doi:10.6041/j.issn.1000-1298.2025.06.010

基于改进分数阶粒子群算法的多无人车取送货任务调度方法

陈玉全　冯丽曼　孙克璇　张楠杰　王　冰
(河海大学人工智能与自动化学院, 常州 213000)

摘要: 针对农产品运输场景下产地与销地配送环节中的多无人车协同任务分配问题,首先构建涵盖行程成本、时间违反成本、负载违反成本和启动成本的多无人车取送货任务调度组合优化模型。提出一种改进分数阶粒子群算法(Improved fractional order particle swarm optimization,IFOPSO)。通过在粒子群算法(PSO)中引入分数阶列维随机步长,提高 PSO 的全局搜索能力,进一步设计列维阶次的自适应调整机制,提高 IFOPSO 的收敛精度和寻优性能。基于 10 个基准函数的对比实验结果表明,提出的 IFOPSO 算法在收敛速度、精度以及全局搜索能力等方面,相较于现有算法表现出显著优势。最后将 IFOPSO 算法应用于多无人车任务分配问题的求解中,并与传统 PSO、改进 PSO 和分数阶 PSO 算法进行对比实验,结果表明该算法能够有效降低调度成本,并快速找到合理的取送货方案。

关键词: 农产品运输; 任务分配; 多车协同; 分数阶粒子群算法; 列维随机步长; 自适应列维阶次
中图分类号: TP301.6; F251.2　　**文献标识码**: A　　**文章编号**: 1000-1298(2025)06-0109-10　　**OSID**:

Task Scheduling for Multi-unmanned Vehicle Delivery Using Improved Fractional-order Particle Swarm Optimization Algorithm

CHEN Yuquan　FENG Liman　SUN Kexuan　ZHANG Nanjie　WANG Bing
(*College of Artificial Intelligence and Automation*, *Hohai University*, *Changzhou* 213000, *China*)

Abstract: Aiming to handle the problem of multi-unmanned vehicle task allocation in agricultural product transportation scenarios between production sites (farms) and distribution sites (markets), a novel combinatorial optimization model that incorporated delivery time requirements, vehicle constraints, and task complexity was established at first. Subsequently, an improved fractional order particle swarm optimization (IFOPSO) algorithm was proposed. By introducing a fractional-order Lévy random step size into the particle swarm optimization (PSO) algorithm, the global search capability was significantly enhanced. Additionally, a mechanism for adaptively adjusting the Lévy order was designed to improve the convergence accuracy, robustness, and overall optimization performance of IFOPSO. Experimental results based on ten benchmark functions demonstrated that the proposed IFOPSO algorithm exhibited significant advantages in terms of convergence speed, accuracy, and global search ability compared with existing algorithms. Furthermore, an optimization model for unmanned vehicle pickup and delivery task scheduling was developed, where the total cost accounted for travel cost, time violation cost, load violation cost, and start-up cost. The IFOPSO algorithm was applied to solve this task allocation problem, and comparative experiments with traditional PSO, improved PSO, and fractional order PSO algorithms showed that the proposed algorithm effectively reduced scheduling costs, improved solution efficiency, and rapidly identified a feasible and optimal pickup and delivery solution.

Key words: transportation of agricultural products; task assignment; multi-vehicle collaboration; fractional-order particle swarm optimization; Lévy random step size; adaptive Lévy order

0　引言

随着无人系统技术的快速发展,无人机、无人车和无人艇在实际应用中的优势愈加显著,尤其是在农产品运输等场景中,基于无人系统的自动化运输已经成为提高效率、减少人为干扰、降低成本的有效

收稿日期: 2024-11-11　修回日期: 2024-12-05
基金项目: 国家自然科学基金项目(51777058)
作者简介: 陈玉全(1992—),男,副教授,博士,主要从事智能控制和分数阶优化算法研究,E-mail: cyq@mail.ustc.edu.cn

途径[1-2]。然而,随着运输任务的复杂性增加,尤其是在面对多点、多批次、多约束的任务时,传统的单一无人车系统在资源调度、路径规划和任务执行等方面的局限性日益突出[3]。

相比之下,多无人车系统通过合理的任务分配和资源整合,能够在更大规模和更复杂的运输场景中展现出更高的效率和可靠性[4]。然而,这也使得任务分配和调度问题变得更加复杂。特别是在多约束环境下,如何合理分配任务以确保运输时效性、提升系统整体性能、实现多车协同,成为一个亟待解决的问题[5-6]。

在农产品运输任务中,时间约束、负载能力、续航能力等因素尤为关键,因此,如何在多车协同任务分配中合理考虑这些实际需求,是一个具有挑战性的课题。尽管现有研究在路径优化和负载均衡方面取得了显著进展[7-9],但对任务分配中时间约束的处理尚未得到充分重视,尤其是在涉及多车同时执行多个任务的动态场景下[10-11]。

任务分配问题通常可以归纳为混合整数规划问题,而智能优化算法由于其高效性和灵活性,已经成为解决该问题的有效工具[12]。粒子群优化(PSO)算法,因其较强的全局搜索能力和低计算复杂性,已被广泛应用于多无人系统的任务分配中[13-14]。然而,传统 PSO 算法在面对复杂场景时,可能陷入局部最优解,导致搜索效率下降。为此,一些改进的 PSO 算法[15-16]尝试引入更高效的全局搜索策略,但在复杂场景下仍然存在粒子多样性不足、局部最优解问题较为突出等挑战。

因此,本文提出一种改进的分数阶粒子群优化算法(IFOPSO),旨在通过引入分数阶列维随机步长增强粒子的跳跃性,改善全局搜索能力,从而提高任务分配的效率与精度。与传统 PSO 算法相比,IFOPSO 能够更好地平衡全局搜索与局部收敛的关系,并有效避免陷入局部最优解。此外,还设计一种自适应机制,动态调整列维步长的阶次,使得搜索过程能够根据迭代进度不断优化。通过改进算法提升全局优化能力,为农产品运输任务中的多无人车协同调度提供一种更为高效的解决方案。

1 问题描述与建模

1.1 问题描述

研究了农产品运输场景下产地与销地配送环节中的多无人车协同任务分配问题[17]。如图 1 所示,问题的具体描述为:在某一地区,农贸市场位于市区,多个小型散户分布在农村,且没有大规模运输需求。无人车从农贸市场出发,依据各任务点的取货时间窗(最早取货时间与最晚取货时间),依次前往散户处取货,完成任务后返回市场。假设所有农产品(如蔬菜、水果、家禽等)具有相容性,可一起装载运输,无需冷藏。本研究聚焦于该场景下的多无人车协同任务分配问题,目标是在满足时间窗、负载限制等约束条件下,优化任务调度与资源消耗,快速、高效地将农产品从农村散户运输至市区农贸市场。研究旨在设计高效的调度与协同机制,解决中短距离运输问题,确保农产品在运输时间短、资源有限的情况下实现高效配送。

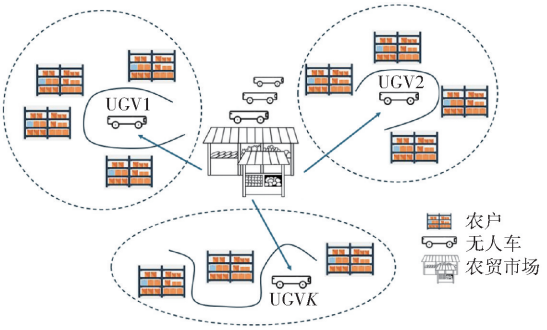

图 1 多无人车任务分配示意图

Fig. 1 Diagrammatic sketch for task allocation problem of multiple unmanned vehicles

该农贸市场共 K 辆型号相同的无人车,共 n 户需要取送货的农户,设无人车集合 $U=\{U_1,U_2,\cdots,U_K\}$,任务点集合 $T=\{T_1,T_2,\cdots,T_n\}$。

本文将行程总代价、时间要求代价、违反最大负载代价以及无人车启动代价作为多无人车任务调度的评价指标。因此,农产品运输场景下产地与销地配送环节中的多无人车协同任务分配问题的数学模型为

$$\min F=f_1+f_2+f_3+f_4 \tag{1}$$

式中,f_1、f_2、f_3、f_4 是行程总代价、违反取货时间总代价、违反最大负载总代价和无人车启动总代价。

1.2 行程总代价函数

行程是指无人车从农贸市场出发到完成所有任务点的取送货返回农贸市场的行驶总距离。这是评价多无人车协同任务分配过程中资源消耗的关键因素。行程总代价函数为

$$f_1=\sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=0}^{K}d_{ij}x_{ijk} \tag{2}$$

式中 d_{ij}——无人车 U_k 从任务点 T_i 到任务点 T_j 的行驶距离

x_{ijk}——决策变量,表示无人车 U_k 是否完成从任务点 T_i 到任务点 T_j 的路段行驶

1.3 时间总代价函数

农产品在当天的最早取货时间前采摘准备完毕,无人车取货时间要求不能早于最早取货时间,无

人车到达不同农户家后需要不同的装载时间取货，为保证农产品能及时送达农贸市场，同样需要无人车取货时间不能晚于最晚取货时间，即无人车必须在一定时间内完成取货任务。违反取货时间总代价函数为

$$f_2 = c_z \sum_{i=1}^{n} t_{i_{zz}} + c_w \sum_{i=1}^{n} t_{i_{zw}} + \sum_{i=1}^{n} t_{i_{wc}} \tag{3}$$

其中

$$\begin{cases} t_{i_{zz}} = \max(ET_i - S_i, 0) \\ t_{i_{zw}} = \max(WT_i - S_i, 0) \end{cases} \tag{4}$$

式中　c_z——无人车早于最早取货时间到达任务点的惩罚系数

c_w——无人车晚于最晚取货时间到达任务点的惩罚系数

$t_{i_{wc}}$——在取货点 i 处的服务时间

$t_{i_{zz}}$——无人车到达任务点 T_i 的早到时间

$t_{i_{zw}}$——无人车到达任务点 T_i 的晚到时间

S_i——无人车到达任务点 T_i 的时间

ET_i——任务点 T_i 的最早服务时间

WT_i——任务点 T_i 的最晚服务时间

1.4　违反最大负载总代价

最大负载是由无人车物理特性决定的最大载货量，无人车所服务的所有任务点中的物理总代价为

$$f_3 = c_3 \sum_{k=1}^{K} \sum_{i=1}^{n} \sum_{j=1}^{n} q_{kv} x_{ijk} \tag{5}$$

式中　c_3——违反无人车最大负载的惩罚系数

q_{kv}——无人车 U_k 所服务的整段路径中超过最大负载的货物重量

1.5　无人车启动总代价

每辆无人车使用时需要相同的启动资金，投入使用的无人车数量越多，任务执行效率越高，但无人车启动代价越大；反之，任务效率较低，启动代价较小，需要根据实际情况兼顾启动代价和任务完成效率，无人车启动总代价为

$$f_4 = AK_k \tag{6}$$

式中　A——每辆无人车的启动代价

K_k——实际任务执行过程中使用的无人车数量

1.6　约束函数

1.6.1　决策变量约束

一般任务点的数量多于无人车的数量，为完成取送货任务，要求无人车能完成所有任务点任务，且每个任务点的取送货仅由一辆无人车完成。

$$\sum_{i=1}^{n} \sum_{k=1}^{K} y_{ik} = 1 \tag{7}$$

其中

$$y_{ik} = \begin{cases} 1 & (\text{任务点 } T_i \text{ 由无人车 } U_k \text{ 取送货}) \\ 0 & (\text{任务点 } T_i \text{ 不由无人车 } U_k \text{ 取送货}) \end{cases} \tag{8}$$

式中　y_{ik}——决策变量，表示任务点 T_i 是否由无人车 U_k 完成取货

决策变量 x_{ijk} 满足

$$\sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{K} x_{ijk} = 1 \tag{9}$$

其中

$$x_{ijk} = \begin{cases} 1 & (\text{无人车 } U_k \text{ 从任务点 } T_i \text{ 行驶到任务点 } T_j) \\ 0 & (\text{无人车 } U_k \text{ 不从任务点 } T_i \text{ 行驶到任务点 } T_j) \end{cases} \tag{10}$$

1.6.2　最大负载约束

无人车所能携带的农产品是有限的，其所能装载的农产品数量不能超过其最大容量，满足

$$\sum_{k=1}^{K} \sum_{i=1}^{n} q_i y_{ik} \leqslant Q_k \tag{11}$$

式中　q_i——每个任务点的取货量

Q_k——无人车 U_k 的最大容量

1.6.3　最大行程约束

由于无人车物理特性约束，每辆无人车所能行驶的最大行程距离是有限的，为保证安全和取送货正常，要求无人车总行驶距离不能超过其自身限定的最大行程，满足

$$\sum_{k=1}^{K} \sum_{i=0}^{n} \sum_{j=0}^{n} d_{ij} x_{ijk} \leqslant L_{\max} \tag{12}$$

式中　$L_{\max}$——每辆无人车限定的最大行程

1.6.4　时间窗约束

考虑实际取送货时间要求，每个任务点的取货需要在要求的最早取货时间后及最晚取货时间前被相应无人车完成，满足

$$\sum_{i=1}^{n} t_{i_{zz}} = 0 \tag{13}$$

$$\sum_{i=1}^{n} t_{i_{zw}} = 0 \tag{14}$$

2　改进分数阶粒子群算法

标准粒子群算法在处理大规模搜索空间或复杂问题时，常常陷入局部最优解，导致全局搜索能力不足。列维随机步长具有非高斯性和重尾分布的优点，能够有效增强算法的全局探索能力，帮助粒子跳出局部最优。因此，提出了一种改进的分数阶粒子群算法（Improved fractional order PSO，IFOPSO）来求解本文提出的优化模型。在该算法中，列维随机步长通过列维飞行机制代替传统的高斯分布步长，结合大跳跃和小步进的特点，使粒子能够更灵活地进行全局搜索，避免陷入局部最优。传统的分数阶粒子群算法阶次一般固定，其一般出现的大步长跳跃可以有效提升算法的全局搜索能力，避免陷入局部

最优。但是固定阶次的分数阶粒子群算法会对后期的搜索精度产生影响,因此进一步设计分数阶列维阶次自适应调整机制。IFOPSO 引入自适应进化机制,通过动态调整粒子的列维阶次,有效平衡全局探索与局部收敛,提升算法的搜索效率和解的精度。通过这些改进,IFOPSO 在多个高维测试函数上进行了验证,结果表明该算法在复杂、高维问题中具有良好的收敛性和鲁棒性。

2.1 粒子群算法

粒子的速度更新公式和位置更新公式为

$$v_{i,d}=\omega v_{i,d-1}+c_1r_1(p_{i,d}-x_{i,d})+c_2r_2(g_d-x_{i,d}) \tag{15}$$

$$X_i(t+1)=X_i(t)+V_i(t+1) \tag{16}$$

式中 $v_{i,d}$——粒子 i 在第 d 个维度上的速度

ω——惯性权重

$p_{i,d}$——粒子 i 在第 d 个维度上的个体历史最佳位置

$x_{i,d}$——粒子 i 在第 d 个维度上的当前位置

g_d——整个群体在第 d 个维度上的历史最佳位置

c_1——个体学习因子

c_2——全局学习因子

r_1——0 ~ 1 之间的随机数

r_2——0 ~ 1 之间的随机数

$X_i(t)$——粒子 i 在 t 时刻的位置

$X_i(t+1)$——粒子 i 在 $t+1$ 时刻的位置

惯性权重 ω 是粒子保留上一次速度的程度,在式(15)影响参数中 ω 对更新结果影响最大。随着迭代次数的增加,一般会逐渐降低惯性权重 ω,让粒子群在开始时保持较高的探索性,然后再随着迭代次数增加,降低 ω 从而降低粒子的探索性,粒子逐渐收敛到局部最优解,实现更加精确的搜索[18]。ω 计算公式为

$$\omega=\omega_{\max}-(\omega_{\max}-\omega_{\min})\frac{n_{gen}}{n_{gen,\max}} \tag{17}$$

式中 $\omega_{\max}$——惯性权重最大值

$\omega_{\min}$——惯性权重最小值

n_{gen}——当前迭代次数

$n_{gen,\max}$——最大迭代次数

惯性权重 ω 不再是定值,而是随着迭代次数线性递减,平衡全局搜索和局部搜索,提高算法的搜索效率。加速因子 c_1、c_2 分别用于调节个体最优粒子和全局最优粒子的行驶速度,c_1 越大,全局搜索能力越强;c_2 越大,局部搜索能力越强。故对加速因子 c_1 和 c_2 分别采用线性递减和线性递增的策略动态调整[18]。

$$c_1=c_{1\max}-\frac{c_{1\max}-c_{1\min}}{n_{gen_{\max}}}n_{gen} \tag{18}$$

$$c_2=c_{2\min}+\frac{c_{2\max}-c_{2\min}}{n_{gen,\max}}n_{gen} \tag{19}$$

式中 $c_{1\max}$——个体加速因子最大值

$c_{1\min}$——个体加速因子最小值

$c_{2\max}$——全局加速因子最大值

$c_{2\min}$——全局加速因子最小值

2.2 列维分布

标准粒子群算法在处理多维问题时,粒子的每一步位移通常遵循均匀分布或高斯分布,难以实现大步长跳跃,因此容易陷入局部最优,导致算法过早收敛。然而,在优化过程中,少部分优质解往往分布在远离当前最优解的区域,标准粒子群算法难以捕捉这些解。列维分布是一种长尾分布,具有频繁的短跳跃和偶然的长跳跃特性,如图 2 所示,使得粒子在空间中可能进行较大的跳跃。列维飞行能够捕捉不寻常或稀有事件,增强全局搜索能力。分数阶列维随机步长的优势在于其多次大步长扰动,有助于提升全局收敛性能。因此,本文考虑将传统粒子群算法的均匀分布随机步长替换为列维分布随机步长,以提高全局收敛能力。

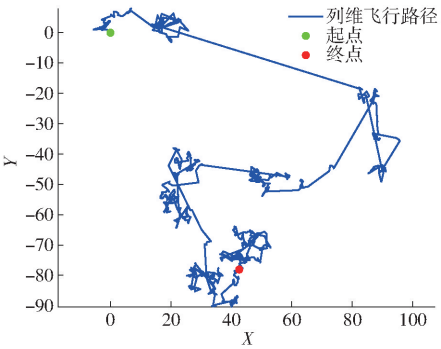

图 2 列维随机步长

Fig. 2 Lévy random step size

一个随机变量 Y 满足 $0\leqslant\alpha\leqslant2$ 的列维稳定分布是指其特征函数满足形式[19]

$$E[e^{ikY}]=\begin{cases}\exp\left(i k\mu-\sigma^{\alpha}|k|^{\alpha}\left(1-i\beta\operatorname{sgn}(k)\tan\left(\frac{\pi\alpha}{2}\right)\right)\right) & (\alpha\neq1)\\ \exp\left(i k\mu-\sigma|k|\left(1-i\beta\left(\frac{\pi}{2}\right)\operatorname{sgn}(k)\ln(k)\right)\right) & (\alpha=1)\end{cases} \tag{20}$$

式中 μ——位置参数,表示列维稳定分布的平均值

β——对称参数,是列维分布的倾斜度,通常 $-1\leqslant\beta\leqslant1$,$\beta=0$ 时为对称分布

α——特征指数,α 越小则脉冲性越强,越容易偏离中心位置,$\alpha=2$ 时对应于高斯分布

σ——尺度参数，影响分布的“宽度”，通常 $\sigma>0$

$\mathrm{sgn}(k)$——符号函数

可以用 $S_\alpha(\mu,\beta,\sigma)$ 表示一个列维稳定分布，即列维分布。图 3 展示了 350 个随机取值下的列维分布和均匀分布的对比。

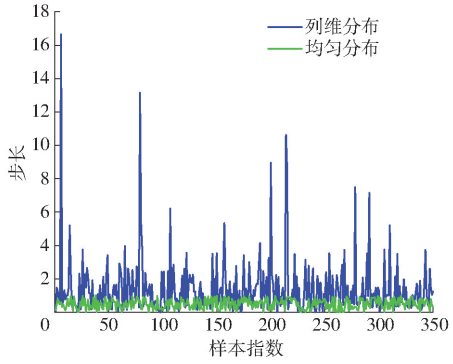

图 3　列维分布步长和均匀分布步长变化曲线

Fig. 3　Variation curves of step size of dimensionality distribution and uniform distribution step size

2.3　参数自适应调整

由上述列维分布可知，α 是特征指数（即列维阶次），α 越小脉冲性越强，粒子越容易出现大步长，寻优前期需要较小 α，以充分探寻整个空间，寻优后期尤其是找到全局最优时，应适当提高 α 值，使粒子前期遵循列维分布（$\alpha<2$），后期遵循高斯分布（$\alpha=2$）。列维分布前期选定 α 会产生较大的脉冲响应，且 α 越小脉冲越强，此时可以提升全局搜索能力，但是无法兼顾后期的搜索精度。因此，设计自适应阶次调整机制

$$\alpha=\alpha_{\min}+(\alpha_{\max}-\alpha_{\min})\frac{n_{gen}}{n_{gen,\max}} \tag{21}$$

式中　$\alpha_{\min}$——列维阶次最小值

$\alpha_{\max}$——列维阶次最大值

$\alpha_{\min}$ 一般取 1～2 之间的值，$\alpha_{\max}$ 取 2。随着迭代的次数增加，列维阶次逐步增加到 2，它的脉冲强度逐步减弱，从而达到前期增强全局搜索能力，后期兼顾搜索精度的效果。在实际中，需要依据搜索范围确定 $\alpha_{\min}$ 值，当搜索范围较小时可以适当增加 $\alpha_{\min}$ 值，当搜索范围较大时可以适当减小 $\alpha_{\min}$ 值。

2.4　各类算法在基准函数中的对比与分析

为验证本文提出的 IFOPSO 算法正确性和有效性，比较粒子群算法（Particle swarm optimization，PSO）[20]、遗传算法（Genetic algorithm，GA）[21]、布谷鸟算法（Cuckoo search，CS）[22]、灰狼算法（Grey wolf optimizer，GWO）[23]、自适应参数粒子群算法（Improved particle swarm optimization，IPSO）、分数阶粒子群算法（Fractional order particle swarm optimization，FOPSO）、改进分数阶粒子群算法（Improved fractional-order particle swarm optimization，IFOPSO）对 10 个基准函数（见表 1）进行测试。设置各算法种群大小为 50，迭代次数为 1 000，各标准群优化算法参数取自对应文献。IPSO 算法中参数设定为惯性权重最大值 $\omega_{\max}=0.9$，惯性权重最小值 $\omega_{\min}=0.4$，个体学习因子最大值 $c_{1\max}=2.5$，个体学习因子最小值 $c_{1\min}=0.5$，全局学习因子最大值 $c_{2\max}=2.5$，全局学习因子最小值 $c_{2\min}=0.5$。FOPSO 算法中参数设定为惯性权重 $\omega=0.7$，个体学习因子 $c_1=1.5$，全局学习因子 $c_2=1.5$，列维阶次 $\alpha=1.75$。IFOPSO 算法参数设定为惯性权重最大值 $\omega_{\max}=0.9$，惯性权重最小值 $\omega_{\min}=0.4$，个体学习因子最大值 $c_{1\max}=2.5$，个体学习因子最小值 $c_{1\min}=0.5$，全局学习因子最大值 $c_{2\max}=2.5$，全局学习因子最小值 $c_{2\min}=0.5$，列维阶次最大值 $\alpha_{\max}=2$，列维阶次最小值 $\alpha_{\min}=1.5$。但在实际工程中应根据实验环境不同适当调整参数，以达到最优效果。考虑减小误差，选取各算法独立运行 30 次的均值和标准差为测试结果。

表 1　基准函数

Tab. 1　Benchmark functions

函数名称	函数表达式	取值范围
Cube 函数	$f_1(\boldsymbol{x})=100(x_2-x_1^3)^2+(1-x_1)^2$	$(-10,10)$
Rotated Ellipse 函数	$f_2(\boldsymbol{x})=x_1^2-x_1x_2+x_2^2$	$(-500,500)$
Alpine 函数	$f_3(\boldsymbol{x})=\sum_{i=1}^{D}\lvert x_i\sin(x_i)+0.1x_i\rvert$	$(-10,10)$
Chung Reynolds 函数	$f_4(\boldsymbol{x})=\left(\sum_{i=1}^{D}x_i^2\right)^2$	$(-100,100)$
Powell Sum 函数	$f_5(\boldsymbol{x})=\sum_{i=1}^{D}\lvert x_i\rvert^{i+1}$	$(-1,1)$
Qing 函数	$f_6(\boldsymbol{x})=\sum_{i=1}^{D}(x_i^2-i)^2$	$(-500,500)$
Quintic 函数	$f_7(\boldsymbol{x})=\sum_{i=1}^{D}\lvert x_i^5-3x_i^4+4x_i^3+2x_i^2-10x_i-4\rvert$	$(-10,10)$
Schumer Steiglitz 函数	$f_8(\boldsymbol{x})=\sum_{i=1}^{D}x_i^4$	$(-10,10)$
Schwefel 函数	$f_9(\boldsymbol{x})=\sum_{i=2}^{D}((x_i-1)^2+(x_1-x_i^2)^2)$	$(-500,500)$
Sum Squares 函数	$f_{10}(\boldsymbol{x})=\sum_{i=1}^{D}ix_i^2$	$(-10,10)$

如表 2 所示的实验对比结果表明，一维函数 f_1、f_2 平均值和标准差显著优于 4 种标准群优化算法，同时也优于自适应参数变化的粒子群算法和仅引入列维扰动的分数阶粒子群算法，这充分彰显了

表 2 基准函数算法寻优对比结果

Tab. 2 Comparison of optimization results of each algorithm on benchmark function

函数	指标	算法						
		PSO	GA	CS	GWO	IPSO	FOPSO	IFOPSO
f_1	平均值	$1.001\,99\times10^{-27}$	0.002 383 74	2.191 452 133	0.338 422 5	$7.818\,09\times10^{-22}$	4.139×10^{-12}	6.574×10^{-33}
	标准差	$5.441\,24\times10^{-27}$	0.007 899 575	1.692 768 424	0.184 918 517	$7.687\,68\times10^{-22}$	$2.260\,99\times10^{-11}$	$3.600\,73\times10^{-32}$
f_2	平均值	$3.115\,95\times10^{-6}$	5.375 786 7	7.519 023	9.506×10^{-102}	7.15×10^{-58}	$2.817\,92\times10^{-84}$	$2.881\,3\times10^{-218}$
	标准差	$9.768\,13\times10^{-6}$	18.990 512	10.952 15	$5.113\,2\times10^{-101}$	2.84×10^{-57}	$1.307\,51\times10^{-83}$	0
f_3	平均值	0.418 02	0.019 123	6.023 723	1.233 583	0.003 362	0.002 739 841	$8.510\,49\times10^{-45}$
	标准差	0.541 060 657	0.008 795	0.613 453	0.882 77	0.003 065	0.003 012 036	$1.733\,64\times10^{-44}$
f_4	平均值	1.586 021 323	3.249 565	0.116 268	0.037 785	$4.209\,4\times10^{-51}$	0.000 875 183	$2.140\,9\times10^{-125}$
	标准差	5.786 039 603	9.984 686	0.350 986	0.058 713	$2.288\,6\times10^{-50}$	0.003 323 857	$5.694\,7\times10^{-125}$
f_5	平均值	$3.315\,59\times10^{-6}$	0.002 384	0.427 276	0.338 423	4.03×10^{-31}	$9.419\,15\times10^{-51}$	$1.334\,96\times10^{-76}$
	标准差	$4.573\,82\times10^{-6}$	0.007 9	0.092 405	0.184 919	1.35×10^{-30}	$4.850\,86\times10^{-50}$	$3.207\,6\times10^{-76}$
f_6	平均值	0.007 121 774	333.110 2	43.125 74	36.334 24	9.07×10^{-30}	9.07×10^{-30}	9.07×10^{-30}
	标准差	0.027 263 244	10.481 04	10.224 12	18.434 84	0	0	0
f_7	平均值	3.125 602 667	0.302 382	30.132 62	1.204 098	3.235 658	$5.290\,81\times10^{-6}$	$3.400\,23\times10^{-14}$
	标准差	1.895 226 291	0.092 781	7.084 052	4.356 637	1.211 386	$4.966\,9\times10^{-6}$	$4.078\,79\times10^{-14}$
f_8	平均值	$5.669\,93\times10^{-11}$	0.011 166	2.220 148	0.033 653	4.44×10^{-6}	$6.767\,7\times10^{-8}$	$9.358\,4\times10^{-177}$
	标准差	$2.003\,14\times10^{-10}$	0.031 237	0.863 456	0.182 522	1.11×10^{-5}	$1.885\,86\times10^{-7}$	0
f_9	平均值	0	8.489 003	0	2.741 793	0	0	0
	标准差	0	0.388 824	0	2.229 769	0	0	0
f_{10}	平均值	$1.022\,47\times10^{-61}$	3.437 433	45.589 21	8.372 027	9.89×10^{-44}	$1.252\,27\times10^{-38}$	$1.476\,15\times10^{-64}$
	标准差	$3.366\,73\times10^{-61}$	9.617 141	8.286 183	13.883 57	2.27×10^{-43}	$4.170\,19\times10^{-38}$	$6.871\,68\times10^{-64}$

IFOPSO 具有更为出色的求解精度、求解速度以及鲁棒性。而在高维测试函数领域 f_3 ~ f_{10},IFOPSO 的求解能力达到了最优水准,在处理高维函数的过程中展现出了卓越的全局搜索能力,并且不易陷入局部最优的困境。IFOPSO 的寻优效果明显优于 PSO、GS、CS、GWO、IPSO 和 FOPSO 等算法,其主要原因在于自适应参数调整和分数阶列维扰动,使得本文所提出的 IFOPSO 算法能够有效地平衡全局和局部的寻优能力,进而呈现出令人满意的寻优效果。

2.5 IFOPSO 求解多无人车协同任务分配问题步骤

采用 IFOPSO 算法求解多无人车协同任务分配问题的过程如下:

(1)输入农贸市场和任务点信息,如位置坐标、取送货时间、取货量、服务时间等。

(2)初始化粒子群算法参数。设置种群大小 N,初始迭代次数 n_{gen},最大迭代次数 $n_{gen,\max}$,惯性权重下界 $\omega_{\min}$、上界 $\omega_{\max}$,个体学习因子下界 $c_{1\min}$、上界 $c_{1\max}$,全局学习因子下界 $c_{2\min}$、上界 $c_{2\max}$,分数阶列维阶次 α,列维阶次上界 $\alpha_{\max}$、下界 $\alpha_{\min}$。

(3)计录当前每个粒子的适应度函数值,并更新个体历史最佳位置。

(4)比较所有粒子的适应度值并更新全局最佳位置。

(5)由个体历史最佳位置和全局最佳位置,根据式(15)、(16)更新每个粒子的速度和位置。

(6)重复步骤(5),直到满足停止条件(达到最大迭代次数)。

3 基于 IFOPSO 的多无人车取送货任务调度与路径规划仿真实验

为验证所提出的 IFOPSO 算法求解农产品运输场景下的多无人车协同任务分配问题有效性,进行多组仿真实验,本文仿真环境为 Intel Core i5 CPU,处理器速度为 2.3 GHz,操作系统为 Windows 10,内存为 16 GB,编程环境为 Matlab R2020a。IFOPSO 算法中的参数为:粒子个数 N 为取货点数量、最大迭代次数 $n_{gen,\max}=300$、惯性权重最大值 $\omega_{\max}=0.9$、惯性权重最小值 $\omega_{\min}=0.4$、个体学习因子最大值 $c_{1\max}=2.5$、个体学习因子最小值 $c_{1\min}=0.5$、全局学习因子最大值 $c_{2\max}=2.5$、全局学习因子最小值 $c_{2\min}=0.5$、列维阶次最大值 $\alpha_{\max}=2$、列维阶次最小值 $\alpha_{\min}=1.5$。假设每辆 UAV 启动成本为 100 元,最大负载为 200 kg,最大行程为 50 km。农产品运输在 100 km × 100 km 范围内发生。

3.1 不同相对位置下的仿真实验结果对比

假设农贸市场有待投入使用的 20 辆无人车,30 个取货点,各个取货点取货量、要求取货时间均不变,农贸市场与取货点的位置发生改变,在 5 个相对位置不同的场景分布中进行实验,对比 PSO、

IPSO、FOPSO 和 IFOPSO 算法在 3 种场景下的结果。为降低算法中随机性的影响，每种算法独立运行30 次。

表 3 显示了不同相对位置下的对比实验结果，图 4 是不同相对位置下的收敛曲线。从 5 组相对位置不同的实验结果可以看出，虽然在位置 1、2、3、4 中，IFOPSO 算法的及时率和超载率相较于 PSO、IPSO、FOPSO 算法略低，并未在所有指标上均达到最优，但在规划得到的总代价上，IFOPSO 算法始终表现出最优结果，各组实验中的总代价差距不大但 IFOPSO 算法的值始终最低。仿真实验结果表明，IFOPSO 算法在不同相对位置的多无人车协同任务分配问题中能够有效降低总代价，找到更优的取送货方案，展示了其在综合性能上的优势。

表 3　不同相对位置下实验结果对比

Tab. 3　Comparison of experimental results at different relative positions

相对位置序号	评价标准	指标	PSO	IPSO	FOPSO	IFOPSO
1	总代价	平均值	2 291. 492 6	2 178. 759	2 225. 057 6	2 138. 529 5
		标准差	0	0	57. 886 4	48. 023 4
	及时率	平均值/%	93. 33	73. 33	80. 83	91. 33
		标准差	0	0	0. 075 6	0. 048 8
	超载率	平均值/%	10	6. 60	7. 16	6. 33
		标准差	0	0	0. 032 9	0. 034 0
2	总代价	平均值	1 232. 481 1	1 200. 836 0	1 214. 858 3	1 137. 674 0
		标准差	0	0	26. 159 3	14. 470 5
	及时率	平均值/%	100	100	97	99. 5
		标准差	0	0	0. 055 0	0. 022 3
	超载率	平均值/%	10	13. 33	9. 16	7. 66
		标准差	0	0	0. 022 3	0. 033 9
3	总代价	平均值	1 090. 899 3	1 076. 765 0	1 076. 410 8	1 070. 966 1
		标准差	0	0	0	0
	及时率	平均值/%	100	100	99. 33	100
		标准差	0	0	0	0
	超载率	平均值/%	6. 67	6. 67	6. 67	6. 67
		标准差	0	0	0	0
4	总代价	平均值	1 553. 378 9	1 594. 938 0	1 609. 034 5	1 523. 940 4
		标准差	0	0	60. 260 3	46. 234 7
	及时率	平均值/%	80	96. 67	86	92. 67
		标准差	0	0	0. 078 4	0. 071 4
	超载率	平均值/%	3. 33	0	7	6
		标准差	0	0	0. 026 2	0. 031 7
5	总代价	平均值	1 535. 527 8	1 641. 227 4	1 602. 085 8	1 496. 214 6
		标准差	0	0	55. 596 4	46. 265 7
	及时率	平均值/%	86. 67	86. 67	76. 67	90
		标准差	0	0	0. 095 7	0. 070 1
	超载率	平均值/%	6. 67	6. 67	7. 16	5. 33
		标准差	0	0	0. 027 1	0. 034 8

3. 2　不同数量取货点下的仿真实验对比

取货点的数量直接决定了粒子的维度，随着取货点数量的增加，粒子维度不断增大，使得寻优的难度逐步上升。进一步深入探讨 IFOPSO 算法在不同取货点数量下处理多无人车协同任务分配问题时的性能表现。实验设置为固定 40 辆无人车，并分别针对取货点数量为 20、30、50、80、100 的场景开展了 5 组仿真实验。每种算法均独立运行 20 次，以确保实验结果的稳定性和可靠性。通过对不同场景下的实验结果进行比较，可以全面评估 IFOPSO 算法在取货点数量变化时的适应性与优化效果。

表 4 展示了不同数量取货点下 4 种算法的实验结果，图 5 显示了不同取货点数量下的收敛曲线。由表 4 可见，在 5 种实验场景中，IFOPSO 算法的总代价平均值均优于其他 3 种 PSO 算法。在 30 个取货点的场景下，IFOPSO 算法与 PSO 算法在取货及时率方面均寻得相同的最优解；在 50 个取货点的场景下，PSO 算法在无人车超载率方面取得了最优解，

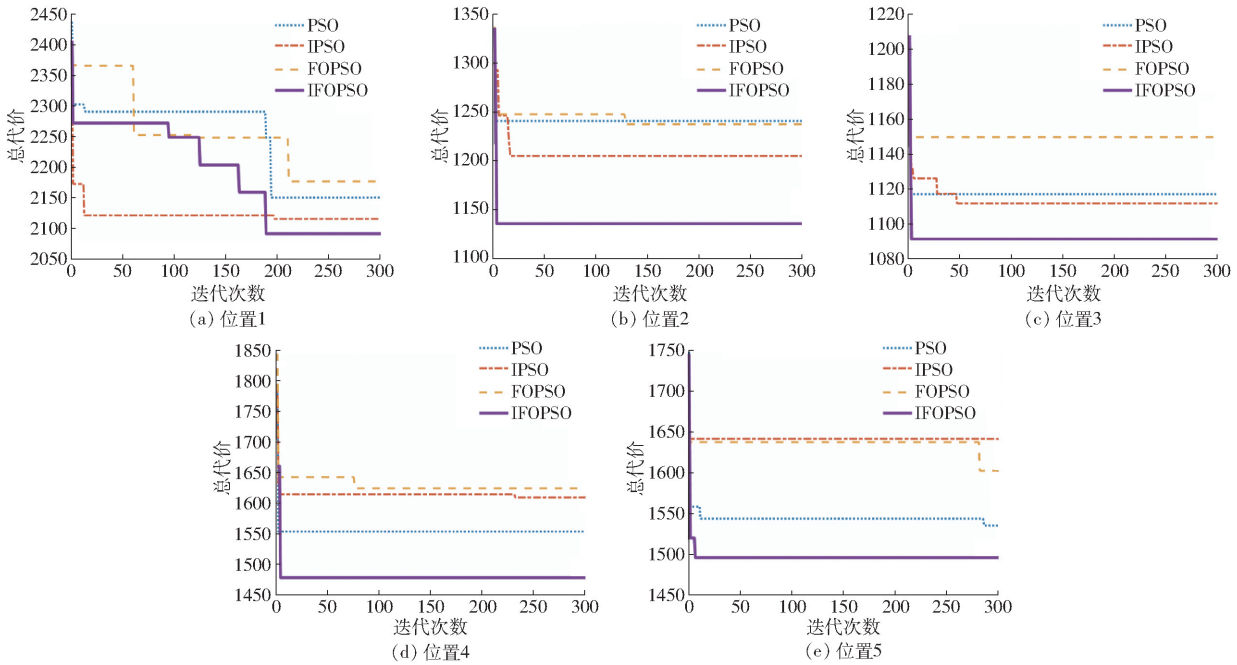

图 4 不同相对位置下总代价收敛曲线

Fig. 4 Convergence curves of total cost at different relative positions

表 4 不同数量取货点下实验结果对比

Tab. 4 Comparison of experimental results under different quantity pick-up points

取货点数量	评价标准	指标	PSO	IPSO	FOPSO	IFOPSO
20	总代价	平均值	1 381. 61	1 371. 936	1 412. 848 3	1 367. 190 1
		标准差	0	0	26. 448 0	28. 393 2
	及时率	平均值/%	95	95	88. 75	100
		标准差	0	0	0. 223 3	0
	超载率	平均值/%	5	5	3. 75	3
		标准差	0	0	0. 031 9	0. 025 1
30	总代价	平均值	2 132. 169 2	2 012. 499 8	2 082. 014 1	2 012. 155 1
		标准差	0	0	34. 946 9	32. 038 7
	及时率	平均值/%	87	87	88. 50	93. 33
		标准差	0	0	0. 042 6	0. 023 7
	超载率	平均值/%	3	7	5	3
		标准差	0	0	0. 020 3	0. 024
50	总代价	平均值	3 072. 802 5	2 983. 798	3 120. 447 5	2 849. 733 6
		标准差	0	0	53. 901 9	48. 181 2
	及时率	平均值/%	88	90	92. 90	93. 30
		标准差	0	0	0. 038 1	0. 022 7
	超载率	平均值/%	5	8	5. 50	5. 70
		标准差	0	0	0. 022 8	0. 033 3
80	总代价	平均值	5 621. 138	5 201. 533 6	5 729. 689 7	5 087. 802 9
		标准差	0	0	62. 608 8	43. 101 1
	及时率	平均值/%	90	93. 75	88	92. 50
		标准差	0	0	0. 039 4	0. 021 8
	超载率	平均值/%	5	5	5	4. 25
		标准差	0	0	0. 013 5	0. 013 7
100	总代价	平均值	6 755. 965 7	6 695. 314 5	6 992. 971 4	6 202. 830 9
		标准差	0	0	102. 737 6	68. 605 7
	及时率	平均值/%	76	92	85. 35	93. 20
		标准差	0	0	0. 057 6	0. 021 8
	超载率	平均值/%	5	5	5. 05	4. 20
		标准差	0	0	0. 012	0. 012

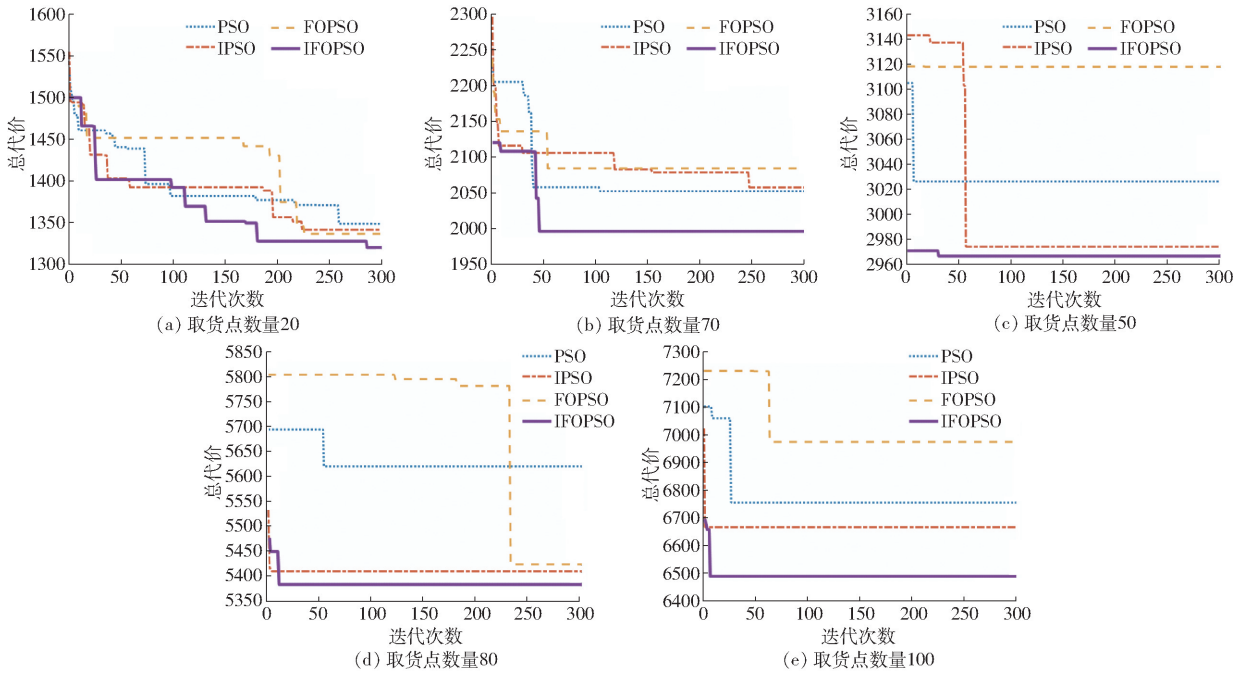

图 5　不同数量取货点下总代价收敛曲线

Fig. 5　Convergence curves of total cost under different numbers of picking points

但 IFOPSO 算法的结果与其非常接近。仿真实验结果表明，随着取货点数量的增加，IFOPSO 算法依然能够保持优异的性能，在复杂场景中找到更优的取货方案，展示了其良好的适应性和优化能力。

4　结束语

研究了农产品运输场景下产地与销地配送环节中的多无人车协同任务分配问题。为更贴合实际取送货任务的需求，在考虑前往农户取货的时间约束基础上，构建了一个从农贸市场出发的取送货任务模型，并提出了一种改进分数阶粒子群算法（IFOPSO）对该模型进行求解。该算法在迭代过程中引入了分数阶列维随机步长，增加了粒子搜索的多样性。为进一步提升算法的寻优能力，加入了列维阶次自适应调整机制，列维阶次随迭代次数线性调整，以优化搜索过程。本文采用 10 个基准测试函数对算法进行寻优实验，并计算了评价指标的平均值和标准差，将 IFOPSO 算法与 PSO、GA、CS、GWO 等标准群优化算法，以及 IPSO、FOPSO 等改进粒子群算法进行对比。结果表明，IFOPSO 算法的寻优性能明显提升，具有较强的鲁棒性和优异的全局搜索能力。后通过多组仿真实验验证，IFOPSO 算法能够有效解决多无人车协同任务分配问题，尤其在大规模取送货场景中展现了良好的搜索性能，能够找到更优的取送货方案。相比于其他标准和改进的粒子群算法，IFOPSO 算法在降低总代价、提高任务响应效率和减少超载率方面表现更加优越。

参　考　文　献

[1] 王峰，付青坡，韩孟臣，等. LeCMPSO 算法求解异构无人机协同多任务重分配问题[J]. 控制理论与应用，2024，41(6)：1009－1017.
WANG Feng，FU Qingpo，HAN Mengchen，et al. LeCMPSO algorithm for cooperative multi-task reassignment of heterogeneous UAVs[J]. Control Theory and Applications，2024，41(6)：1009－1017.（in Chinese）

[2] HUANG J，LI B，SONG C，et al. Multi UAV cooperative reconnaissance based on dynamic programming IDQN algorithm[C]// International Conference on Cyber-Physical Social Intelligence，2023.

[3] 杜永浩，邢立宁，蔡昭权. 无人飞行器集群智能调度技术综述[J]. 自动化学报，2020，46(2)：222－241.
DU Yonghao，XING Lining，CAI Zhaoquan. Unmanned aerial vehicles cluster intelligent scheduling technology review[J]. Journal of Automation，2020，46(2)：222－241.（in Chinese）

[4] 王猛，赵博，刘阳春，等. 基于多变异分组遗传算法的多机协同作业静态任务分配[J]. 农业机械学报，2021，52(7)：19－28.
WANG Meng，ZHAO Bo，LIU Yangchun，et al. Static task tllocation for multi-machine cooperative work based on multi-variant grouping genetic algorithm[J]. Transactions of the Chinese Society for Agricultural Machinery，2021，52(7)：19－28.（in Chinese）

[5] GUO H，WANG Y，LIU J，et al. Multi-UAV cooperative task offloading and resource allocation in 5G advanced and beyond

[J]. IEEE Transactions on Wireless Communications, 2024, 23(1): 347 - 359.
[6] CHEN Z, HOU S, WANG Z, et al. Delivery route scheduling of heterogeneous robotic system with customers satisfaction by using multi-objective artificial bee colony algorithm[J]. Drones, 2024, 8: 519.
[7] 宫金良,王伟,张彦斐,等. 基于动态刺激响应模型的异质农业 Agent 群任务分配策略[J]. 农业机械学报,2021,52(5): 142 - 150.
GONG Jinliang, WANG Wei, ZHANG Yanfei, et al. Task assignment strategy of heterogeneous agricultural Agent groups based on dynamic stimulus response model[J]. Transactions of the Chinese Society for Agricultural Machinery, 2021,52(5):142 - 150. (in Chinese)
[8] 曹如月,李世超,季宇寒,等. 基于蚁群算法的多机协同作业任务规划[J]. 农业机械学报,2019,50(增刊 1):34 - 39.
CAO Ruyue, LI Shichao, JI Yuhan, et al. Task planning for multi-machine cooperative work based on ant colony algorithm[J]. Transactions of the Chinese Society for Agricultural Machinery,2019,50(Supp. 1):34 - 39. (in Chinese)
[9] 张祥银, 夏爽, 张天. 基于自适应遗传学习粒子群算法的多无人机协同任务分配[J]. 控制与决策, 2023, 38(11):3103 - 3111.
ZHANG Xiangyin, XIA Shuang, ZHANG Tian. Multi-UAV collaborative task assignment based on adaptive genetic learning particle swarm optimization[J]. Control and Decision, 2023, 38(11):3103 - 3111. (in Chinese)
[10] XU B, FANG H, MAO Y, et al. A coupling algorithm for task and path planning of multi-UGVs under environmental inspiration[C]//IEEE International Conference on Control and Automation, 2024.
[11] LI Y, ZHANG Z, HE Z, et al. A heuristic task allocation method based on overlapping coalition formation game for heterogeneous UAVs[J]. IEEE Internet of Things Journal, 2024, 11(17): 28945 - 28959.
[12] WANG S, LIU Y, QIU Y, et al. Cooperative task allocation for multiple UAVs based on min-max ant colony system[C]// Asian Conference on Artificial Intelligence Technology, 2021.
[13] WANG W, RU L, LU B, et al. Multi-task cooperative assignment of two-stage heterogeneous multi-UAV based on improved CBBA[C]//International Symposium on Computer Technology and Information Science, 2023.
[14] ZANG J, CHEN Y, YANG Q, et al. Dynamic task allocation of multiple UAVs based on improved A - QCDPSO[J]. Electronics, 2022, 11: 1028.
[15] GAO Y, ZHANG Y, ZHU S, et al. Multi-UAV task allocation based on improved algorithm of multi-objective particle swarm optimization[C]//International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 2018.
[16] 万忠海, 叶生进, 郑姣. 线性微分递减自适应粒子群算法在水电站优化调度中的应用[J]. 水力发电, 2017, 43(9): 85 - 88.
WAN Zhonghai, YE Shengjin, ZHENG Jiao. Application of linear differential decline adaptive particle swarm optimization in hydropower station scheduling[J]. Hydropower, 2017, 43(9): 85 - 88. (in Chinese)
[17] 王学达. 基于离散粒子群算法的多车协同任务分配研究[D]. 北京:北京交通大学,2023.
WANG Xueda. Research on multi-vehicle collaborative task assignment based on discrete particle swarm optimization[D]. Beijing: Beijing Jiaotong University,2023. (in Chinese)
[18] 崔东文, 金波. 动态自适应粒子群优化算法与最小二乘支持向量机在年径流预测中的应用[J]. 人民珠江, 2016, 37(10): 27 - 33.
CUI Dongwen, JIN Bo. Application of dynamic adaptive particle swarm optimization algorithm and least square support vector machine in annual runoff prediction[J]. People's Pearl River, 2016, 37(10): 27 - 33. (in Chinese)
[19] 陈玉全. 分数阶梯度下降法基础理论研究[D]. 合肥:中国科学技术大学, 2020.
CHEN Yuquan. Research on the basic theory of fractional step-down method[D]. Hefei: University of Science and Technology of China, 2020. (in Chinese)
[20] 刘惊. 基于改进粒子群算法和支持向量结合的辅助发声训练系统研究[J]. 自动化与仪器仪表, 2024(4): 176 - 179, 184.
LIU Jing. Research on assisted voice training system based on improved particle swarm optimization algorithm and support vector[J]. Automation & Instrumentation, 2024(4): 176 - 179, 184. (in Chinese)
[21] 刘长石,李君宇,易鲲翔,等. 分时电价背景下电动车配送-充电/放电路径规划[J]. 控制与决策,2025,40(3):794 - 802.
LIU Changshi, LI Junyu, YI Kunxiang, et al. Electric vehicle distribution-charge/discharge path planning under the background of TOU[J]. Control and Decision, 2025,40(3):794 - 802. (in Chinese)
[22] 赵伟,侯保林. 电磁炮纤维缠绕约束结构多目标优化[J]. 兵工学报,2024,45(11):3820 - 3832.
ZHAO Wei, HOU Baolin. Multi-objective optimization of fiber winding restraint structure of electromagnetic gun[J]. Acta Ordnance Engineering, 2024,45(11):3820 - 3832. (in Chinese)
[23] 何爽, 胡丹, 王骏骏, 等. 基于改进灰狼算法的海岛微电网优化调度[J]. 电气自动化, 2024, 46(3): 41 - 44, 48.
HE Shuang, HU Dan, WANG Junjun, et al. Optimal scheduling of island microgrid based on improved gray wolf algorithm [J]. Electrical Automation, 2024, 46(3): 41 - 44, 48. (in Chinese)