

# 不确定环境下无人机任务分配的 种群交互式粒子群算法

饶凌风<sup>1</sup>, 耿娜<sup>2\*</sup>, 张勇<sup>1</sup>, 郭广颂<sup>3</sup>, 肖磊<sup>4</sup>

(1. 中国矿业大学信息与控制工程学院, 江苏徐州 221008; 2. 江苏师范大学电气工程及自动化学院, 江苏徐州 221116;  
3. 郑州航空工业管理学院自动化学院, 河南郑州 450046; 4. 中国轻工业联合会中国自行车协会, 北京 100079)

**摘要:** 针对灾后救援场景中多无人机(Unmanned Aerial Vehicle, UAV)任务分配所面临的时空不确定性、复杂约束耦合及动态环境适应等挑战, 本文提出了一种基于双种群协作的交互式改进粒子群算法. 构建了双目标优化模型, 将救援价值量化为时变区间函数, 并结合航程约束引入惩罚机制, 有效刻画了灾后环境的动态特性与多维度约束条件; 在算法设计上, 突破传统集中式框架的局限, 构建了主-辅种群协同进化架构, 采用问题导向的初始化策略生成高质量解集, 确保算法初始解的优越性; 进一步引入学习库驱动的双模态更新策略, 结合种群交互的实例更新机制和局部精细化搜索, 有效协调了多样性和算法收敛性. 实验结果表明, 在48架无人机的大规模场景下, 所提算法相较于多种较新算法, 在双重超体积评价指标上提升了11.8%~26.1%, 且在多变体场景中表现出更强的鲁棒性和适应性. 多种场景的测试结果验证了算法在求解效率、解集质量及环境适应性方面的综合优势, 展现了显著的理论和应用价值.

**关键词:** 不确定; 无人机任务分配; 粒子群算法; 多目标; 多种群

**基金项目:** 徐州市科技计划项目(No. KC23008); 国家自然科学基金(No. 62103255); 河南省科技攻关项目(No. 242102211095); 江苏省市场监督管理局科技计划项目(No. KJ2025018)

**中图分类号:** TP312 **文献标识码:** A **文章编号:** 0372-2112(2025)08-2678-13

**电子学报 URL:** <http://www.ejournal.org.cn>

**DOI:** 10.12263/DZXB.20250410

## Population Interactive Particle Swarm Optimization Algorithm for UAV Task Allocation in Uncertain Environments

RAO Ling-feng<sup>1</sup>, GENG Na<sup>2\*</sup>, ZHANG Yong<sup>1</sup>, GUO Guang-song<sup>3</sup>, XIAO Lei<sup>4</sup>

(1. School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, Jiangsu 221008, China;

2. School of Electrical Engineering and Automation, Jiangsu Normal University, Xuzhou, Jiangsu 221116, China;

3. School of Automation, Zhengzhou University of Aeronautics, Zhengzhou, Henan 450046, China;

4. China Bicycle Association, China National Light Industry Council, Beijing 100079, China)

**Abstract:** This paper proposes an improved particle swarm algorithm based on population interaction to tackle spatio-temporal uncertainty, complex constraint coupling, and dynamic adaptation in multi-UAV (Unmanned Aerial Vehicle) task allocation for post-disaster rescue. A bi-objective optimization model quantifies rescue value as a time-varying interval function, integrating a penalty mechanism with range constraints to capture the environment's dynamic and multidimensional features. The proposed method departs from traditional centralized frameworks by introducing a main-auxiliary population co-evolution architecture. A problem-oriented initialization strategy ensures high-quality initial solutions. Additionally, a dual modal update strategy driven by a learning library, combined with population interaction and local refinement search, balances convergence speed and population diversity. Experiments with 48 UAVs demonstrate a 11.8%~26.1% improvement in the dual hypervolume metric compared to recent algorithms. Results confirm superior robustness, efficiency, and adaptability, underscoring its theoretical and practical significance.

**Key words:** uncertain; UAV task allocation; particle swarm optimization algorithm; multi-objective; multi-populations

Foundation Item(s): Xuzhou Science and Technology Project of China (No.KC23008); National Natural Science Foundation of China (No.62103255); Science and Technology Research Project of Henan Province (No.242102211095); Science and Technology Project of Jiangsu Provincial Market Supervision Administration (No.KJ2025018)

## 1 引言

无人机因其高效、灵活及低成本等优势,被广泛应用于灾难救援、军事侦察、物流运输等多个领域.传统灾后救援受时间紧迫、信息不足和安全风险等限制,效率较低,而无人机的引入为提升救援效率提供了新思路.灾难发生后,如何在最短时间内将物资送达是救援的核心任务.因此,依据环境和无人机信息,合理安排任务分配和规划飞行顺序显得尤为关键<sup>[1]</sup>.

面对任务目标多样、偏好复杂及约束条件严格救援环境,传统任务分配方法难以应对任务权重不同及时间敏感等需求,效率较低<sup>[2]</sup>.为此,研究人员针对多种应用场景设计了多样化的模型和优化方法.常见建模方法包括旅行商问题模型、协同多任务分配模型、混合整数线性规划模型和马尔科夫决策过程模型<sup>[3,4]</sup>.如,Wen等人<sup>[5]</sup>针对核事故救援,构建了结合多目标优化与环境信息的模型;Wang等人<sup>[6]</sup>则综合距离、角度等因素建模.求解方法主要包括集中式和分布式.分布式方法通过局部信息交互实现协同决策<sup>[7]</sup>,如基于共识的捆绑算法、模型预测控制和拍卖法<sup>[8]</sup>.但难以获得全局最优且在大规模不确定环境下存在局限.集中式方法便于整合全局信息与全局优化,适用于多目标强耦合和高精度需求,主要包括最优化和启发式方法.最优化方法如分支定界法、整数规划、博弈论和动态规划等<sup>[9]</sup>,可获得全局最优解,但在任务复杂时计算难度和时间大幅增加,且对强约束适应性较差<sup>[10,11]</sup>.启发式算法如粒子群(Particle Swarm Optimization, PSO)、蚁群和模拟退火<sup>[12]</sup>在复杂模型求解中表现良好,随着模型复杂度增加,学者对这些算法不断改进<sup>[13-15]</sup>.然而,上述方法在复杂问题下泛化性和性能仍有限.粒子群算法因参数少、易于拓展等特点,在任务分配问题中表现突出<sup>[16,17]</sup>.

灾后救援场景还具有显著的信息和环境不确定性及任务耦合性.常用的不确定信息表示方法包括置信水平、随机数、模糊数和区间数<sup>[18]</sup>.如,Xiong等人<sup>[19]</sup>从目标分类不确定性出发,提出了相应的模型与方法,但模型与算法衔接仍存在不足,影响算法的实际应用效果.

针对上述问题,本文量化了任务救援价值,建立以降低救援惩罚和缩短任务完成时间为目标的模型,提出种群交互学习的粒子群算法(Particle Swarm Optimization for Population Interaction Learning, PILPSO).通过构建不确定救援价值体系,提出基于区间数动态变化

的任务价值量化模型,实现任务价值的动态表征.设计了双种群PSO优化算法,提出“全局引导-局部精修”学习机制,设计区间数优化的双重超体积指标,设计混合非支配排序机制.

## 2 问题描述与模型建立

### 2.1 问题描述

考虑某一灾后场景,共有 $I$ 架无人机,每架无人机分配一组救援任务,分别记为 $O_1, O_2, \dots, O_r$ .区域内有 $m$ 个位置已知且可互通的物资补给站;灾区内分布多个救援点(即任务或目标点),每个救援点中有不同物资需求的被困人员. $N$ 个任务位置已知.此外,无人机的航程、载货量和速度等都有限制,需从基站Base出发,途中可补给物资.在任务信息不确定情况下,需依据任务的救援价值,合理规划无人机的任务分配策略,以最小化救援惩罚和任务完成时间.图1为多无人机协同救援示意图.基于问题描述,给出参数信息,如表1所示.

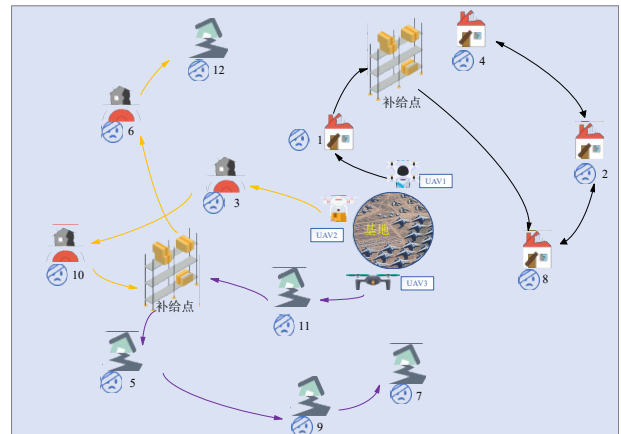


图1 救援任务分配示意图

### 2.2 救援价值的计算

为了实现在时间最短和惩罚最小的情况下完成救援任务,需根据任务的信息量化其救援价值.鉴于复杂且不确定的灾后,参阅文献[20]和文献[21],利用获取的环境信息,估算救援点内区间被困人员数量.

本文将救援优先级分为轻、中和重,分别用 $L, M$ 和 $H$ 表示.根据卫星等获取的先验信息采用区间数确定救援点各等级的伤员数量.若根据文献[20]确定某一等级人数为 $P_r$ ,则该等级人数为 $[P_r(1 - \text{err}), P_r(1 + \text{err})]$ ,其中 $\text{err}$ 为误差(模型结果与实际结果之间).依据文献[20],三个登记的人数误差分别为27%、20%以及12%.据

表1 参数定义及注释

符号	属性
Task	救援任务集合, Task={task <sup>1</sup> , task <sup>2</sup> , ..., task <sup>N</sup> }
$\bar{V}(t)$	救援任务在 $t$ 时刻的价值向量, $\bar{V}(t) = \{\bar{v}_1(t), \bar{v}_2(t), \dots, \bar{v}_n(t)\}$
$V_s$	救援任务所需物资向量, $vs_n$ 为救援点 $n$ 所需的救援物资量
$T_j^i$	无人机 $i$ 到达任务点 $j$ 时所需时间
$t$	到达救援点 $j$ 卸载物资、到达补给站补充物资的时间
$D_{(N+M+1)}$	任务点间距离矩阵, $D_{(N+M+1)} = \{d_{ij}\}_{I \times N}$
Pt	$N$ 个救援点集合, Pt={Pt <sub>1</sub> , Pt <sub>2</sub> , ..., Pt <sub>N</sub> }
Sp	$M$ 个补给站集合, Sp={Sp <sub>1</sub> , Sp <sub>2</sub> , ..., Sp <sub>M</sub> }
Battery	无人机的电池容量(最大航程限制)
$S_{\max}$	无人机的最大容量限制
$vel_{\max}$	无人机的最大飞行速度
$\rho$	无人机的单位飞行耗电量

此,可以估算出每个优先级的伤员人数区间。

确定各等级伤员人数时,以整个救援区域内所有待救人员的区间总数为基准,计算每个救援点内各等级伤员数区间占总数的比值,并转换为相对值,具体如式(1)所示:

$$\begin{aligned} H(\text{ph}_i) &= [\text{ph}_i^- / H_{\text{sum}}^+, \text{ph}_i^+ / H_{\text{sum}}^-] \\ M(\text{pm}_i) &= [\text{pm}_i^- / M_{\text{sum}}^+, \text{pm}_i^+ / M_{\text{sum}}^-] \\ L(\text{pl}_i) &= [\text{pl}_i^- / L_{\text{sum}}^+, \text{pl}_i^+ / L_{\text{sum}}^-] \end{aligned} \quad (1)$$

其中,  $H(\text{ph}_i)$ 、 $M(\text{pm}_i)$  和  $L(\text{pl}_i)$  分别表示救援点  $i$  内重、中以及轻伤人数的相对比值,且  $\text{ph}_i = [\text{ph}_i^-, \text{ph}_i^+]$ 、 $\text{pm}_i = [\text{pm}_i^-, \text{pm}_i^+]$  和  $\text{pl}_i = [\text{pl}_i^-, \text{pl}_i^+]$  分别表示救援点  $i$  内重、中以及轻三个等级的区间伤员数;三个等级的总人数分别表示为  $[H_{\text{sum}}^-, H_{\text{sum}}^+]$ 、 $[M_{\text{sum}}^-, M_{\text{sum}}^+]$  和  $[L_{\text{sum}}^-, L_{\text{sum}}^+]$ 。

救援过程中,伤员伤情会随时间恶化,相应的救援价值也会逐步提升。其计算公式如式(2)所示:

$$\begin{aligned} \bar{V}(t) &= \begin{bmatrix} \bar{v}_h(t) \\ \bar{v}_m(t) \\ \bar{v}_l(t) \end{bmatrix} = \begin{bmatrix} v_h^-(t) & v_h^+(t) \\ v_m^-(t) & v_m^+(t) \\ v_l^-(t) & v_l^+(t) \end{bmatrix} \\ \begin{cases} v_h(t) = \omega_1 \times H(\text{ph}_i) \times (1 + e^{\gamma_1 \times t}) \\ v_m(t) = \omega_2 \times M(\text{pm}_i) \times (1 + e^{\gamma_2 \times t}) \\ v_l(t) = \omega_3 \times L(\text{pl}_i) \end{cases} \end{aligned} \quad (2)$$

其中,  $\gamma_1$  和  $\gamma_2$  分别为重和中伤的价值增长系数;  $t$  为时间;  $\omega_1$ 、 $\omega_2$  和  $\omega_3$  分别为救援价值权重,由决策者确定。

## 2.3 不确定优化模型

### 2.3.1 目标函数

(1) 救援综合惩罚  $f_1$

本文所研究问题的救援综合惩罚函数如式(3)所示:

$$f_1 = \sum_{i=1}^I \sum_{j=1, i \neq j}^J \int_0^{T_j^i} \bar{v}_{O_{i,j}}(t) dt \quad (3)$$

式中,  $\bar{v}_{O_{i,j}}(t)$  为  $t$  时刻救援点  $O_{i,j}$  的救援价值;  $\int_0^{T_j^i} \bar{v}_{O_{i,j}}(t) dt$  表示无人机  $i$  对救援点  $O_{i,j}$  的累计惩罚;  $T_j^i$  为第  $i$  个无人机到  $O_{i,j}$  的累计时间:

$$T_j^i = \sum_{j=1}^{J_i} \frac{d_{O_{i,j-1}, O_{i,j}}}{\text{vel}_{i,j}} + \Delta T \quad (4)$$

式中,  $\Delta T$  为卸货时间和补货时间之和;  $\text{vel}_{i,j}$  为无人机  $i$  的飞行速度,采用式(5)计算:

$$\text{vel}_{i,j} = (1 - \varepsilon) \times \text{vel}_{\max} + \varepsilon \times \text{vel}_{\max} \times \frac{(S_{\max} - S_{i,j})}{S_{\max}} \quad (5)$$

其中,  $S_{i,j}$  为无人机  $i$  到达救援点  $j$  前的载货量;  $\varepsilon$  为系数。

(2) 最短救援时间  $f_2$

无人机完成任务所需时间  $f_2$ , 根据式(6)计算:

$$f_2 = \sum_{i=1}^I T_{\text{end}-1}^i + I \times \Delta T \quad (6)$$

其中,  $T_{\text{end}-1}^i$  为第  $i$  个无人机抵达其分配列表中最后一个救援点的时间;  $T_{\text{end}}^i$  为无人机  $i$  执行完任务返回基地的时刻。

### 2.3.2 约束条件

(1) 无人机自基地出发并完成任务返航的总飞行距离不得超过该型号无人机的续航(距离)上限:

$$\sum_{j=1}^{J_i-1} d_{O_{i,j-1}, O_{i,j}} \leq \frac{\text{Battery}}{\rho} \quad (7)$$

其中,  $J_i$  为无人机  $i$  执行任务数量;  $\rho$  为单位路程耗电量。

(2) 无人机卸载的物资总量不大于无人机的容量:

$$\max(vs_{O_{i,j}}) \leq S_{\max} \quad (8)$$

其中,  $J_i'$  为无人机  $i$  执行任务序列  $O_i$  中任意两个基地/补给站的连续执行的救援任务。

(3) 每项任务必须由一架无人机承担,且所有任务都需被分配并由无人机完成:

$$\begin{cases} \text{size} \left( \sum_{i=1}^I x_{i,j} \right) = N, \quad j = 1, 2, \dots, J \\ \delta \left( \sum_{j=1}^J x_{i,j} \right) = 1, \quad i = 1, 2, \dots, N \end{cases} \quad (9)$$

其中,  $\text{size} \left( \sum_{i=1}^I x_{i,j} \right)$  表示矩阵  $X$  中第  $j$  列包含的任务点数量;  $\delta \left( \sum_{j=1}^J x_{i,j} \right)$  表示矩阵  $X$  中第  $i$  列所含任务  $j$  的数量。

(4) 当无人机剩余的物资不满足下一个任务的要求时,需要进行物资补给。完成  $m$  个任务后补给,编码如下:

$$O_{i,j+1} = 100 + m, \quad \text{if } S_{O_{i,j}} < vs_{O_{i,j+1}} \quad (10)$$

其中,  $S_{O_{i,j}}$  为无人机  $i$  完成救援任务  $O_{i,j}$  后剩余的物资量。

(5)每个无人机均从基地出发:

$$O_{i,1} = 0 \quad (11)$$

(6)每个无人机执行完任务后均须返回基地:

$$O_{i,\text{end}} = 0 \quad (12)$$

(7)每个无人机至少执行一个任务:

$$\exists O_{i,j} < 100 \cap O_{i,j} > 0, j = 1, 2, \dots, J_i \quad (13)$$

### 2.4 不确定模型转化

为便于计算,本节将表示不确定信息的区间目标函数 $f_i$ 转换为确定性信息的精确数,提出一种基于区间可信度加权的转换方法,将区间目标值转换为确定目标值.

具体地,通过引入决策者容忍系数 $\zeta \in [0, 1]$ ,将目标函数 $f_i$ 转为确定目标,具体如式(14)所示:

$$f_{\text{lexa}} = \sqrt{f_{1m}^2 + f_{1u}^2} \quad (14)$$

其中, $f_{1u} = f_1^+ - f_2^-$ 表示目标 $f_1$ 的不确定度; $f_{1m} = f_1^- + f_{1u} \times \zeta$ 为偏好加权值.当决策者较为乐观时, $\zeta$ 取较小值;反之,则增大 $\zeta$ 取值.

为减少转换过程中的信息失真,采用区间数进行比较与非支配排序.为此,本文给出一种新的区间大小比较方法:设有区间数 $a = [a^-, a^+]$ 和 $b = [b^-, b^+]$ ,当 $\zeta \times (b^+ - a^+) + (1 - \zeta) \times (b^- - a^-) > 0$ 时,则 $b > a$ ;反之,则 $a > b$ ;若 $\zeta \times (b^+ - a^+) + (1 - \zeta) \times (b^- - a^-) = 0$ ,则 $b = a$ .

基于上述转换,本文所求的数学模型表示如下:

$$\begin{cases} F = \min(f_1, f_2) \\ \sum_{j=1}^{J_i-1} d_{O_{i,j-1}, O_{i,j}} \leq \frac{\text{Battery}}{\rho} \\ \max(vs_{O_{i,j}}) \leq S_{\max} \\ \text{size}(\sum_{i=1}^I x_{i,j}) = N, \quad j = 1, 2, \dots, J \\ \text{s.t.} \begin{cases} \delta(\sum_{j=1}^J x_{i,j} = j) = 1, \quad i = 1, 2, \dots, N \\ O_{i,j+1} = 100 + m, \text{ if } S_{O_{i,j}} < vs_{O_{i,j+1}} \\ O_{i,1} = 0 \\ O_{i,\text{end}} = 0 \\ \exists O_{i,j} < 100, \quad j = 1, 2, \dots, J_i \end{cases} \end{cases} \quad (15)$$

### 3 算法设计

本文提出了一种双种群协同进化框架(见图2),构建主-辅种群双驱动更新机制:主种群采用分解式学习策略,动态权衡救援价值与时间成本;辅种群执行“变异+学习”机制,并结合基于非支配等级划分的多算子更新策略,提升全局探索能力;同时,引入区间数支配排序定义,保留原始不确定信息用于解的评估.设计了

两个核心策略:

(1)问题导向的种群初始化策略.结合初始化聚类分配、任务排序以及概率补给策略,生成高质量的初始解集.

(2)主辅种群交互式学习更新策略.主种群通过目标空间投影,实现个性化和分解式学习,优化更新方向与冲突任务协调;辅助种群采用基于排序的多模态更新算子和帕累托前沿延拓操作,拓展解集多样.

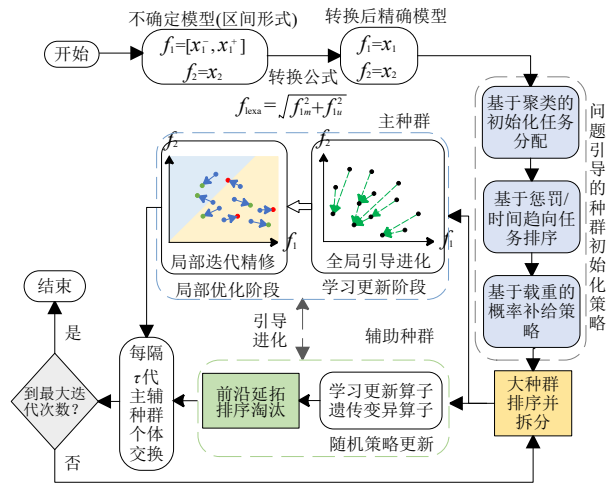


图2 算法框架

迭代过程中综合利用原始不确定性模型和加权法后的确定性模型.采用双重超体积(Double Hypervolume, DHV)评价指标(第4.1节)对每代解集进行排序(第2.4节),减少信息失真;确定性模型参与算子运算以提高计算效率.伪代码如算法1所示.

#### 3.1 编码与解码

本文中,采用整数编码方式,每个粒子代表一个问题的解,解码为无人机的任务序列,粒子结构由两段组成:

第一段:分配的任务,结构为 $1 \times I$ 的嵌套数组(矩阵),数组内整数为每个无人机分配的任务.

第二段:补给的情况,大小为 $1 \times ns_i$ 的数组,其中, $ns_i$ 为无人机 $i$ 补给的次数.数组元素为任务序号,表示执行完该任务后补给.

解码时,采用两段嵌入方法:根据第二段数组,在第一段任务序列中嵌入起点、补给点和终点.无人机起点和终点为基地,用0表示;补给点依据式(16)选择距离下一个执行任务最近的那个.嵌入上述信息后,得到无人机路径 $O_i$ .

以表2为例,无人机1的救援序列为[2 5 9 6],第二段为[2],因此其路径为 $O_1 = [0 2 5 102 9 6 0]$ ;同理,有 $O_2 = [0 8 4 101 5 7 102 12 0]$ 和 $O_3 = [0 11 10 0]$ .

**算法 1 基于种群交互学习的粒子群算法**

输入: 种群规模  $P$ 、最大迭代次数  $G$ 、救援任务集合  $\text{Task}$ 、救援无人机数量  $I$ 、补给站集合  $S_p$

输出: 帕累托最优解集  $P\_best$ ;

```

1.Begin
2. For  $p = 1$  to  $P$  do %种群初始化
3. Clusters $\leftarrow$ initial(Task,  $I$ ); %任务分配初始化操作
4.  $X_p \leftarrow$ taskseq(Clusters); %任务排序操作
5.  $X_{p\_supply} \leftarrow$ prorep( $X_p, S_p$ ); %概率补给操作
6.  $Op \leftarrow$ generate( $X_{p\_supply}, X_p$ ); %生成粒子操作
7.  $Pop \leftarrow Pop \cup \{Op\}$ ;
8. End For
9. ( $Pop1, Pop2$ ) $\leftarrow$ division( $Pop, \theta$ ) %划分主辅种群操作
10. For  $g = 1$  to  $G$  do %迭代进化
11. For each  $Op$  in  $Pop1$  do %主种群更新
12.  $Op\_learn \leftarrow$ personsel( $Op, Pop2$ ); %选择学习对象
13.  $Op \leftarrow$ decompose( $Op, Op\_learn$ ); %分解式学习更新
14. End For
15.  $Pop1 \leftarrow$ localopt( $Pop1$ ); %局部优化操作
16. For each  $Op$  in  $Pop2$  do %辅助种群更新
17.  $Op \leftarrow$ update( $Op, \Delta_1, \Delta_2$ ); %多策略更新操作
18. End For
19. if  $\text{mod}(g, \tau) = 0$  %取余操作
20. ( $Pop1, Pop2$ ) $\leftarrow$ exchange( $Pop1, Pop2$ ); %交换个体
21. End If
22. End For
23.  $P\_best \leftarrow$ optimalset( $Pop1 \cup Pop2$ ); %获取最优解集操作
24. Return  $P\_best$ 
25. End

```

表 2 粒子两段编码实例

无人机编号	1	2	3
所被分配的任务(第一段)	[2 5 9 6]	[8 4 1 5 7 12]	[11 10]
补给时刻(第二段)	[2]	[2 5]	[1]

### 3.2 问题特征引导的初始化策略

针对灾后任务空间分布的强区域关联性、任务动态优先级以及补给决策随机性等挑战,本文给出三级初始化策略,具体如下。

#### 3.2.1 基于聚类的初始化任务分配

如图 3 所示,采用  $K$ -means 聚类方法来生成初始解,用于确定粒子首行的取值。首先,用于确定粒子首行的取值。首先,将所有任务依据距离特征划分为  $I$  个类别,然后将这  $I$  个类别分别随机指派给  $I$  架无人机。同时,构建外部档案集  $Ar1$ ,用于保存多种不同的聚类分配结果,并通过限定聚类次数有效控制档案集规模。分配过程中,依次选取  $Ar1$  中存储的不同聚类方

案,分配给尚未初始化的粒子,以提升粒子的多样性,减少同质化现象。

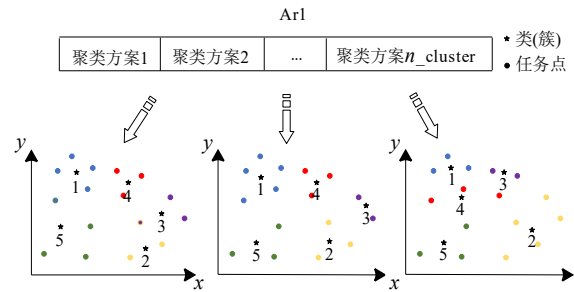


图 3 初始化任务分配示意图

#### 3.2.2 基于惩罚或时间趋向的任务排序

在获取初始任务分配方案后,须明确任务执行顺序。针对这一需求,本文引入了一种同时考虑两个优化目标的综合评价方法。假设某一救援任务为  $\text{task}^n$ ,则执行该任务的综合指标  $\text{Gain}_i^t$  为

$$\text{Gain}_i^t(\text{task}^n) = \frac{(1-\alpha)T_j^i + \alpha \times v_n(t)}{d_{\text{pos}_i, n}} \quad (16)$$

其中,  $d_{\text{pos}_i, n}$  表示  $\text{task}^n$  与无人机  $i$  之间的距离;  $\text{pos}_i$  为无人机  $i$  的位置;  $\alpha$  为权重参数,取值为 0 或 1。

#### 3.2.3 基于载重的概率补给策略

任务执行过程中,合理选择物资补给时刻至关重要。假设无人机  $i$  在完成第  $j$  个任务后,其已卸载的物资总量为  $\text{unload}_i^j$ ,那么,无人机  $i$  的补给概率为

$$\text{Sr}_i^j = \frac{\text{unload}_i^j}{S_{\max}} \quad (17)$$

当  $\text{Sr}_i^j > 1$  且  $\text{Sr}_i^{j-1} < 1$  时,无人机须强制补给。

### 3.3 主种群进化

本文中,主种群通过设计个性化学习方向和分解式学习方法两个机制/算子,解决如下两个关键问题:

第一,为每个个体选择个性化的学习对象,避免种群集中于单一方向,确保种群的多样性。

第二,引入分解式学习方法,通过局部修复策略将不可行解转化为可行解,避免计算资源浪费,提高算法效率。

初始化后,采用第 2.4 节区间非支配排序,进化主种群  $Pop$ ,生成新的粒子排序种群  $Pop'$ ,如图 4 所示。

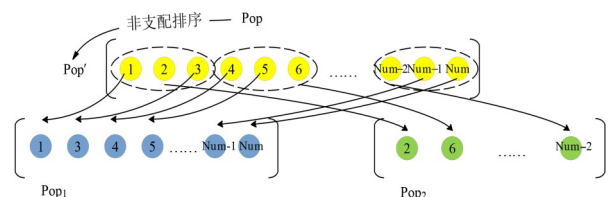


图 4 种群划分示意

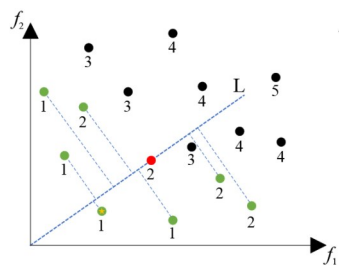
### 3.3.1 个性化学习方向

为实现个体学习对象的个性化选取,采用以下步骤:

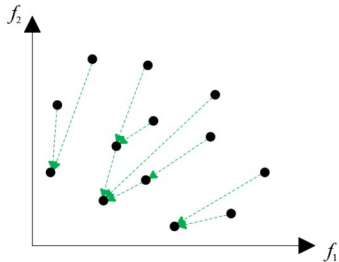
(1)确定学习对象的范围,从学习库中筛选出未被更新粒子支配的粒子.

(2)选择最终学习对象,在上述集合中,挑选出与待更新粒子到原点连线距离最近的粒子.

图5(a)中,粒子用圆点表示,数字为粒子支配等级.连接原点与红色粒子得直线L,计算绿色粒子到L的垂直距离,选取距离最近者为学习对象(黄色粒子).该方法可防止陷入局部最优.图5(b)为学习对象选择示例,部分非前沿粒子未选择前沿粒子,从而保证了种群多样性.



(a) 个体学习对象示例



(b) 学习对象选择示例

图5 学习对象选择过程

### 3.3.2 分解式学习更新步骤

在选择学习对象后,本文采用文献[21]所提方法对粒子进行更新.图6给出两阶段更新示意图.图中,UAV2与UAV1以及UAV3的分配方案存在冲突,任务6、8重复分配,而任务1、7未被分配,该解不可行.为修正上述粒子,本文采用文献[21]所提基于分解学习的粒子更新方法进行处理,将粒子修正/更新为可行解.

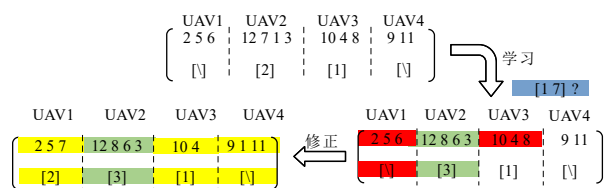


图6 更新两阶段示意图

### 3.4 基于辅助种群的学习库进化

辅助种群作为学习库,通过基于排序的多策略更新算子与主种群进行交互,从而增强种群的寻优能力,提升算法的收敛速度,有效解决算法收敛缓慢问题.

#### 3.4.1 学习库内部更新方式

学习库内部个体的更新采用两种策略 $\Delta_1$ 和 $\Delta_2$ ,两者分别依据第3.3节和文献[21]所提遗传变异策略进行更新.具体实施时,种群其中更新算子 $\Delta_2$ ,引入文献[21]所提遗传变异算子.

具体实施时,辅助种群依据式(18)被划分为两个部分,分别采用不同的更新算子分为两个区块,每个区块采用不同的算子:

$$\text{rank}_{\text{scope}} = \lceil 1, \text{rand}([1, \text{rank}_{\text{max}}/2]) \rceil \quad (18)$$

其中, $\text{rank}_{\text{scope}}$ 表示种群划分的非支配等级阈值; $\text{rank}_{\text{max}}$ 为当前最大非支配等级;若根据式(18)生成的随机数小于 $\text{rank}_{\text{scope}}$ 的优质个体采用 $\Delta_2$ 更新,有助于跳出局部最优,扩大搜索范围.大于则采用 $\Delta_1$ 更新,加速种群收敛.

采用 $\Delta_2$ 算子的个体根据其在目标空间的位置引导变异,将个体按 $f_1/f_2$ 值排序,前半部分以优化 $f_1$ 为主,后半部分以优化 $f_2$ 为主,变异过程采用第3.2.2节提出策略.

图7给出了学习库内部进化过程:蓝色和黑色个体为待更新个体,红色为不可替换的劣质个体,绿色为可替换优质个体.蓝色个体根据蓝色箭头所指 $f_1$ 为导向的变异方向采用 $\Delta_2$ 算子更新,淡黄色区域则以 $f_2$ 为导向.黑色个体采用 $\Delta_1$ 算子更新.上述操作能够保证算法收敛性以及解的帕累托前沿的延拓性.

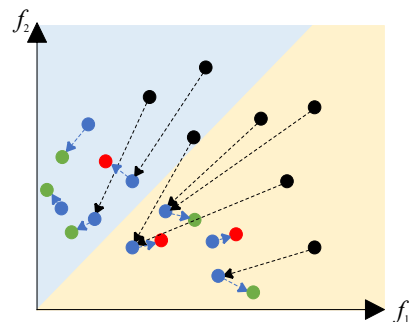


图7 学习库内部种群更新

#### 3.4.2 学习库与主种群的交互

由于学习库规模有限,随着新个体的加入,需引入淘汰机制以保持规模不变.主种群依据第3.3节方法根据个体位置和非支配等级来选择学习对象.而辅助种



## 4.2 实验及分析

### 4.2.1 参数实验

选取 5 个关键参数进行测试,包括救援价值增长速度参数  $\gamma_1$  和  $\gamma_2$ ,速度影响系数  $\varepsilon$ ,决策者容忍系数  $\zeta$ ,主辅种群规模之比  $\theta$  以及主辅种群交互频率  $\tau$ .

#### (1) 算法相关参数取值分析

首先,分别测试参数  $\tau$  对算法性能的影响,结果如表 4 所示. 以  $\tau=20$  为 Wilcoxon 秩和检验基准,结果显示,在主要场景 1 中,算法性能与  $\tau$  负相关;在复杂度较高的主要场景 2 和 3 中, $\tau=20$  时性能最优. 说明合理控制主辅种群交互频率有助于平衡算法收敛速度与解的多样性.

表 4 参数  $\tau$  不同取值对比结果

场景	$\tau$ 取值	Mean	Std	Wilcoxon
MS1	$\tau=5$	$2.74 \times 10^5$	$3.04 \times 10^3$	--
	$\tau=10$	$2.72 \times 10^5$	$3.30 \times 10^3$	$\approx$
	$\tau=20$	$2.72 \times 10^5$	$3.61 \times 10^3$	$\approx$
	$\tau=40$	$2.71 \times 10^5$	$4.63 \times 10^3$	++
	$\tau=60$	$2.70 \times 10^5$	$4.18 \times 10^3$	++
MS2	$\tau=5$	$7.60 \times 10^5$	$2.93 \times 10^4$	++
	$\tau=10$	$7.66 \times 10^5$	$2.17 \times 10^4$	$\approx$
	$\tau=20$	$7.70 \times 10^5$	$3.54 \times 10^4$	$\approx$
	$\tau=40$	$7.65 \times 10^5$	$2.47 \times 10^4$	++
	$\tau=60$	$7.64 \times 10^5$	$2.44 \times 10^4$	++
MS3	$\tau=5$	$7.60 \times 10^5$	$2.93 \times 10^4$	++
	$\tau=10$	$7.66 \times 10^5$	$2.17 \times 10^4$	$\approx$
	$\tau=20$	$7.70 \times 10^5$	$3.54 \times 10^4$	$\approx$
	$\tau=40$	$7.65 \times 10^5$	$2.47 \times 10^4$	++
	$\tau=60$	$7.64 \times 10^5$	$2.44 \times 10^4$	++

其次,测试主辅种群规模之比  $\theta$  对算法性能的影响,结果如表 5 所示. 种群规模为 80,  $\theta=5/3$  时(主种群 50,辅种群 30)作为 Wilcoxon 秩和检验基准. 结果表明,  $\theta=5/3$  时算法性能最优,且在场景复杂度增加时优势更为明显.

表 5 参数  $\theta$  不同取值对比结果

场景	$\theta$ 取值	Mean	Std	Wilcoxon
MS1	$\theta=5/3$	$2.72 \times 10^5$	$3.60 \times 10^3$	$\approx$
	$\theta=1$	$2.71 \times 10^5$	$3.41 \times 10^3$	$\approx$
	$\theta=3/5$	$2.69 \times 10^5$	$3.67 \times 10^3$	++
MS2	$\theta=5/3$	$7.70 \times 10^5$	$2.54 \times 10^4$	$\approx$
	$\theta=1$	$7.56 \times 10^5$	$2.62 \times 10^4$	++
	$\theta=3/5$	$7.43 \times 10^5$	$2.83 \times 10^4$	++
MS3	$\theta=5/3$	$1.83 \times 10^6$	$7.86 \times 10^4$	$\approx$
	$\theta=1$	$1.75 \times 10^6$	$7.53 \times 10^4$	++
	$\theta=3/5$	$1.74 \times 10^6$	$8.55 \times 10^4$	++

#### (2) 模型相关参数取值分析

为验证模型参数对求解方案的影响,分别统计 30 次实验的  $f_2$  均值 ( $avg\_f_2$ )、无人机补给次数均值 ( $avg\_sup$ ) 及方案不确定度均值 ( $avg\_unc$ ),以评估各参数对模型的影响.

如表 6 所示,当  $\gamma_1$  和  $\gamma_2$  较小时,救援点价值增长速度较慢,距离无人机较近的救援点获得更高综合收益,  $f_2$  更优;当  $\gamma_1$  和  $\gamma_2$  较大时,  $f_2$  表现相对较差.

表 6 参数  $\gamma_1$  和  $\gamma_2$  不同取值对比结果

参数 $\gamma_1$ 和 $\gamma_2$ 取值	$avg\_f_2$		
	MS1	MS2	MS3
$\gamma_1/\gamma_2=0.04/0.02$	223.83	453.49	921.43
$\gamma_1/\gamma_2=0.02/0.01$	217.22	450.37	914.51
$\gamma_1/\gamma_2=0.08/0.04$	229.62	456.53	954.68

表 7 中,当  $\varepsilon=0.33$  时,无人机载重对速度影响小,无人机倾向于单次补给更多物资,以牺牲速度换取更短路程和更高收益;当  $\varepsilon=0.67$  时,补给次数增加,单次物资携带量减少;当  $\varepsilon=0.99$  时,无人机通过多次补给以减轻载重.

表 7 参数  $\varepsilon$  不同取值对比结果

参数 $\varepsilon$ 取值	$avg\_sup$		
	MS1	MS2	MS3
$\varepsilon=0.67$	1.81	6.51	12.36
$\varepsilon=0.33$	0.95	4.40	9.00
$\varepsilon=0.99$	2.69	8.43	16.41

表 8 可以看出不同  $\zeta$  值对整体平均不确定度有明显影响. 适中的  $\zeta$  取值可以使方案的不确定度最低;过大或过小的  $\zeta$  值都会使整体方案的不确定度增加.

表 8 参数  $\zeta$  不同取值对比结果

参数 $\zeta$ 取值	$avg\_unc$		
	MS1	MS2	MS3
$\zeta=0.67$	70.05	70.12	114.00
$\zeta=0.10$	70.99	75.70	122.38
$\zeta=1.00$	69.12	73.24	118.64

#### (3) 综合性能指标分析

图 9 给出了算法在三个主要场景下  $f_1$  为首要目标的结果,可以看出无人机能高效完成任务分配. 在场景 1 中,无人机载货量小,无需补给. 主要场景 2 中,无人机的需补给以保证物资充足. 主要场景 3 救援任务更密集,无人机需权衡补给次数与单次携带物资量对目标函数  $f_1$  的影响.

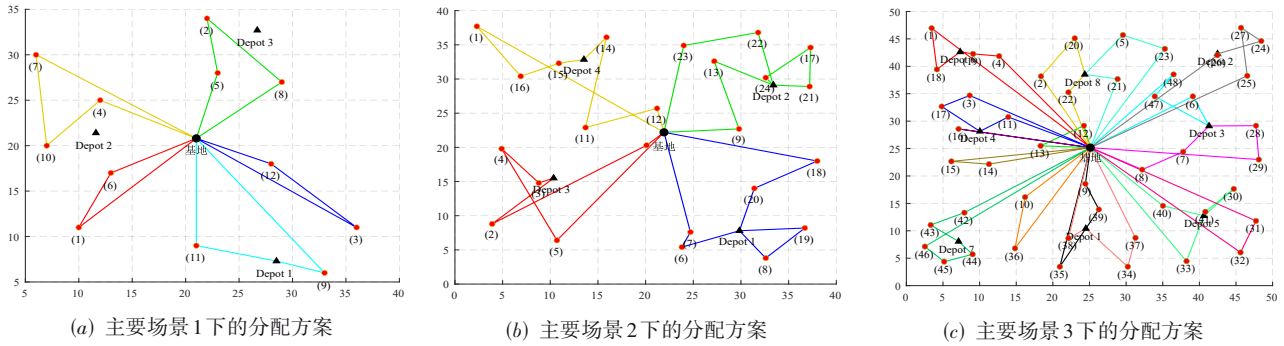


图9 PILPSO在三种主要场景下的任务分配方案

为突出所提算法处理复杂问题的优势,重点分析MS3场景.选取了三个代表性解:分别优先考虑时间、救援价值以及两者平衡的“拐点”解.表9列出了三种解在主要性能指标上的结果.其中,任务分配公平性以各无人机分配任务数的标准差衡量,反映分配的均

衡性.

从表9可见,各种解表现出不同的策略倾向:时间优先解以牺牲救援价值为代价,任务完成时间缩短约14.4%;价值优先解最小化救援惩罚,但路径和时间增加;均衡解在两个目标上表现较好,分配公平性最佳.

表9 主要场景3下解的性能指标对比

性能指标	时间优先解	价值优先解	均衡解
总任务完成时间( $f_2$ )/min	895.3	1 024.7	955.1
综合救援惩罚( $f_1$ )	$1.85 \times 10^6$	$1.65 \times 10^6$	$1.71 \times 10^6$
无人机平均飞行路径/km	1 452	1 683	1 586
任务分配公平性(任务数标准差)	2.51	1.89	1.45

#### 4.2.2 消融实验

为验证所提主要策略或算子的有效性,本部分给出消融实验.所涉及的策略或算子包括:问题引导的种群初始化策略(Abl-Init)、局部更新策略(Abl-Part)以及基于学习库的粒子更新策略(Abl-Update).参数设置如下:

算法参数:主辅种群规模比 $\theta = 5/3$ ;主辅种群交互频率 $\tau = 20$ ;初始种群规模 $\text{Num} = 80$ ;最大迭代次数 $G = 200$ .

模型参数:救援价值权重参数 $\omega_1 = 4, \omega_2 = 2, \omega_3 = 1$ ;救援价值增长速度参数 $\gamma_1 = 0.04, \gamma_2 = 0.02$ ;速度影响系数 $\varepsilon = 0.67$ ;决策者容忍系数 $\zeta = 0.67$ .

##### (1)问题引导的种群初始化策略

算法 Abl-Init 采用随机方式生成初始解,并与 PILPSO 算法在 9 个场景下进行对比,结果见表 10.结果表明:在简单场景下,两者无显著差距;而在复杂场景中, PILPSO 显著优于 Abl-Init,验证了问题导向初始化策略的有效性.

##### (2)基于学习库的粒子更新策略

算法 Abl-Update 种群未拆分,没有种群交互学习机制和辅种群更新策略.为保证公平性, Abl-Update 种群规模为 80, PILPSO 主种群规模为 50, 辅种群为 30. 表 11 结果显示 Abl-Update 在所有场景中均劣于 PILPSO. 说明基于学习库的粒子更新策略在算法迭代中具有关键作用.

##### (3)局部更新策略

表10 关键策略—消融实验结果

场景	Abl-Init				PILPSO				Wilcoxon	性能提升比例/%
	Mean	Std	Best	Worst	Mean	Std	Best	Worst		
MS1	$2.73 \times 10^5$	$3.70 \times 10^3$	$2.79 \times 10^5$	$2.65 \times 10^5$	$2.72 \times 10^5$	$3.92 \times 10^3$	$2.77 \times 10^5$	$2.64 \times 10^5$	≈	0.4
MS2	$7.38 \times 10^5$	$3.24 \times 10^4$	$7.95 \times 10^5$	$6.56 \times 10^5$	$7.71 \times 10^5$	$2.94 \times 10^4$	$8.20 \times 10^5$	$6.70 \times 10^5$	++	4.2
MS3	$1.70 \times 10^6$	$1.14 \times 10^5$	$1.88 \times 10^6$	$1.40 \times 10^6$	$1.79 \times 10^6$	$8.50 \times 10^4$	$2.02 \times 10^6$	$1.65 \times 10^6$	++	5.0
VS1a	$3.02 \times 10^5$	$1.82 \times 10^4$	$3.40 \times 10^5$	$2.75 \times 10^5$	$3.24 \times 10^5$	$7.73 \times 10^3$	$3.35 \times 10^5$	$3.13 \times 10^5$	++	6.8
VS1b	$1.92 \times 10^5$	$2.12 \times 10^3$	$1.94 \times 10^5$	$1.86 \times 10^5$	$1.92 \times 10^5$	$1.99 \times 10^3$	$1.95 \times 10^5$	$1.88 \times 10^5$	≈	0.1
VS2a	$7.67 \times 10^5$	$1.04 \times 10^5$	$9.31 \times 10^5$	$3.76 \times 10^5$	$7.88 \times 10^5$	$8.15 \times 10^4$	$9.28 \times 10^5$	$6.06 \times 10^5$	++	2.7
VS2b	$5.23 \times 10^5$	$1.41 \times 10^4$	$5.59 \times 10^5$	$4.98 \times 10^5$	$5.30 \times 10^5$	$1.37 \times 10^4$	$5.58 \times 10^5$	$4.90 \times 10^5$	≈	1.3
VS3a	$1.95 \times 10^6$	$4.20 \times 10^5$	$2.57 \times 10^6$	$9.04 \times 10^5$	$2.22 \times 10^6$	$3.86 \times 10^5$	$2.79 \times 10^6$	$1.22 \times 10^6$	++	12.2
VS3b	$9.33 \times 10^5$	$5.39 \times 10^4$	$1.04 \times 10^6$	$7.84 \times 10^5$	$9.67 \times 10^5$	$3.87 \times 10^4$	$1.07 \times 10^6$	$9.11 \times 10^5$	++	3.5

表 11 关键策略二消融实验结果

场景	Abl-Update				PILPSO				Wilcoxon	性能提升 比例/%
	Mean	Std	Best	Worst	Mean	Std	Best	Worst		
MS1	2.64×10 <sup>5</sup>	3.27×10 <sup>3</sup>	2.68×10 <sup>5</sup>	2.54×10 <sup>5</sup>	2.72×10 <sup>5</sup>	3.96×10 <sup>3</sup>	2.77×10 <sup>5</sup>	2.64×10 <sup>5</sup>	++	2.9
MS2	7.29×10 <sup>5</sup>	3.08×10 <sup>4</sup>	7.80×10 <sup>5</sup>	6.42×10 <sup>5</sup>	7.71×10 <sup>5</sup>	2.94×10 <sup>4</sup>	8.20×10 <sup>5</sup>	6.70×10 <sup>5</sup>	++	5.4
MS3	1.63×10 <sup>6</sup>	1.11×10 <sup>5</sup>	1.97×10 <sup>6</sup>	1.42×10 <sup>6</sup>	1.79×10 <sup>6</sup>	8.49×10 <sup>4</sup>	2.02×10 <sup>6</sup>	1.65×10 <sup>6</sup>	++	8.9
VS1a	3.03×10 <sup>5</sup>	1.41×10 <sup>4</sup>	3.32×10 <sup>5</sup>	2.79×10 <sup>5</sup>	3.24×10 <sup>5</sup>	7.73×10 <sup>3</sup>	3.35×10 <sup>5</sup>	3.13×10 <sup>5</sup>	++	6.4
VS1b	1.85×10 <sup>5</sup>	3.15×10 <sup>3</sup>	1.91×10 <sup>5</sup>	1.80×10 <sup>5</sup>	1.92×10 <sup>5</sup>	1.99×10 <sup>3</sup>	1.95×10 <sup>5</sup>	1.88×10 <sup>5</sup>	++	3.6
VS2a	6.76×10 <sup>5</sup>	7.48×10 <sup>4</sup>	8.20×10 <sup>5</sup>	5.61×10 <sup>5</sup>	7.88×10 <sup>5</sup>	8.15×10 <sup>4</sup>	9.28×10 <sup>5</sup>	6.06×10 <sup>5</sup>	++	15.6
VS2b	4.97×10 <sup>5</sup>	1.81×10 <sup>4</sup>	5.32×10 <sup>5</sup>	4.69×10 <sup>5</sup>	5.30×10 <sup>5</sup>	1.37×10 <sup>4</sup>	5.58×10 <sup>5</sup>	4.90×10 <sup>5</sup>	++	6.2
VS3a	1.92×10 <sup>6</sup>	3.64×10 <sup>5</sup>	2.58×10 <sup>6</sup>	1.25×10 <sup>6</sup>	2.22×10 <sup>6</sup>	3.86×10 <sup>5</sup>	2.79×10 <sup>6</sup>	1.22×10 <sup>6</sup>	++	13.5
VS3b	8.56×10 <sup>5</sup>	4.39×10 <sup>4</sup>	9.38×10 <sup>5</sup>	7.26×10 <sup>5</sup>	9.67×10 <sup>5</sup>	3.87×10 <sup>4</sup>	1.07×10 <sup>6</sup>	9.11×10 <sup>5</sup>	++	11.4

变体算法 Abl-Part 移除了局部更新策略. 两种算法在 9 个场景下的实验结果如表 12 所示, Abl-Part 在多数场景中显著劣于 PILPSO 算法. 这是因为, 局部更新策略在迭代更新的基础上对帕累托前沿进行进一步优化, 不仅影响当前种群, 还对后续迭代产生积极作用.

综合表 10~表 12 可知, PILPSO 算法的整体性能优于其他三种变体算法, 并且随着场景复杂性的提升, 其展现出更为突出的性能优势. 本节中, 三个关键策略对算法性能的影响依次为: 问题引导的种群初始化策略影响 < 局部更新策略 < 基于学习库的粒子更新策略.

4.2.3 对比实验

本实验将 PILPSO 算法与基于指标的储备集自适应更新进化算法(Adaptive Archive Update Cycle Indicator Based Improved Evolutionary Algorithm, AAUC-IBEA)<sup>[23]</sup>、通信协同 PSO 算法(Collaborative Communication PSO, CCPSO)<sup>[7]</sup>、多目标分布式人工蜂群算法(Multi-Objective Discrete Artificial Bee Colony, MODABC)<sup>[24]</sup>、按步学习 PSO 算法(Step Learning PSO, SLPSO)<sup>[17]</sup>和蚁群算法(Ant Colony Optimization, ACO)<sup>[25]</sup>这 5 种算法在

9 个场景下进行对比. 算法的参数和编码方式同消融实验. 表 13 统计了 6 种算法的 DHV 值, 并对平均值、标准差、最优值、最差值及 Wilcoxon 值和检验结果进行了比较.

由表 13 可知, PILPSO 算法在多任务和多无人机配置下表现出较强的稳健性和一致性, 尤其在复杂任务中具有更好的适应性. 与其他算法相比, PILPSO 在执行效率和结果稳定性方面表现突出, 标准差较低且最优值更佳. 在简单场景(MS1)中, PILPSO 与 MODABC、SLPSO 差距不大, 但随着场景复杂度提升, PILPSO 的优势愈加显著, 除变体场景 1b 外, 展现出较强的环境适应和寻优能力.

PILPSO 算法在不同任务数量和无人机规模下均保持良好稳定性和较高效率, 适用性强, 能灵活应对实际需求. 其优异表现主要源于: 问题引导的种群初始化策略提升了种群多样性并加快收敛; 局部更新策略利用已有信息提升搜索效率; 基于学习库的粒子更新策略使粒子根据历史最优表现自我调整, 从而实现更高的稳定性和搜索效率.

表 12 关键策略三消融实验结果

场景	Abl-Part				PILPSO				Wilcoxon	性能提升 比例/%
	Mean	Std	Best	Worst	Mean	Std	Best	Worst		
MS1	2.69×10 <sup>5</sup>	3.16×10 <sup>3</sup>	2.76×10 <sup>5</sup>	2.63×10 <sup>5</sup>	2.72×10 <sup>5</sup>	3.96×10 <sup>3</sup>	2.77×10 <sup>5</sup>	2.64×10 <sup>5</sup>	++	1.1
MS2	7.37×10 <sup>5</sup>	3.74×10 <sup>4</sup>	7.97×10 <sup>5</sup>	6.37×10 <sup>5</sup>	7.71×10 <sup>5</sup>	2.94×10 <sup>4</sup>	8.20×10 <sup>5</sup>	6.70×10 <sup>5</sup>	++	4.4
MS3	1.67×10 <sup>6</sup>	7.97×10 <sup>4</sup>	1.79×10 <sup>6</sup>	1.51×10 <sup>6</sup>	1.79×10 <sup>6</sup>	8.49×10 <sup>4</sup>	2.02×10 <sup>6</sup>	1.65×10 <sup>6</sup>	++	6.7
VS1a	3.22×10 <sup>5</sup>	9.10×10 <sup>3</sup>	3.37×10 <sup>5</sup>	3.00×10 <sup>5</sup>	3.24×10 <sup>5</sup>	7.73×10 <sup>3</sup>	3.35×10 <sup>5</sup>	3.13×10 <sup>5</sup>	≈	0.6
VS1b	1.90×10 <sup>5</sup>	1.47×10 <sup>3</sup>	1.94×10 <sup>5</sup>	1.86×10 <sup>5</sup>	1.92×10 <sup>5</sup>	1.99×10 <sup>3</sup>	1.95×10 <sup>5</sup>	1.88×10 <sup>5</sup>	++	1.0
VS2a	7.51×10 <sup>5</sup>	8.76×10 <sup>4</sup>	9.12×10 <sup>5</sup>	5.60×10 <sup>5</sup>	7.88×10 <sup>5</sup>	8.15×10 <sup>4</sup>	9.28×10 <sup>5</sup>	6.06×10 <sup>5</sup>	≈	4.7
VS2b	5.05×10 <sup>5</sup>	1.30×10 <sup>4</sup>	5.28×10 <sup>5</sup>	4.76×10 <sup>5</sup>	5.30×10 <sup>5</sup>	1.37×10 <sup>4</sup>	5.58×10 <sup>5</sup>	4.90×10 <sup>5</sup>	++	4.7
VS3a	2.03×10 <sup>6</sup>	4.49×10 <sup>5</sup>	2.78×10 <sup>6</sup>	1.07×10 <sup>6</sup>	2.22×10 <sup>6</sup>	3.86×10 <sup>5</sup>	2.79×10 <sup>6</sup>	1.22×10 <sup>6</sup>	≈	8.6
VS3b	9.02×10 <sup>5</sup>	5.11×10 <sup>4</sup>	9.98×10 <sup>5</sup>	7.84×10 <sup>5</sup>	9.67×10 <sup>5</sup>	3.87×10 <sup>4</sup>	1.07×10 <sup>6</sup>	9.11×10 <sup>5</sup>	++	6.7

表 13 对比实验结果

算法	主要场景 1:12 任务/4 无人机					主要场景 2:24 任务/6 无人机				
	Mean	Std	Best	Worst	Wilcoxon	Mean	Std	Best	Worst	Wilcoxon
AAUC-IBEA	$2.63 \times 10^5$	$5.13 \times 10^3$	$2.75 \times 10^5$	$2.48 \times 10^5$	++	$6.91 \times 10^5$	$3.85 \times 10^4$	$7.62 \times 10^5$	$6.07 \times 10^5$	++
CCPSO	$2.65 \times 10^5$	$5.37 \times 10^3$	$2.73 \times 10^5$	$2.54 \times 10^5$	++	$6.89 \times 10^5$	$3.65 \times 10^4$	$7.54 \times 10^5$	$6.17 \times 10^5$	++
MODABC	$2.74 \times 10^5$	$2.75 \times 10^3$	$2.78 \times 10^5$	$2.66 \times 10^5$	≈	$7.44 \times 10^5$	$2.68 \times 10^4$	$7.86 \times 10^5$	$6.84 \times 10^5$	++
SLPSO	$2.71 \times 10^5$	$4.81 \times 10^3$	$2.77 \times 10^5$	$2.52 \times 10^5$	≈	$7.22 \times 10^5$	$4.59 \times 10^4$	$7.86 \times 10^5$	$6.29 \times 10^5$	++
ACO	$2.35 \times 10^5$	$8.52 \times 10^3$	$2.51 \times 10^5$	$2.15 \times 10^5$	++	$6.12 \times 10^5$	$5.83 \times 10^4$	$6.90 \times 10^5$	$5.21 \times 10^5$	++
PILPSO	$2.72 \times 10^5$	$3.96 \times 10^3$	$2.77 \times 10^5$	$2.64 \times 10^5$	≈	$7.71 \times 10^5$	$2.94 \times 10^4$	$8.20 \times 10^5$	$6.70 \times 10^5$	≈
算法	主要场景 3:48 任务/12 无人机									
	Mean	Std	Best	Worst	Wilcoxon					
AAUC-IBEA	$1.42 \times 10^6$	$1.86 \times 10^5$	$1.77 \times 10^6$	$1.06 \times 10^6$	++					
CCPSO	$1.51 \times 10^6$	$8.22 \times 10^4$	$1.68 \times 10^6$	$1.32 \times 10^6$	++					
MODABC	$1.68 \times 10^6$	$1.20 \times 10^5$	$1.94 \times 10^6$	$1.47 \times 10^6$	++					
SLPSO	$1.60 \times 10^6$	$1.08 \times 10^5$	$1.82 \times 10^6$	$1.35 \times 10^6$	++					
ACO	$1.15 \times 10^6$	$2.95 \times 10^5$	$1.43 \times 10^6$	$8.25 \times 10^5$	++					
PILPSO	$1.79 \times 10^6$	$8.49 \times 10^4$	$2.02 \times 10^6$	$1.65 \times 10^6$	≈					
算法	变体场景 1a:12 任务/3 无人机					变体场景 1b:12 任务/5 无人机				
	Mean	Std	Best	Worst	Wilcoxon	Mean	Std	Best	Worst	Wilcoxon
AAUC-IBEA	$2.80 \times 10^5$	$1.77 \times 10^4$	$3.37 \times 10^5$	$2.55 \times 10^5$	++	$1.86 \times 10^5$	$3.39 \times 10^3$	$1.93 \times 10^5$	$1.79 \times 10^5$	++
CCPSO	$2.91 \times 10^5$	$1.38 \times 10^4$	$3.14 \times 10^5$	$2.66 \times 10^5$	++	$1.88 \times 10^5$	$3.53 \times 10^3$	$1.95 \times 10^5$	$1.78 \times 10^5$	++
MODABC	$3.07 \times 10^5$	$1.47 \times 10^4$	$3.44 \times 10^5$	$2.79 \times 10^5$	++	$1.92 \times 10^5$	$1.85 \times 10^3$	$1.94 \times 10^5$	$1.88 \times 10^5$	≈
SLPSO	$2.96 \times 10^5$	$2.04 \times 10^4$	$3.43 \times 10^5$	$2.68 \times 10^5$	++	$1.89 \times 10^5$	$2.22 \times 10^3$	$1.93 \times 10^5$	$1.82 \times 10^5$	++
ACO	$2.55 \times 10^5$	$2.15 \times 10^4$	$2.98 \times 10^5$	$2.23 \times 10^5$	++	$1.73 \times 10^5$	$5.31 \times 10^3$	$1.84 \times 10^5$	$1.63 \times 10^5$	++
PILPSO	$3.24 \times 10^5$	$7.73 \times 10^3$	$3.35 \times 10^5$	$3.13 \times 10^5$	≈	$1.92 \times 10^5$	$1.99 \times 10^3$	$1.95 \times 10^5$	$1.88 \times 10^5$	≈
算法	变体场景 2a:24 任务/4 无人机					变体场景 2b:24 任务/8 无人机				
	Mean	Std	Best	Worst	Wilcoxon	Mean	Std	Best	Worst	Wilcoxon
AAUC-IBEA	$4.87 \times 10^5$	$1.58 \times 10^5$	$7.24 \times 10^5$	$1.51 \times 10^5$	++	$4.74 \times 10^5$	$3.08 \times 10^4$	$5.59 \times 10^5$	$4.22 \times 10^5$	++
CCPSO	$6.44 \times 10^5$	$1.06 \times 10^5$	$8.63 \times 10^5$	$3.24 \times 10^5$	++	$4.83 \times 10^5$	$1.62 \times 10^4$	$5.13 \times 10^5$	$4.42 \times 10^5$	++
MODABC	$7.01 \times 10^5$	$1.08 \times 10^5$	$8.76 \times 10^5$	$4.69 \times 10^5$	++	$5.19 \times 10^5$	$1.86 \times 10^4$	$5.49 \times 10^5$	$4.69 \times 10^5$	++
SLPSO	$6.11 \times 10^5$	$1.62 \times 10^5$	$8.60 \times 10^5$	$1.50 \times 10^5$	++	$4.95 \times 10^5$	$2.08 \times 10^4$	$5.38 \times 10^5$	$4.50 \times 10^5$	++
ACO	$4.33 \times 10^5$	$2.05 \times 10^5$	$6.81 \times 10^5$	$2.12 \times 10^5$	++	$4.45 \times 10^5$	$4.12 \times 10^4$	$5.05 \times 10^5$	$3.81 \times 10^5$	++
PILPSO	$7.88 \times 10^5$	$8.15 \times 10^4$	$9.28 \times 10^5$	$6.06 \times 10^5$	≈	$5.30 \times 10^5$	$1.37 \times 10^4$	$5.58 \times 10^5$	$4.90 \times 10^5$	≈
算法	变体场景 3a:48 任务/8 无人机					变体场景 3b:48 任务/16 无人机				
	Mean	Std	Best	Worst	Wilcoxon	Mean	Std	Best	Worst	Wilcoxon
AAUC-IBEA	$8.56 \times 10^5$	$3.11 \times 10^5$	$1.38 \times 10^6$	$1.88 \times 10^5$	++	$7.82 \times 10^5$	$6.79 \times 10^4$	$9.04 \times 10^5$	$6.67 \times 10^5$	++
CCPSO	$1.01 \times 10^6$	$2.70 \times 10^5$	$1.53 \times 10^6$	$4.91 \times 10^5$	++	$8.35 \times 10^5$	$4.17 \times 10^4$	$9.27 \times 10^5$	$7.59 \times 10^5$	++
MODABC	$1.37 \times 10^6$	$4.74 \times 10^5$	$2.49 \times 10^6$	$6.62 \times 10^5$	++	$9.33 \times 10^5$	$4.32 \times 10^4$	$1.02 \times 10^6$	$8.28 \times 10^5$	++
SLPSO	$1.01 \times 10^6$	$4.15 \times 10^5$	$1.99 \times 10^6$	$2.34 \times 10^5$	++	$9.01 \times 10^5$	$5.87 \times 10^4$	$1.02 \times 10^6$	$7.84 \times 10^5$	++
ACO	$6.88 \times 10^5$	$5.15 \times 10^5$	$1.15 \times 10^6$	$1.31 \times 10^5$	++	$7.16 \times 10^5$	$9.33 \times 10^4$	$8.54 \times 10^5$	$5.98 \times 10^5$	++
PILPSO	$2.22 \times 10^6$	$3.86 \times 10^5$	$2.79 \times 10^6$	$1.22 \times 10^6$	≈	$9.67 \times 10^5$	$3.87 \times 10^4$	$1.07 \times 10^6$	$9.11 \times 10^5$	≈

## 5 结束语

本文研究了任务信息不确定和多约束的无人机集群协同任务分配问题。首先,明确了不确定信息的表示,并设计了不确定到精确信息的转换方法;其次,综合考虑任务、无人机和环境约束,建立了协同任务分配的数学模型。针对该模型,提出了基于种群交互学习的

粒子群算法。该算法通过问题引导的种群初始化,生成高质量初始解,并通过基于学习库的粒子更新和局部优化策略,以及双种群交互进化、信息共享与协同作用,有效提升了收敛效率,并改善了帕累托前沿的分布特性。实验结果显示,本文所提方法能具有高效性,还具备良好的鲁棒性和稳定性。未来,将进一步考虑任

务、环境及无人机引入的不确定性,构建模型并改进算法,以提升求解效率。

#### 参考文献

- [1] SKALTSIS G M, SHIN H S, TSOURDOS A. A review of task allocation methods for UAVs[J]. *Journal of Intelligent & Robotic Systems*, 2023, 109(4): 76.
- [2] LI L, CHEN Z Y, WANG H, et al. Fast task allocation of heterogeneous robots with temporal logic and inter-task constraints[J]. *IEEE Robotics and Automation Letters*, 2023, 8(8): 4991-4998.
- [3] 陈晟宗, 张纪会, 于守水, 等. 求解旅行商问题的波动温控模拟退火算法[J]. *控制与决策*, 2023, 38(4): 911-920.  
CHEN S Z, ZHANG J H, YU S S, et al. A simulated annealing algorithm with wave temperature control for the traveling salesman problem[J]. *Control and Decision*, 2023, 38(4): 911-920. (in Chinese)
- [4] KANG W, JEONG E, SHIM S, et al. Optimization of task allocation for resource-constrained swarm robots[J]. *IEEE Transactions on Automation Science and Engineering*, 2025, 22: 3068-3085.
- [5] WEN C X, MA H B. An efficient two-stage evolutionary algorithm for multi-robot task allocation in nuclear accident rescue scenario[J]. *Applied Soft Computing*, 2024, 152: 111223.
- [6] WANG G H, WANG F M, WANG J H, et al. Collaborative target assignment problem for large-scale UAV swarm based on two-stage greedy auction algorithm[J]. *Aerospace Science and Technology*, 2024, 149: 109146.
- [7] CAO Y, LONG T, SUN J L, et al. Comparison of distributed task allocation algorithms considering non-ideal communication factors for multi-UAV collaborative visit missions[J]. *IEEE Robotics and Automation Letters*, 2025, 10(2): 1928-1935.
- [8] 李鑫滨, 郭力争, 韩松. 一种分布式异构多AUV任务分配鲁棒拍卖算法[J]. *北京航空航天大学学报*, 2022, 48(5): 736-746.  
LI X B, GUO L Z, HAN S. A robust auction algorithm for distributed heterogeneous multi-AUV task assignment[J]. *Journal of Beijing University of Aeronautics and Astronautics*, 2022, 48(5): 736-746. (in Chinese)
- [9] WU X J, GAO Z Y, YUAN S, et al. A dynamic task allocation algorithm for heterogeneous UUV swarms[J]. *Sensors*, 2022, 22(6): 2122.
- [10] LIU C X, LU K H, CHEN X J, et al. Game-theoretical approach for task allocation problems with constraints[J]. *Applied Mathematics and Computation*, 2023, 458: 128251.
- [11] HUO X, ZHANG H, HUANG C, et al. Task allocation with minimum requirements for multiple mobile robot systems: A game-theoretical approach[J]. *IEEE Transactions on Network Science and Engineering*, 2024, 11(1): 1202-1213.
- [12] 刘志华, 张冉, 郝梦男, 等. 基于改进T分布烟花-粒子群算法的AUV全局路径规划[J]. *电子学报*, 2024, 52(9): 3123-3134.  
LIU Z H, ZHANG R, HAO M N, et al. AUV global path panning based on improved T-distribution fireworks-particle swarm optimization algorithm[J]. *Acta Electronica Sinica*, 2024, 52(9): 3123-3134. (in Chinese)
- [13] 王巍, 解慧, 魏忠诚, 等. 不确定需求下无人机任务分配的两阶段鲁棒优化方法[J]. *电子学报*, 2024, 52(10): 3552-3561.  
WANG W, XIE H, WEI Z C, et al. Two-stage robust optimization method for UAV task assignment under uncertain demand[J]. *Acta Electronica Sinica*, 2024, 52(10): 3552-3561. (in Chinese)
- [14] TAN Y F, ZHOU C, QIAN F. Cooperative task allocation method for multi-unmanned aerial vehicles based on the modified genetic algorithm[J]. *IET Intelligent Transport Systems*, 2024, 18(6): 1164-1173.
- [15] SHI J Q, TAN L, LIAN X F, et al. A multi-unmanned aerial vehicle dynamic task assignment method based on bionic algorithms[J]. *Computers and Electrical Engineering*, 2022, 99: 107820.
- [16] VAN ZYL J P, ENGELBRECHT A P. Set-based particle swarm optimisation: A review[J]. *Mathematics*, 2023, 11(13): 2980.
- [17] WU Y, LIANG T J, GOU J Z, et al. Heterogeneous mission planning for multiple UAV formations via meta-heuristic algorithms[J]. *IEEE Transactions on Aerospace and Electronic Systems*, 2023, 59(4): 3924-3940.
- [18] 孙靖. 用于区间参数多目标优化问题的遗传算法[D]. 徐州: 中国矿业大学, 2012.  
SUN J. Genetic Algorithms for Solving: Multi-Objective Optimization Problems with Interval Parameters[D]. Xuzhou: China University of Mining and Technology, 2012. (in Chinese)
- [19] XIONG J, LI J, LI J, et al. Probability-tuned market-based allocations for UAV swarms under unreliable observations[J]. *IEEE Transactions on Cybernetics*, 2023, 53(11): 6803-6814.
- [20] 李媛媛, 陈建国, 张小乐, 等. 基于建筑结构破坏的地震

伤亡评估方法及应用[J]. 清华大学学报(自然科学版), 2015, 55(7): 803-807, 814.

LI Y Y, CHEN J G, ZHANG X L, et al. Structure destruction based earthquake casualty estimates[J]. Journal of Tsinghua University (Science and Technology), 2015, 55(7): 803-807, 814. (in Chinese)

- [21] 耿娜, 饶凌风, 张勇, 等. 基于种群交互学习式粒子群算法的无人机集群协同救援任务分配方法: CN119443626 A[P]. 2025-02-14.
- [22] WILCOXON F. Individual comparisons by ranking methods[M]//Breakthroughs in Statistics. New York: Springer New York, 1992: 196-202.
- [23] WEN C X, MA H B. An indicator-based evolutionary algorithm with adaptive archive update cycle for multi-ob-

jective multi-robot task allocation[J]. Neurocomputing, 2024, 593: 127836.

- [24] DAI L L, PAN Q K, MIAO Z H, et al. Multi-objective multi-picking-robot task allocation: Mathematical model and discrete artificial bee colony algorithm[J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(6): 6061-6073.
- [25] 王世科, 游晓明, 尹玲, 等. 融合随机趋邻策略的协同演化蚁群算法[J]. 计算机集成制造系统, 2025, 31(2): 697-710.
- WANG S K, YOU X M, YIN L, et al. Co-evolution ant colony system based on random convergent neighborhood strategy[J]. Computer Integrated Manufacturing Systems, 2025, 31(2): 697-710. (in Chinese)

### 作者简介



饶凌风 男, 2001年1月出生于江苏省徐州市. 现为中国矿业大学信息与控制工程学院博士研究生. 主要研究方向为智能优化及其应用.  
E-mail: raodawang@126.com



郭广颂 男, 1978年9月出生于黑龙江省. 2007年毕业于中国矿业大学控制理论与控制工程专业, 现为郑州航空工业管理学院教授. 主要研究方向为无人机任务分配.  
E-mail: guogs78@126.com



耿娜 女, 1985年5月生于江苏省徐州市. 2016年毕业于中国矿业大学控制理论与控制工程专业, 现为江苏师范大学副教授. 主要研究方向为任务分配.  
E-mail: gengna@126.com



肖磊 男, 1984年11月出生于江苏省徐州市. 2011年毕业于江苏科技大学材料学专业, 现为中国自行车协会高级工程师. 主要研究方向为机器人.  
E-mail: 372119639@qq.com



张勇 男, 1979年9月出生于山东省莱芜市. 2009年毕业于中国矿业大学控制理论与控制工程专业, 现为中国矿业大学教授. 主要研究方向为机器人路径规划与任务分配.  
E-mail: yongzh401@126.com