.

[25] G. M. Levchuk, Y. N. Levchuk, K. R. Pattipati, and D. L. Kleinman, "Mapping flows onto networks to optimize organizational processes," *Sess. Adapt. Archit. Command. Control. (A2C2)*, 2005, pp. 14-39.

[26] T. Shima, S. J. Rasmussen, and A. G. Sparks, "UAV cooperative multiple task assignments using genetic algorithms," *Proc. 2005, Amer. Control Conf., 2005*, 2005, pp. 2989-2994, vol. 5.

[27] F. Wang, H. Zhang, M. Han, and L. Xing, "Co-evolution-based mixed-variable multi-objective particle swarm optimization algorithm to solve the cooperative multi-task assignment problem of unmanned aerial vehicles," *Chin J. Comput.*, vol. 44, no. 10, Oct. 2021.

[28] Y. Wei, B. Wang, W. Liu, and L. Zhang, "Hierarchical task assignment of multiple UAVs with improved firefly algorithm based on simulated annealing mechanism,"*2021 40th Chin. Control. Conf. (CCC)*, 2021, pp. 1943-1948.

[29] M. Deng, Z. Wu, and Y. Chen, "Interference resource scheduling algorithm based on potential game for UAV cooperative combat," *2021 IEEE 21st Intl. Conf. Commun. Technol. (ICCT)*, 2021, pp. 982-987.

[30] Y. Sun, J. Xu, H. Wu, G. Lin, and S. Mumtaz, "Deep learning based semi-supervised control for vertical security of maglev vehicle with guaranteed bounded airgap," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4431-4442, Jul. 2021.

[31] J. Wang, G. Jia, J. Lin, and Z. Hou, "Cooperative task allocation for heterogeneous multi-UAV using multi-objective optimization algorithm," *J Cent. South. Univ.*, vol. 27, no. 2, pp. 432-448, Apr. 2020.

[32] C. R. Atencia, D. S. Javier, and D. Camacho, "Weighted strategies to guide a multi-objective evolutionary algorithm for multi-UAV mission planning,"*Swarm. Evol. Comput.*, vol. 44, pp. 480-495, Feb. 2019.

[33] 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.

[34] G. Sun, J. Li, Y. Liu, S. Liang and H. Kang, "Time and Energy Minimization Communications Based on Collaborative Beamforming for UAV Networks: A Multi-Objective Optimization Method," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3555-3572, Nov. 2021.

[35] G. Yu, Y. Jiang, L. Xu and G. Y. Li, "Multi-Objective Energy-Efficient Resource Allocation for Multi-RAT Heterogeneous Networks," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 10, pp. 2118-2127, Oct. 2015.

[36] S. Sahmoud, and H. R. Topcuoglu, "Sensor-based change detection schemes for dynamic multi-objective optimization problems," *2016 IEEE Symp. Series. Comput. Intell. (SSCI)*, 2016, pp. 1-8.

[37] H. Richter, "Detecting change in dynamic fitness landscapes," *2009 IEEE Congr. Evol. Comput.*, 2009, pp. 1613-1620.

[38] S. Jiang, and S. Yang, "A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 1, pp. 65-82, Feb. 2017.

[39] R. Shang, L. Jiao, Y. Ren, L. Li, and L. Wang, "Quantum immune clonal coevolutionary algorithm for dynamic multiobjective optimization," *Soft Comput.*, vol. 18, no. 4, pp. 743-756, Jul. 2013.

[40] M. Rong, D. Gong, Y. Zhang, Y. Jin, and W. Pedrycz, "Multidirectional prediction approach for dynamic multiobjective optimization problems," *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3362-3374, Sept. 2019.

[41] S. Sahmoud, and H. R. Topcuoglu, "A memory-based NSGA-II algorithm for dynamic multi-objective optimization problems," *Europ. Conf. Appl. Evol. Comput.*, 2016, pp. 296-310.

[42] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, and X. Liu, "A novel sigmoid-function-based adaptive weighted particle swarm optimizer," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 1085-1093, Feb. 2021.

[43] X. Fu, P. Feng, and X. Gao, "Swarm UAVs task and resource dynamic assignment algorithm based on task sequence mechanism," *IEEE Access*, vol. 7, pp. 41090-41100, Mar. 2019.

[44] J. Ruiye, C. Tao, W. Songyan, and Y. Ming, "A modified whale optimization algorithm based on chaos initialization and regulation operation," *2019 Chin. Control. Conf. (CCC)*, 2019, pp. 2702-2707.

[45] S. Yang, Y. Shao, and K. Zhang, "An effective method for solving multiple travelling salesman problem based on NSGA-II," *Syst. Sci. Control. Eng.*, vol. 7, no. 2, pp. 108-116, Oct. 2019.

[46] Y. Yang, T. Zhang, W. Yi, L. Kong, X. Li, B. Wang, and X. Yang, "Deployment of multistatic radar system using multiobjective particle swarm optimisation," *IET. Radar. Sonar. Navig.*, vol. 12, no. 5, pp. 485-493, Jan. 2018.

[47] C. Chen, and L. Y. Tseng, "An improved version of the multiple trajectory search for real value multi-objective optimization problems," *Eng. Optimiz.*, vol. 46, no. 10, pp. 1430-1445, Nov. 2013.

[48] R. Liu, J. Li, J. Liu, and L. Jiao, "Review of dynamic multi-objective optimization research," *J. Comput. Sci.*, vol. 43, no. 7, pp. 1246-1278, Jul. 2020.

[49] L. Wang, Y. Ren, Q. Qiu, and F. Qiu, "Review of research on performance evaluation metrics of multi-objective evolutionary algorithms," *Chin J. Comput.*, vol. 44, no. 8, pp. 31-60, Aug. 2021.

[50] W. Hu, G. Gary, X. Zhang, "Multiobjective Particle Swarm Optimization Based on Pareto Entropy," *Ruan Jian Xue Bao/ J. Softw.*, vol. 25, no. 5, pp. 1025-1050, Sept. 2014.

[51] A. Azadeh, M. Ravanbakhsh, M. Rezaei-Malek, M. Sheikhalishahi, and A. Taheri-Moghaddam, "Unique NSGA-II and MOPSO algorithms for improved dynamic cellular manufacturing systems considering human factors," *Appl. Math. Model.*, vol. 48,pp. 655-672, Aug. 2017.

[52] H. Wang, Y. Jin, and X. Yao, "Diversity assessment in many-objective optimization," *IEEE Trans. Cybern.*, vol. 47, no. 6, pp. 1510-1522, Jun. 2017.

[53] E. Zitzler, and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257-271, Nov. 1999.

[54] F. Habibi, F. Barzinpour, and S. J. Sadjadi, "A multi-objective optimization model for project scheduling with time-varying resource requirements and capacities," *J. Ind. Syst. Eng.*, vol. 10, pp. 92-118, Nov. 2017.

[55] M. Li, L. Zhen, and X. Yao, "How to read many-objective solution sets in parallel coordinates [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 88-100, Nov. 2017.

[56] S. Lin, B. Zheng, G. C. Alexandropoulos, M. Wen, F. Chen, and S. sMumtaz, "Adaptive transmission for reconfigurable intelligent surface-assisted OFDM wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2653-2665, Nov. 2020.