Cybernetics and Intelligence
Research Article

Research on multi-objective intelligent unmanned operation scheduling method based on two-layer PSO

Haixu Li[1,2], Tao Zhang[1], Guang Liu[2], Jian Yin[3,4], Zhan Shi[3,4], Liran Shen[3,4], Yunsheng Fan[3,4,✉]

Cite this article: Li HX, Zhang T, Liu G, et al. *Cybernetics and Intelligence* 2026, **1**(2): 9390011. https://doi.org/10.26599/CAI.2025.9390011

ABSTRACT: This paper investigates the multi-objective scheduling problem of intelligent unmanned operations and analyzes the interdependent constraints among the various links of the operation workflow. To address the coupled challenges of job sequencing and resource allocation inherent in unmanned operation scenarios, an integrated scheduling model based on a two-layer particle swarm optimization framework is proposed. A mixed-integer programming formulation is adopted to rigorously characterize the structural constraints and logical dependencies within the scheduling process. Building upon this model, an enhanced two-layer particle swarm optimization algorithm with fragment-based particle encoding is introduced to expand the feasible search space. Moreover, a dynamic inertia weight adjustment mechanism and an adaptive mutation operator are incorporated to strengthen the algorithm's global exploration capability while accelerating convergence. Simulation experiments verify that the proposed model and algorithm effectively optimize the scheduling of unmanned operations under complex operational constraints, significantly improving system-level support performance. These results demonstrate that the method provides a robust and efficient solution for multi-objective intelligent scheduling tasks in highly constrained unmanned operation environments.

KEYWORDS: intelligent unmanned operation; support scheduling; mixed-integer programming; heuristic algorithms

1 Introduction

Multi-objective intelligent unmanned job scheduling is an inevitable trend to solve the bottleneck of large manpower demand, complex resource scheduling and guarantee efficiency improvement in current complex operation scenarios. In the mode of command and support by a large number of personnel, multi-objective scheduling needs to consider the understandability of the plan, the convenience of collaboration, and the regularity of the operation, which cannot realize the optimal scheduling. In the intelligent unmanned operation mode, through the optimization scheduling algorithm, the job timing and resource allocation elements can be optimized to the greatest extent, without considering the constraints of personnel factors.

For the scheduling problem of multi-objective group work in manned mode, domestic scholars have proposed a variety of algorithm models, and used heuristic algorithms, swarm intelligence optimization and other methods to explore the optimal scheduling scheme. These researches aim to arrange time and resources reasonably, and reduce task delay and resource waste to the greatest extent while ensuring the successful completion of tasks, so as to improve operation efficiency and accuracy.

(1) Aspects of solving pre-optimization and scheduling problems: Wang et al. [1] designed an improved discrete particle swarm optimization (IDPSO) algorithm to solve the complex scheduling problem of multiple supply points and multiple demand points involved in the support process. The experimental results show that IDPSO algorithm has faster convergence speed and stronger global optimization ability. Zhang et al. [2] abstracted the complex guaranteed job scheduling problem into a flexible flow shop scheduling problem with workpiece due date constraints. A greedy local search genetic algorithm with dual-level coding is proposed. GLSGA-DC is used to solve the support scheduling model with the objective of minimizing the completion time of support jobs. The simulation results show that the GLSGA-DC algorithm has excellent performance in the actual objects transfer scene. Lyu et al. [3] analyzed the constraint relationship between jobs and proposed an improved non-dominated sorting genetic algorithms (NSGA-II) algorithm for the support job problem, which can effectively reduce the completion time of support jobs. And it optimizes the job load distribution while ensuring the feasibility and rationality of job scheduling. Liu et al. [4] discussed the influence of fixed allocation of support resource stations on support operation efficiency and takeoff rate, and proposed a variable neighborhood search (VNS) to solve the bi-level optimization problem of the joint scheduling model of resource station allocation and pre-flight preparation scheduling. The experimental results show that the VNS algorithm shows superior performance in solving the optimization

[1] Department of Automation, Tsinghua University, Beijing 100084, China. [2] CSSC Systems Engineering Research Institute, Beijing 100036, China. [3] Marine Electrical Engineering College, Dalian Maritime University, Dalian 116026, China. [4] Liaoning Key Laboratory of Intelligent Ship Technology and Systems, Dalian 116026, China.

✉ Corresponding author. E-mail: yunsheng@dlmu.edu.cn

Received: June 27, 2025; Revised: December 5, 2025; Accepted: December 8, 2025

problem, and can effectively improve the overall efficiency of the preparation operation.

(2) Resource scheduling and conflict resolution: Zhu et al. [5] designed an improved gravitational search algorithm (IGSA) to effectively coordinate and guarantee operation procedures and personnel scheduling, combined with the series mechanism of resource conflict resolution strategy. It can effectively deal with the contention conflicts of guarantee resources in the scheduling process. On this basis, the performance indicators of support resources are regarded as decision variables satisfying interval constraints, and a bi-level optimization model [6] is constructed. Harris hawks optimization algorithm (HHO) is used to solve the optimization design problem of performance indicators. Simulation results show that the proposed method can effectively solve the joint optimization problem of one-stop support resource performance index and support job scheduling. Liu et al. [7] considered the scene characteristics of multiple lifts and multiple transporters, established a problem model of starting from multiple depots and transferring to multiple targets, and used the improved grey wolf optimizer (GWO) algorithm to optimize resource scheduling and path planning in support operations. The simulation results show that the improved GWO algorithm can significantly improve the efficiency of the support operation scheduling.

(3) Intelligent scheduling and safety guarantee: Jin et al. [8] constructed a serialization double-layer resource heap model that conforms to the characteristics of support operations, which can effectively represent the resource allocation and operation steps involved in the operation. And an intelligent planning method of the support operation based on formal verification is proposed to solve the safety and correctness problems in the current support operation scheduling. Guo et al. [9] focused on the problem of resource transportation support scheduling, and added the interference of layout on resource transportation path and the constraint of operation priority, and used the double population immune algorithm to improve the solution quality of the scheduling model. The experimental results show that the proposed algorithm can effectively respond to the resource support requirements of object under different task requirements, and significantly improve the efficiency of object resource transportation scheduling.

(4) Job matching and scheduling decisions: Liu [10] established a scheduling model considering the constraints of station, equipment, job priority and safety for the resource-constrained support operation scheduling problem with multi-resources and multi-constraints, and verified the practicability of the proposed model in the support operation scheduling problem through the simulation results of eight objects support in a single wave. Liu et al. [11] proposed an adaptive batch matching decision method for support jobs, which uses reinforcement learning to realize the optimal matching between support jobs and support bits. Simulation experiments show that this method can quickly generate high-quality matching schemes while meeting the real-time requirements.

(5) Virtual simulation and system display: Liu et al. [12] proposed a 3D virtual demonstration method of the support operation scheduling based on Unity3D on the basis of Ref. [7], and built a 3D virtual simulation system for the support operation scheduling, which can realize the virtual interactive scene of the support operation scheduling.

On the one hand, the existing research often optimizes the scheduling model for a single process or a specific scheduling scenario, lacking the overall consideration of the operation process, which is difficult to adapt to the comprehensive scheduling requirements of multiple operations. At the same time, many studies fail to fully consider the interaction between the complexity of deck layout and operational constraints, resulting in insufficient applicability and robustness in complex scenarios. On the other hand, the existing optimization algorithms [13] can solve the job scheduling problem in complex support scenarios to a certain extent, but there are still obvious shortcomings in rapid response to dynamic changes and dealing with multi-job resource conflicts, which limits their applicability in actual jobs.

In order to more clearly show the advantages and disadvantages of the existing scheduling job research and the comparison with the improved algorithm in this paper, Table 1 summarizes the key characteristics, advantages and limitations of the existing scheduling job research, and highlights the significant advantages of the improved PSO algorithm in the solution space expansion, dynamic adjustment ability and global search ability.

First, for target operations, an integrated approach recovery and support scheduling model was constructed based on the constraint relationships among various operational phases.

Second, to address the insufficient solution space diversity of the traditional particle swarm optimization (PSO) algorithm in target support scheduling problems, an improved PSO (IPSO) algorithm based on particle fragment encoding was proposed. By employing a double-layer fragment encoding structure to expand the feasible solution domain and incorporating dynamic inertia weight adjustment and adaptive mutation operators, the algorithm's global search capability and convergence efficiency were significantly enhanced.

Finally, a case study on support scheduling was conducted based on the deck layout of a certain foreign ship, validating the

Table 1 Comparison table of existing scheduling job studies

Algorithm	Scenario	Highlight
IDPSO [1]	Scheduling of multiple points and multiple request points in support	Fast convergence speed, strong global optimization capability
GLSGA-DC [2]	Minimizing the completion time of support operations	Excellent performance in practical transportation scenarios
NSGA-II [3]	Scheduling of loading and support operations	Reduces completion time and optimizes load distribution
VNS [4]	Joint optimization of resource station configuration and pre-flight preparation scheduling	Improves the overall efficiency of pre-flight preparation operations
IGSA [5]	Scheduling of support operation processes and personnel with resource conflicts	Effectively resolves resource competition conflicts
GWO [7]	Scheduling of support operations involving multiple elevators and transport vehicles	Significantly improves scheduling efficiency
Proposed algorithm	Scheduling of object approach recovery and support operations	Double-layer fragment encoding expands solution space, dynamically adjusts inertia weight and mutation operator, strong global search capability

effectiveness and robustness of the proposed model and algorithm under complex constraints. This provides an efficient optimization method for multi-task, multi-objects scheduling scenarios.

2 Problem description and modeling analysis

2.1 Problem description

Multi-objective scheduling job tasks exist in a variety of scenarios, such as shop scheduling, cluster scheduling, etc., generally involving multi-objective entry into the specified area, transfer, multi-link guarantee work, exit the specified area, etc. The operation process involves complex scheduling processes such as resource allocation, timing allocation, space allocation, etc. For multi-person collaborative work, it is also necessary to consider the coordination of personnel. A typical operation scenario is shown in Fig. 1.

According to the research in Ref. [2], the multi-objective scheduling operation task can be abstracted as hybrid flow-shop scheduling problem (HFSP), i.e., the problem of scheduling multiple guaranteed service stations. The multi-objective scheduling operation task can be abstracted as a closed-loop queuing network, in which each operation link can be regarded as a service station, and scheduling can be regarded as a cyclic transfer process of objects in this network.

The intelligent unmanned operation scheduling studied in this paper is different from the traditional manned mode security operation scheduling, which is mainly reflected in the following: intelligent unmanned operation scheduling needs to pursue multi-objective completion of the overall task optimization, solve the resource allocation and scheduling in different operation links, and does not have to consider the personnel operation constraints in the operation task.

2.2 Scheduling model

Since the coordination and resource allocation among various operations have an important impact on the smooth operation of tasks, how to optimize the scheduling process under the constraints of limited resources and time in order to improve operational efficiency and reduce resource waste has become a key problem to be solved. The optimal scheduling of such problems involves complex constraint relationships and multi-dimensional resource allocation, which requires the consideration of the mutual influence and synergy effect between various stages in the whole process scheduling.

In order to be able to better construct a multi-objective guaranteed job scheduling optimization model, the following assumptions are made before establishing the model:

S1: The operation is non-preemptive, i.e., non-interruptible.

S2: The target transit process is carried out according to the path library, i.e., the glide time is mainly determined by the parking point and the selected safeguard position.

S3: The target entry area mission process only considers a single unsuccessful re-entry, and after entry the transshipment to the corresponding safeguard position.

S4: The multi-target array information and safeguard position information are known, and the safeguard resource type of the safeguard position is limited but the resource quantity is not limited.

The essence of the multi-objective intelligent unmanned scheduling optimization problem is to pursue the safeguard operation efficiency, and its model can be described as follows: the objective cluster $I=\{1,2,\ldots,n\}$ to be safeguarded, under the premise of satisfying various types of constraints, enters the region sequentially in the order of minimizing the objective function. In Eq. (1), T_i^m denotes the entry time of objective i in the m-th scheduling, and $\max(T_i^m)$ denotes the latest entry time among all objectives, i.e., the optimization objective is to minimize the safeguard operation time.

$$\min(\max(T_i^m)) \tag{1}$$

The corresponding constraints are as follows:

(1) The wake interval time ΔTs_{ij} between two targets is constrained by Eq. (2). Among them, $\forall i,i\in I$, $m=0,1,\ldots,N$, T_j^m represents the time when target j enters completion in the m-th scheduling, T_i^m represents the completion time of target i enters completion during the m-th scheduling, and ΔTs_{ij} represents the wake interval time between i and j.

$$T_j^m \geq \Delta Ts_{ij}+T_i^m \tag{2}$$

(2) The post-entry transit preparation time ΔTc constraint is expressed as Eq. (3), during which each transit preparation is completed.

$$T_j^m \geq \Delta Tc+T_i^m \tag{3}$$

(3) For targets where there is an entry failure, there may be a conflict with a subsequent target, in which case the timing needs to be adjusted to resolve the conflict, where $(T_i^m)_{\text{landing}}$ denotes the current entry time of target i, and ΔTr represents the minimum adjustment time required for objective i.

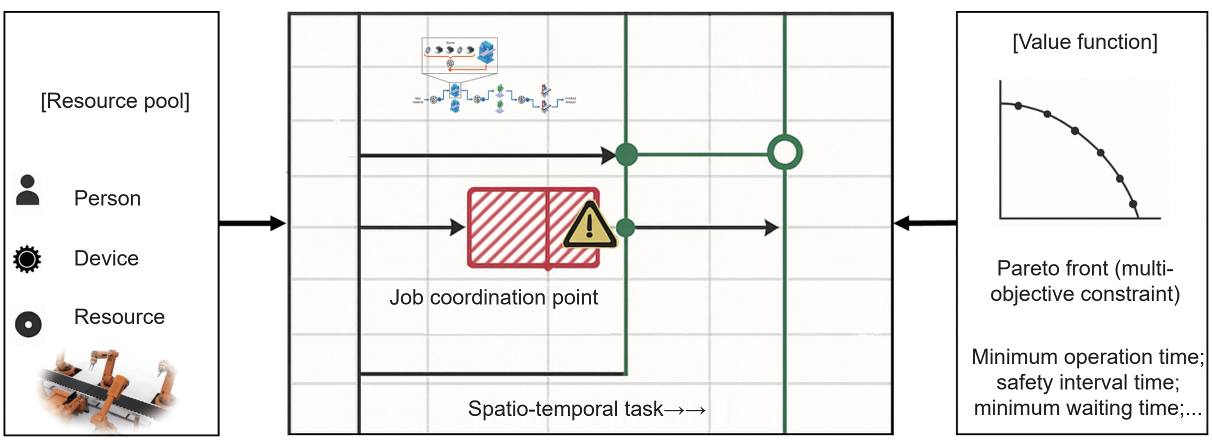

Fig. 1 Schematic diagram of the multi-objective integrated scheduling problem scenario.

$$T_i^m \geq \Delta Tr + (T_{i-1}^m)_{\text{landing}} \tag{4}$$

(4) The sequence of target entry operations should follow a certain logical order. During each scheduling process, it must be ensured that the sequence is realistic. Set the order constraint as Eq. (5). That is, if target j enters before target i, then T_j^m must be greater than T_i^m.

$$T_j^m > T_i^m \tag{5}$$

(5) The convention is that at most one target can be selected for safeguard operations in each scheduling cycle. I_i^m is a binary decision variable indicating whether target i is selected in the m-th scheduling cycle. In Eq. (6) if $I_i^m = 1$, then target i enters in that cycle m, and if $I_i^m = 0$, then target i does not enter in cycle m.

$$\sum_{i=1}^{n} I_i^m = 1,\ m = 1, 2, \ldots, M \tag{6}$$

On the other hand, the core of safeguard operation scheduling optimization is to optimize the sequence of safeguard operations, improve the safeguard efficiency, and reduce the total completion time under the premise of ensuring the successful completion of safeguard tasks. After the target enters the operation area, according to the safeguard resource demand for the safeguard position scheduling, at the same time transfer the target to the corresponding safeguard position for resource safeguard. The minimization of the maximum completion time of the problem is the objective function to be solved:

$$\min(\max_i(t_i)) \tag{7}$$

where t_i represents the time for target i to complete the support operation. The corresponding constraints are as follows:

(1) Each target must be assigned a support bit in the support process, and each support bit can only serve one target at the same time. This indicates that each target i must be assigned to a certain safeguard bit.

$$\sum_{j=1}^{n} x_{ij} = 1 \tag{8}$$

(2) The target must be transferred to the support position before the support operation can begin. The start time of support operation is related to the transfer completion time, and the start support operation time t_i of target i should not be less than its transfer completion time $w_i + q_i$.

$$t_i \geq w_i + q_i \tag{9}$$

(3) A maximum of four targets can carry out the transfer operation at the same time. Thus, for each time point t, a maximum of four targets are allowed to transship simultaneously.

$$\sum_{i=1}^{n} (w_i \leq t \leq w_i + q_i) \leq 4 \tag{10}$$

(4) If the priority of target i is high $(c_i \leq c_j)$, the support operation of target i must be earlier than that of target j to ensure that the high priority task is processed in time, $\forall i, j \in \{1, 2, \ldots, n\}$.

$$t_i \leq t_j \tag{11}$$

(5) The time for target i to complete the support operation is its start support operation time plus the support operation duration p_i.

$$T_i^c = t_i + p_i \tag{12}$$

Considering the two aspects, the overall objective function of the scheduling model is as Eq. (13):

$$\min(\max(T_i^m)) + \min(\max_i(t_i)) \tag{13}$$

3 Improve the scheduling design of the heuristic algorithm

Particle swarm optimization (PSO) is a heuristic optimization algorithm that simulates the foraging behavior of a flock of birds, and is widely used in the fields of function optimization, combinatorial optimization, scheduling problems [1]. PSO algorithm approaches the optimal solution step by step by simulating the group collaboration and individual cognition of the particles in solution space, and continuously adjusting the position and velocity of the particles. However, the shipboard security scheduling problem usually contains complex constraints, and the direct adoption of the unimproved PSO algorithm may lead to a large number of infeasible solutions in the particle search process, which in turn affects the convergence speed and solving efficiency of the algorithm.

To solve this problem, this paper proposes an improved particle swarm optimization (PSO) algorithm, which extends the solution space through a two-layer fragment encoding structure and combines the inertia weight dynamic adjustment strategy and dynamic variation operator to significantly improve the global search capability and convergence efficiency of the algorithm. The algorithm is capable of generating a preliminary intelligent unmanned multi-objective safeguard scheduling scheme, and assigning the set scheduling tasks to the job sequences in the experimental part to ensure the timely start of the jobs and the efficient generation of high-quality feasible solutions.

3.1 Algorithm framework

This section presents a framework for the application of improved particle swarm optimization (PSO) algorithms to the intelligent unmanned cluster assurance scheduling problem, as shown in Fig. 2, and the specific steps are as follows:

(1) Swarm initialization: the swarm size is set according to the preset constraints, and the initial particle swarm is randomly generated. Each particle represents a potential solution, and its position and velocity are randomly initialized to ensure the diversity of the initial population in the solution space, thus providing a broader search space for the subsequent evolutionary process.

(2) Particle update and mutation operation: in each iteration, particles are updated based on their fitness evaluations. Specifically, each particle is guided jointly by its personal best position (p^{best}) and the global best position (g^{best}). By integrating both cognitive and social components, the particle adjusts its search direction and iteratively updates its velocity and position to approach promising regions in the solution space. In addition, random perturbations are introduced to enhance population diversity and reduce the risk of premature convergence.

Furthermore, to strengthen local exploitation, a triggering probability is defined for activating the local search mechanism. Once triggered, neighborhood-based perturbation operators—such as swap and insertion—are applied to high-quality solutions. A modification is accepted only if it leads to an improvement in fitness, following an "improvement-only acceptance" criterion. This mechanism effectively enhances the depth of local search and

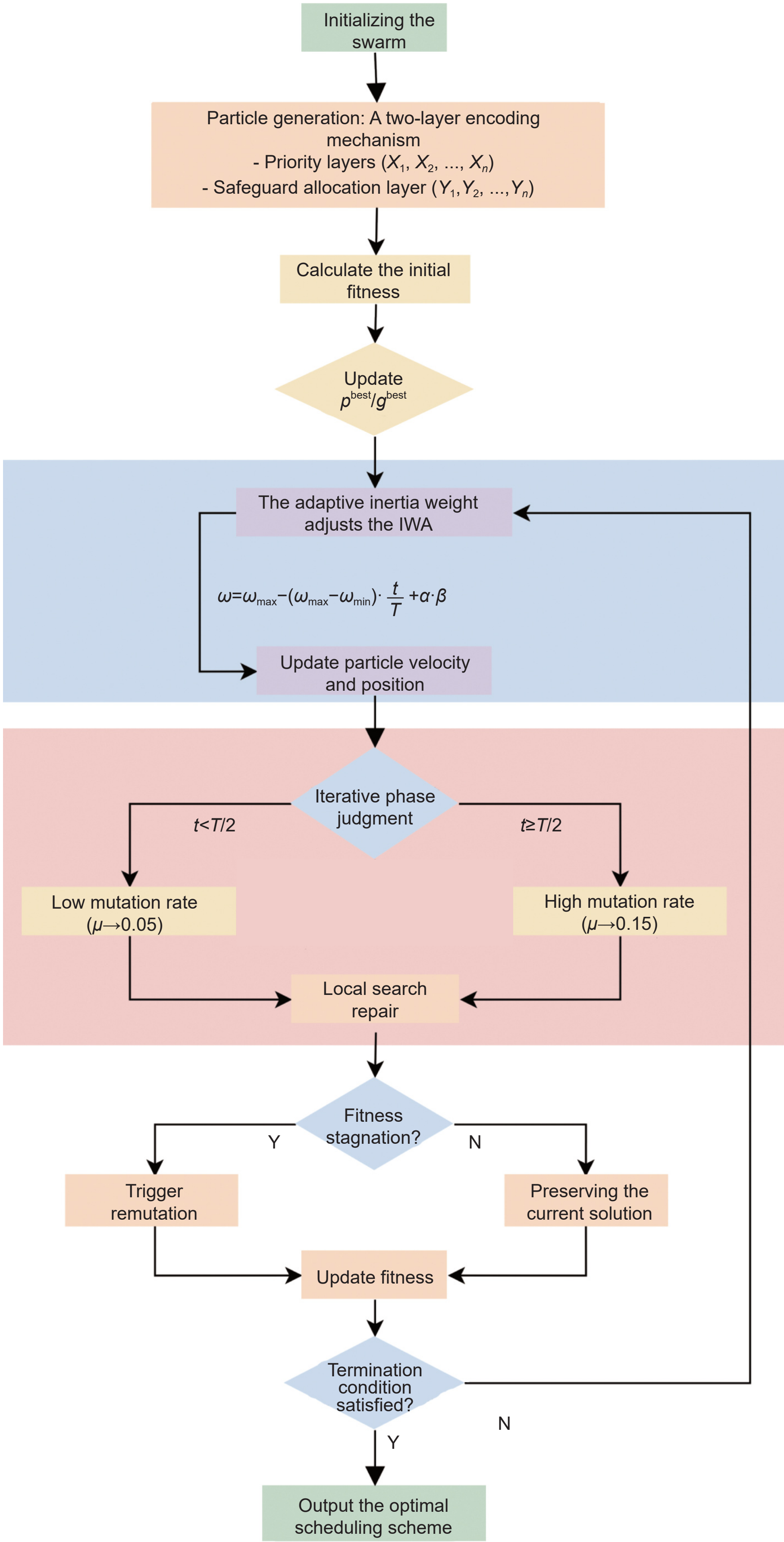

Fig. 2 Block diagram of the scheduling algorithm based on PSO optimization.

improves the overall optimization performance of the algorithm.

(3) Fitness evaluation: each particle is evaluated using the fitness function to quantify its performance in solving the cluster guaranteed scheduling problem. The level of the fitness value reflects the merit of the solution, the higher the fitness value, the better the solution optimizes the scheduling problem.

(4) Termination condition check: check whether the preset termination condition is satisfied, such as reaching the maximum number of iterations or finding a satisfactory optimal solution. If the termination condition is satisfied, the algorithm is terminated and the optimal solution is returned. Otherwise, the update and iteration of particles continue until the termination condition is satisfied.

3.2 Improved PSO algorithm

(1) Encoding

In multi-objective safeguard scheduling, different safeguard bit assignments and job priorities vary. In the coding stage, if the traditional process-based single-layer coding method is used, it may lead to too limited feasible solution space. For this reason, this paper proposes a two-layer fragment-based coding approach, which uses an integer coding method to divide each individual into a two-layer structure: the first layer of target entry region order and the second layer of safeguard bit allocation. The first layer is the prioritization of target entry, which is expressed as vector $\boldsymbol{X}=[X_1,X_2,\ldots,X_n]$, and each X_i represents the landing sequence of the i-th target. The second layer is the support position allocation, which is represented by vector $\boldsymbol{Y}=[Y_1,Y_2,\ldots,Y_n]$. Each Y_i represents the support position of the i-th target, where n is the number of scheduled targets.

For example, as shown in Fig. 3, the first eight codes of the first layer represent the processing order of jobs 6, 5, 8, 2, 1, 4, 7, and 3. The first eight codes of the second layer represent the corresponding machines; job 6 will be preferentially processed on the second machine, and job 4 will be processed on the second machine after job 6 is processed. In order to avoid falling into the infeasible solution region, the initial population is randomly generated according to the corresponding constraints in the model.

(2) Fitness function

The criterion for evaluating the scheduling program is the size of the adaptability value, combined with the guaranteed job scheduling model established in the previous section, the adaptability function is set to the guaranteed job completion time according to the objective function.

(3) Update

The velocity and position update formula of the traditional particle swarm optimization algorithm is as Eq. (14), where t represents the current iteration number; $v_{i,j}^{(t)}$ represents the velocity component of the i-th particle in the j-th dimension in the t-th iteration. $x_{i,j}^{(t)}$ represents the position component of the i-th particle in the j-th dimension in the t-th iteration. Let ω denote the inertia weight coefficient; c_1 and c_2 represent the learning factors, usually taking values in [0, 2]. r_1 and r_2 are random numbers in [0, 1]. $p_{i,j}^{\text{best}}$ means the best position of the i-th particle so far. g_j^{best} denotes the global best position up to iteration t.

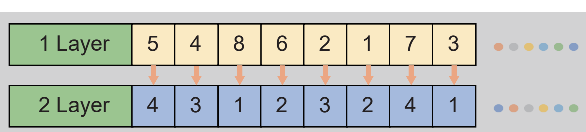

Fig. 3 Schematic representation of the corresponding two-layer encoding.

$$\begin{cases} v_{i,j}^{(t+1)}=\omega v_{i,j}^{(t)}+c_1r_1\left(p_{i,j}^{\text{best}}-x_{i,j}^{(t)}\right)+c_2r_2\left(g_j^{\text{best}}-x_{i,j}^{(t)}\right) \\ x_{i,j}^{(t+1)}=x_{i,j}^{(t)}+v_{i,j}^{(t+1)} \end{cases} \tag{14}$$

For the actual situation of multi-objective safeguard operations, the introduction of inertia weight adjustment (IWA) strategy is considered to optimize and improve the particle swarm. The adjustment of inertia weight (ω) is regarded as a key factor in balancing exploration and exploitation in the optimization search process. In order to better adapt to the scheduling requirements of different tasks and different models in the scheduling problem, the inertia weights should take into account the priorities, type differences, and scheduling complexity of different target tasks.

The inertia weight adjustment strategy of each particle is as Eq. (15):

$$\omega_i(t)=\omega_{\min}+(\omega_{\max}-\omega_{\min})\cdot\frac{T_{\max}-t}{T_{\max}}\cdot\omega_{\text{task}}\cdot\omega_{\text{model}} \tag{15}$$

where $T_{\max}$ represents the maximum number of iterations, and ω_{task} is the adjustment factor of task priority, which indicates the complexity or importance of the task. For critical tasks such as approach recovery, ω_{task} can be set to a high value, while for support tasks, the value is relatively small. ω_{model} is an adjustment factor for the model to indicate the difference between different types of scheduling. ω_{model} for differences in the target type, adjustments can be made by the difference situation.

(4) Mutation operator

The introduction of mutation operator can enhance the global search ability of PSO algorithm, maintain the diversity of population, and assist the local search ability of particles. The variation factor used to introduce the disturbance is usually a small positive number.

$$x_{i,j}^{\prime(t+1)}=x_{i,j}^{(t+1)}+\delta\cdot(g_j^{\text{best}}-x_{i,j}^{t}) \tag{16}$$

PSO algorithm is prone to local optimum stagnation in the optimization process. In order to improve the global search effect, a lower mutation rate is set in the early iteration to facilitate the retention of good solutions. The mutation rate is increased in the later stage to avoid premature convergence of the population. At the same time, considering the local search mechanism, when the number of iterations of the solution reaches the set threshold and the fitness is not significantly improved, the remutation operation will be performed on the particle to enhance the ability of the algorithm to jump out of the local optimum.

4 Experimental verification

4.1 Simulation environment

In order to more intuitively simulate the process of intelligent unmanned cluster assurance operations, this paper builds a visual virtual simulation platform based on Unity3D, and the simulation environment is shown in Fig. 4. Through this platform, the takeoff operation process of object can be deeply discussed and optimized, and finally the combat capability of object can be improved.

4.2 Case study

A typical multi-target scenario is used as an example to test the scheduling algorithm for safeguard jobs. Considering the different operational task demand pressures in manned and unmanned modes to ensure that the robustness and reliability of the scheduling algorithms can be effectively verified, the only variable in the experiment is the number of targets that need to undergo

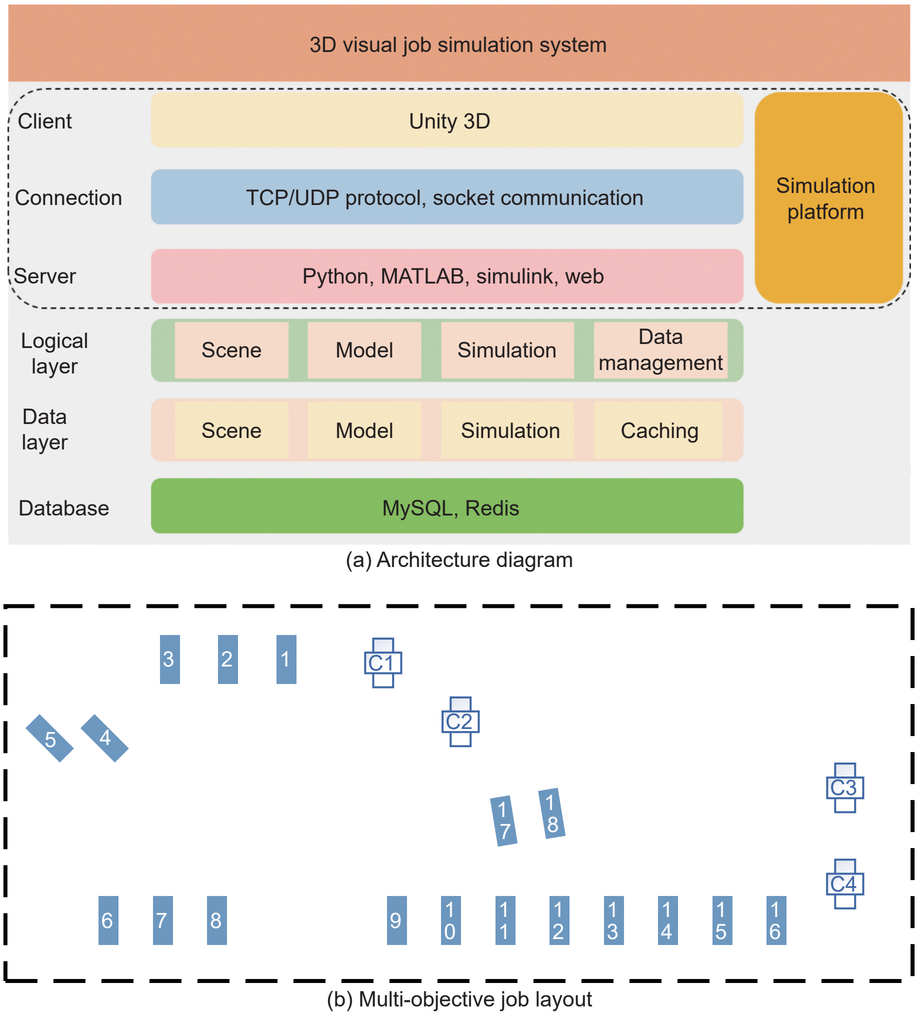

Fig. 4 Visual job simulation system.

job scheduling planning. The scheduling results are also visualized and analyzed using Gantt charts in order to visualize the operation time and sequence of each target. The experiments were run on a Win11 i5-8250U CPU @ 1.60 GHz MATLAB R2021a version, and the simulation parameters are given in Table 2.

The first set of experiments assumes that 15 targets need to be recovered and secured consecutively, and the scheduling system plans the scheduling of consecutive approach and securing operations for these 15 targets.

To clearly describe the applicability of the model to different problem sizes, this paper classifies the scheduling scale according to the number of targets: small-scale (⩽ 20 targets), medium-scale (20–30 targets), and large-scale (⩾ 30 targets). The 15-target scenario therefore represents a typical small-scale scheduling case.

In order to evaluate the processing capability and efficiency of the scheduling algorithm under a small number of targets. In this operation planning, the scheduling algorithm not only needs to reasonably arrange the target landing and transfer order, but also optimize the operation time of each target, so as to improve the overall operation efficiency and response speed.

Table 2 Experimental simulation parameters

Task	Configuration	Value
Enter	Time	60 s
Enter waiting	Time	30–60 s
Re-entry	Probability	0.05–0.07
Transfer	Time	90–180 s
Support position	Unit	45

The job scheduling simulation results are shown in Fig. 5. Dark blue means entering the waiting state, orange red means re-entering state, orange yellow means entering the waiting interval added because of re-entering, purple means entering the state, green means grounding operation, blue means entering the state of transshipping operation, and its number means the number of the assigned safeguard bit. It can be seen from the figure that the *K*-th target enters the state according to the given entry order. After completion, the subsequent entry operation of the (*K*+1)-th target is carried out. After successful entry, the *K*-th target needs to be immediately grounded and is pulled to the corresponding support position by the tractor. Due to the probability of unsuccessful entry, re-entry occurs at the 12th target, which leads to waiting intervals for the following three targets.

The second set of experiments is a medium-scale continuous operation task scenario, where 25 targets are set to continuously enter the working condition, as shown in Fig. 6. The third set of experiments was conducted for a large-scale continuous operation task scenario, setting up 35 targets to enter the working conditions consecutively, as shown in Fig. 7. The entry unsuccessful processing strategy proposed in this paper is effectively verified in the simulation: when a re-flight situation occurs in a target, the system automatically triggers re-entry, rejoins the target into the queue, and makes re-entry attempts, and at the same time the order of the subsequent targets is deferred sequentially to ensure

Tsinghua University Press | Available on IEEE Xplore

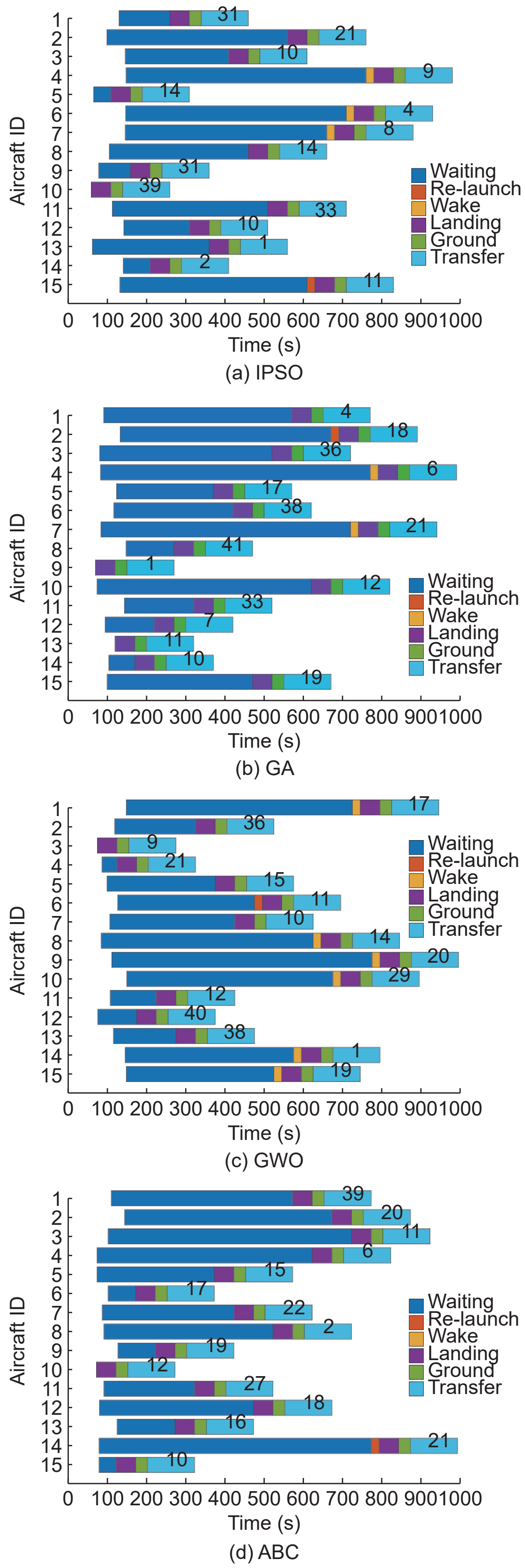

Fig. 5 Gantt chart of landing support operations for 15 targets.

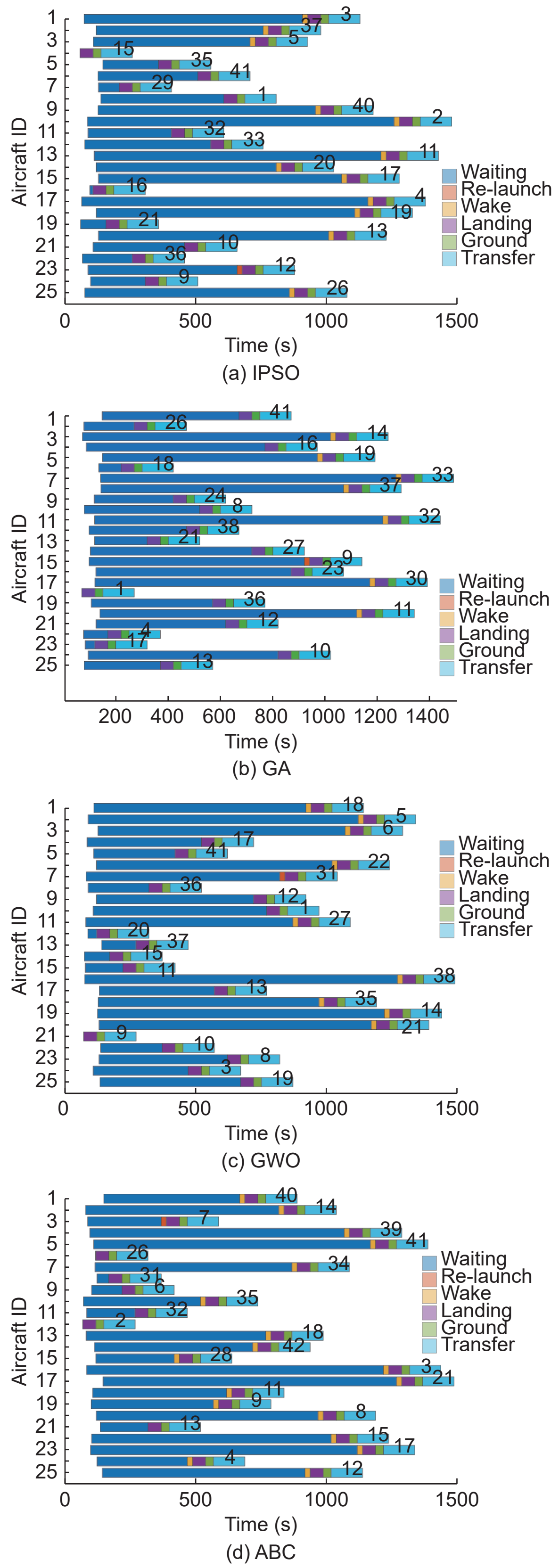

Fig. 6 Gantt chart of landing support operations for 25 targets.

the continuity and stability of the task scheduling. Through the Gantt chart, you can clearly observe the complete operation flow of the 35 targets, including the time allocation and resource scheduling of each key link.

It can be found through the analysis of the results of the two sets of simulation experiments, as shown in Table 3. In small-scale problems, the scheduling model based on the improved particle swarm algorithm proposed in this paper has similar solution

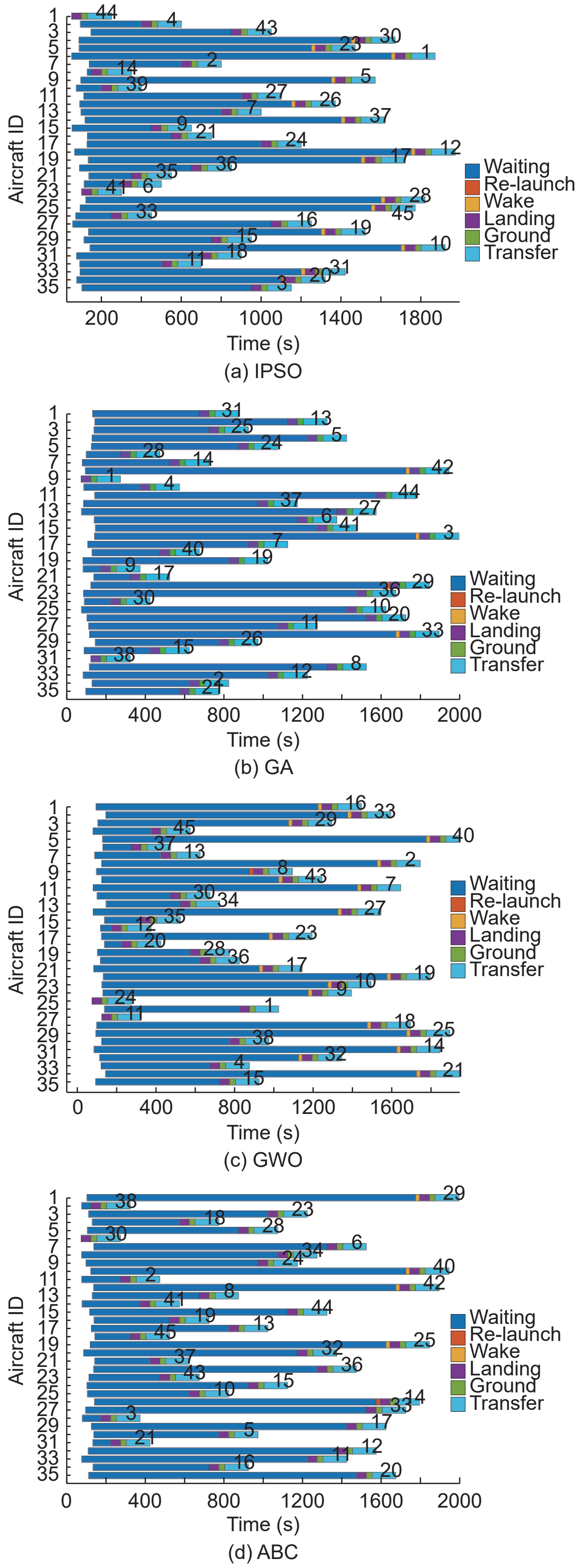

Fig. 7 Gantt chart of landing support operations for 35 targets.

results to the traditional genetic algorithm, and there is almost no significant difference between the scheduling time and the algorithm running time. For simpler scheduling problems, both the genetic algorithm and the particle swarm algorithm can effectively search for an approximate optimal solution, and the computational complexity of both is relatively low.

Table 3 Comparison table of experimental results

Group	Algorithm model	Total scheduling time (s)	Number of objects
Exp. 1	IPSO	987	15
	GA	993	
	GWO	995	
	ABC	993	
Exp. 2	IPSO	1478	25
	GA	1485	
	GWO	1492	
	ABC	1488	
Exp. 3	IPSO	1953	35
	GA	1981	
	GWO	1995	
	ABC	1994	

Across the three representative scales—small (15), medium (25), and large (35)—the proposed algorithm consistently outperforms GA, artificial bee colony (ABC), and GWO in terms of scheduling efficiency, stability, and resource utilization. These results provide stronger empirical evidence supporting the scalability and robustness of the proposed model.

However, in the large-scale multi-objective task scheduling problem, the improved particle swarm algorithm shows obvious advantages, and especially in the algorithm running time has been significantly reduced. Thanks to the global search capability and update strategy of the particle swarm algorithm, the improved particle swarm algorithm is able to complete more iterations in a shorter period of time, and converge to a more optimal solution in fewer iterations, thus improving the overall computational efficiency.

5 Conclusions

In this paper, a multi-objective cluster guarantee scheduling model is established for the easily neglected transit constraint problem in intelligent unmanned multi-objective guarantee operation scheduling. The model takes minimizing the maximum value of the completion time of all target guarantees as the optimization objective, introduces the constraint of target transit time, and enhances the anti-perturbation capability of the scheduling process through the idle mechanism of the operating machines. Example simulations verify the efficiency of the proposed algorithm in the recovery and safeguard scheduling problems, and it can adapt to the solution requirements of different scale scenarios.

However, there are some limitations, and there is still room for further improvement, which can be optimized in the following two directions: (1) resource allocation optimization: combining with the task requirements of shipboard objects, study the dynamic resource allocation strategy, and further improve the operational efficiency through resource allocation optimization; (2) dynamic scheduling adaptability: explore the inertia weight dynamic adjustment mechanism for the differences between different task types and models to cope with diversified scheduling challenges.

Acknowledgements

This research was supported by the Key Program for Basic Research of China (Grant number JCKY2023206B026).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article. The author Tao Zhang is the Editor-in-Chief of this journal.

References

[1] Wang, L.; Li, F.; Huang, J.; Zheng, X. Optimization design of ammunition scheduling scheme for carrier-based aircraft based on improved DPSO algorithm. In: Proceedings of the 5th International Conference on Algorithms, Computing and Artificial Intelligence, Article No. 34, 2022.

[2] Zhang, S.; Liu, S.; Li, Y.; Jin, Z.; Jin, Y.; Wang, S.; Zhao, J.; Xu, M. Optimization algorithm for ammunition support operation scheduling of carrier-borne aircraft. *Acta Aeronautica et Astronautica Sinica* Vol. 44, No. 20, Article No. 228485, 2023.

[3] Lyu, X. F.; Yang, D. Z.; Wang, L. T.; Ma, L. Carrier based aircraft support operation scheduling based on ammunition loading work. *Journal of Ordnance Equipment Engineering* Vol. 45, No. 1, 114–122, 165, 2024. (in Chinese)

[4] Liu, Y.; Han, W.; Su, X.; Cui, R. Optimization of fixed aviation support resource station configuration for aircraft carrier based on aircraft dispatch mission scheduling. *Chinese Journal of Aeronautics* Vol. 36, No. 2, 127–138, 2023.

[5] Zhu, X.; Zhao, Y.; Fan, J. Optimization method for one-stop support operation scheduling of carrier-based aircraft based on improved gravitational search algorithm. *Ship Electronic Engineering* Vol. 43, No. 8, 98–103, 2023.

[6] Zhu, X. D.; Zhao, Y.; Fan, J. L.; Huang, K. Joint optimization method of Pit-Stop support resource performance metrics based on layered optimization strategy. *Chinese Journal of Ship Research* Vol. 18, No. 5, 92–103, 2023.

[7] Liu, Z.; Ma, J. F.; Chen, J. F.; Ma, S. H. Carrier-based aircraft ammunition support scheduling optimization based on improved grey wolf optimizer algorithm. *Systems Engineering and Electronics* Vol. 46, No. 4, 1264–1272, 2024.

[8] Jin, Z.; Jin, L.; Zhang, B. W.; Wu, Q. S.; Feng, S.; Li, G. F.; Xu, M. L. Formal modeling and verification of carrier-borne aircraft ammunition support operation scheduling. *Journal of Software* Vol. 35, No. 9, 4100–4122, 2024. (in Chinese)

[9] Guo, F.; Han, W.; Su, X. C.; Liu, Y. J.; Cui, R. W. A bi-population immune algorithm for weapon transportation support scheduling problem with pickup and delivery on aircraft carrier deck. *Defence Technology* Vol. 22, 119–134, 2023.

[10] Liu, A. On the carrier-based aircraft deck operation scheduling problem solved by variable neighborhood search-based estimation of distribution algorithm. *Journal of Systems Science and Mathematical Sciences* Vol. 44, No. 2, 391–407, 2024.

[11] Liu, G.; Wang, H.; Lin, Y. Adaptive batch matching decision method for shipborne aircraft support operations. *Acta Aeronautica et Astronautica Sinica* 1–15, 2024.

[12] Liu, Z.; Chen, J. F.; Ma, J. F.; Ma, S. H. Simulation system for carrier-based aircraft ammunition support scheduling. *Journal of System Simulation* Vol. 36, No. 7, 1621–1630, 2024.

[13] Liu, Y. J.; Han, W.; Su, X. C.; Guo, F. Carrier aircraft landing scheduling problem based on improved gray wolf optimization. *Journal of Beijing University of Aeronautics and Astronautics* Vol. 50, No. 3, 803–813, 2024.

Haixu Li, a Ph.D. candidate at Tsinghua University and a researcher at the Systems Engineering Research Institute of China State Shipbuilding Corporation (CSSC). His research is dedicated to shipborne aviation command and support technology and shipborne aircraft takeoff and landing technology.

Tao Zhang, a tenured professor and Ph.D. supervisor at Tsinghua University. He currently serves as the chair of the Department of Automation and the dean of the Institute for Embodied Intelligence and Robotics. His long-term research interests include intelligent robotics, artificial intelligence, and control theory.

Guang Liu, a researcher at the Systems Engineering Research Institute of China State Shipbuilding Corporation (CSSC). He has long been engaged in the study of shipborne aviation command and support technology and shipborne aircraft command and dispatch algorithms.

Jian Yin, a Ph.D. candidate at Dalian Maritime University. His primary research directions include multi-equipment collaborative planning and scheduling and shipborne aircraft operation scheduling.

Zhan Shi, an M.S. student at Dalian Maritime University. His research focuses on the scheduling and planning of heterogeneous equipment operations.

Liran Shen, a professor and Ph.D. supervisor at Dalian Maritime University. Her research interests encompass pattern recognition, computer vision, image processing, machine learning, and digital signal processing.

Yunsheng Fan, a professor and Ph.D. supervisor at Dalian Maritime University. His research focuses on the intelligent navigation of unmanned ships, intelligent perception and autonomous control, swarm intelligence and testing/evaluation of unmanned systems, and situational awareness and collaborative games for marine unmanned systems.