

DOI:10.19651/j.cnki.emt.2417499

融合多策略改进的班翠鸟算法及微电网调度^{*}何 印^{1,2} 孔玲玲^{1,2} 郑哲明^{1,2}

(1. 云南民族大学电气信息工程学院 昆明 650050; 2. 云南省无人自主系统重点实验室 昆明 650500)

摘 要: 为解决班翠鸟优化算法(PKO)收敛精度低和易陷入局部最优等问题,本文提出了一种多策略改进的班翠鸟优化算法(IPKO)。首先,采用拉丁超立方抽样避免在高维问题中出现过度集中或忽视潜在有利区域的情况,降低局部最优的风险;其次,引入鱼鹰算法(OOA)中的定位捕鱼机制,增强最优区域的探索和逃逸局部最优的能力;最后,融合新的坠落机制提高搜索稳定性,避免过早收敛,同时通过自适应变异率终止条件,自适应地平衡全局探索与局部开发,从而优化解的质量和搜索效率。比较了在不同特征维度下的训练-测试准确率、运行时间等,并分析了种群大小和迭代次数对算法性能的影响;通过在12个基准测试函数上进行实验对比,实验结果表明,IPKO在收敛速度、求解精度、稳定性以及Friedman检验方面均优于其他对比算法。将IPKO应用于微电网调度问题中,证明了此算法较其他算法成本更低,较原算法PKO降低了1.92%,验证了实际问题中的有效性与可靠性。

关键词: 班翠鸟优化算法;拉丁超立方抽样;鱼鹰定位捕鱼机制;坠落机制;优化调度

中图分类号: TN819.1 **文献标识码:** A **国家标准学科分类代码:** 470.4054

A multi-strategy improved pelican optimization algorithm
for microgrid schedulingHe Yin^{1,2} Kong Lingling^{1,2} Zheng Zheming^{1,2}

(1. School of Electrical and Information Technology, Yunnan Minzu University, Kunming 650050, China;

2. Key Laboratory of Unmanned Autonomous Systems of Yunnan Province, Kunming 650500, China)

Abstract: To address the issues of low convergence accuracy and susceptibility to local optima in the PKO algorithm, this paper proposes a multi-strategy improved IPKO algorithm. First, Latin hypercube sampling is used to avoid over-concentration or neglect of potentially beneficial areas in high-dimensional problems, thus reducing the risk of local optima. Secondly, the positioning fishing mechanism from the OOA algorithm is introduced to enhance exploration of the optimal region and improve the ability to escape from local optima. Finally, a new falling mechanism is integrated to improve search stability and prevent premature convergence. An adaptive mutation rate termination condition is also applied to dynamically balance global exploration and local exploitation, optimizing solution quality and search efficiency. The training-testing accuracy and runtime under different feature dimensions are compared, and the impact of population size and iteration count on the algorithm's performance is analyzed. Experimental results on 12 benchmark test functions show that IPKO outperforms other comparison algorithms in terms of convergence speed, solution accuracy, stability, and the Friedman test. When applied to the microgrid scheduling problem, IPKO demonstrates lower costs compared to other algorithms, with a reduction of 1.92% over the original PKO, confirming its effectiveness and reliability in practical applications.

Keywords: pied kingfisher optimizer; Latin hypercube sampling; osprey positioning fishing mechanism; falling mechanism; optimization scheduling

0 引 言

班翠鸟优化算法(pied kingfisher optimizer, PKO)是

Bouaouda等^[1]于2024年提出的一种群智能优化算法。该算法凭借新颖性和高效性在优化领域引起广泛关注。与粒子群优化算法、哈里斯鹰优化算法(harris hawks optimization,

收稿日期:2024-12-01

* 基金项目:国家自然科学基金(52061042)项目资助

HHO)^[2]、鲸鱼优化算法 (whale optimization algorithm, WOA)^[3]、正弦余弦算法 (sine cosine algorithm, SCA)^[4]、灰狼优化算法 (grey wolf optimization, GWO)^[5]、飞蛾火焰优化算法 (moth-flame optimization, MFO)^[6] 等经典算法相比, PKO 算法处理复杂优化问题方面展现出了显著优势,

在微电网优化调度算法中, 优化变量较多, 从而微电网的优化问题大多是非线性多目标优化问题^[7-8], 张林等^[9]采用改进的蝙蝠算法进行求解, 尽管改进后的蝙蝠算法在最终解的质量上优于传统蝙蝠算法, 邢毓华等^[10]提出了改进的多目标粒子群算法, 通过引入变异策略和自适应网格法, 提升了全局搜索能力, 增强了算法多样性与解的质量, 使其在 Pareto 前沿的均匀分布与收敛速度方面表现更优, 蒋翱徽等^[11]在蜣螂优化算法中引入混沌映射初始化种群位置, 结合可变螺旋策略和 Levy 飞行策略, 显著提升了算法的全局搜索能力与收敛速度, 有效避免了局部最优解。王雨虹等^[12]将混沌映射用于麻雀算法的种群初始化, 并提出均匀分布自适应权重和动态拉普拉斯算法, 增强了算法摆脱局部最优的能力, 优化了全局收敛性。智能优化算法的不断提出和改进, 有效解决了现实中的复杂问题, 如: 障碍路径规划^[13]、光伏模型参数辨识和工程设计^[14]等。

尽管 PKO 算法在解决工程问题上取得了显著成效, 但其全局搜索能力有限, 易陷入局部最优。为提升优化性能, 本文提出了多策略改进的 IPKO 算法。首先, 采用拉丁超立方抽样初始化^[15] (Latin hypercube sampling, LHS), 避免高维问题中出现过度集中或忽视有利区域, 从而减少局部最优的风险; 引入鱼鹰算法^[16] (osprey optimization algorithm, OOA) 中的定位捕鱼机制, 提高探索最优区域和逃离局部最优的能力; 融合坠落机制^[17] 和变异率终止条件, 增强搜索稳定性和优化效率。通过对 12 个标准测试函数的对比分析, IPKO 在收敛速度、精度和全局最优解的达到上均表现更优。最后, 在微电网优化调度问题中的应用验证该算法的优势。

1 原班翠鸟优化算法

1.1 初始化阶段

与许多其他基于种群方式类似, PKO 通过在搜索空间随机生成一组起始解来作为第一次尝试来启动搜索过程:

$$X_{i,j} = LB + (UB - LB) \times rand \quad (1)$$

$$i = 1, 2, \dots, N \text{ and } j = 1, 2, \dots, Dim \quad (2)$$

式中: $X_{i,j}$ 表示第 i 个个体在第 j 维的位置, $rand \in [0, 1]$ 的随机数, UB 和 UL 分别表示搜索空间的上限和下限。适应度函数为每个个体分配一个适应度值。根据这些值, 挑选出表现最优的个体, 并使用它们来生成下一代。

1.2 探索阶段

PKO 的探索阶段受到班翠鸟的栖息和盘旋行为的启发。班翠鸟的位置根据以下公式进行更新迭代:

$$X_i(t+1) = X_i(t) + \alpha \times T \times [X_j(t) - X_i(t)] \quad (3)$$

$$\alpha = 2 \times randn(1, dim) \quad (4)$$

式中: $X_i(t+1)$ 、 $X_i(t)$ 分别为下一次迭代的位置和当前迭代位置, $randn$ 为正态分布中的随机数, $i, j \in [1, N]$ 的正整数且 $i \neq j$, N 为总体的规模, dim 为研究问题的维度。参数 T 在 PKO 中至关重要, 确保在不同模式下的最佳性能, 它是根据当前策略动态而确定, 有栖息和悬停两种状态。

1) 在栖息策略中, T 值随着迭代次数的增加逐渐减小。初期 T 较大, 允许算法进行较广泛的搜索, 这时会有较强的探索性。随着迭代进行, T 逐渐减小, 算法更多进行局部优化, 栖息策略参数 T 数学模型:

$$T = \left\{ \exp(1) - \exp \left[\left(\frac{t-1}{Max_Iter} \right)^{\frac{1}{BF}} \right] \right\} \times \cos(Crest_angles) \quad (5)$$

$$Crest_angles = 2 \times \pi \times rand \quad (6)$$

式中: Max_Iter 为最大迭代次数; 跳动因子 $BF = 8$ 。

2) 在悬停策略中, T 与适应度函数值相关, 适应度值较好的个体 T 值较小, 有助于加速局部搜索。随着算法推进, T 值逐渐减小, 迫使个体进行更精细的搜索, 以寻找最优解, 悬停策略参数 T 数学模型:

$$T = rand \left(\frac{PKO_Fitness(j)}{PKO_Fitness(i)} \right) \times \left(\frac{t^{\frac{1}{BF}}}{Max_Iter^{\frac{1}{BF}}} \right) \quad (7)$$

式中: $PKO_Fitness(i)$ 和 $PKO_Fitness(j)$ 分别为第 i 和第 j 只班翠鸟的适应度值。

1.3 开发阶段

班翠鸟凭借快速精准的潜水、锋利的喙和敏锐的视力, 成为出色的水生捕食者。其数学模型如下:

$$X_i(t+1) = X_i(t) + HA \times o \times \alpha \times (b - X_{best}(t)) \quad (8)$$

$$\begin{cases} HA = rand \times \left(\frac{PKO_Fitness(i)}{Best_Fitness} \right) \\ o = \exp \left(\frac{-t}{Max_Iter} \right)^2 \\ \alpha = 2 \times randn(1, dim) - 1 \\ b = X_i(t) + o^2 \times randn \times X_{best}(t) \end{cases} \quad (9)$$

式中: $Best_Fitness$ 为所有迭代后的最佳适应度值。 $i \in [1, N]$ 的正整数, α 为控制参数, o 为狩猎能力。

1.4 共生阶段

班翠鸟与多种种类有共生关系。其数学模型如下:

$$X_i(t+1) = \begin{cases} X_m(t) + o \times \alpha \times abs & rand > (1 - PE) \\ (X_i(t) - X_n(t)), & \\ X_i(t), & \text{其他} \end{cases} \quad (10)$$

$$\begin{cases} o = \exp \left(\frac{-t}{Max_Iter} \right)^2 \\ \alpha = 2 \times randn(1, dim) - 1 \\ PE = PE_{\max} - (PE_{\max} - PE_{\min}) \times \left(\frac{t}{Max_Iter} \right) \end{cases} \quad (11)$$

式中: X_m 、 X_n 为种群中随机抽取 2 个个体位置。 PE 为斑翠鸟的捕食效率, $PE_{\max}$ 、 $PE_{\min}$ 分别为 0.5 和 0。

2 多策略改进的班翠鸟优化算法

本文提出多策略改进的原班翠鸟优化算法(IPKO), 具体地, PKO 采用随机函数初始化, 但在高维复杂情况下存在局限, 导致算法在某些过度集中或忽略某些潜在的有利区域, 从而影响整体的优化效率。为解决上述问题, IPKO 算法初始化阶段采用 LHS。由于 PKO 算法的局部和全局搜索能力不足, 后期易陷入局部最优, 因此引入鱼鹰优化算(OOA)中的定位捕鱼机制, 增强局部和全局搜索能力。在原 PKO 算法中共生策略效果有限, 本文引入坠落机制, 模拟班翠鸟在飞行和觅食过程中坠落或被捕食的现象, 通过坠落步长和位置建立坠落机制模型, 以保持种群数量不变。

2.1 拉丁超立方抽样

种群初始化影响算法的收敛速度和精度。班翠鸟优化算法采用随机函数初始化, 但在高维复杂情况下存在局限, 导致算法在某些过度集中或忽略某些潜在的有利区域, 从而影响整体的优化效率。

为解决上述问题, 算法初始化阶段采用 LHS 抽样, 可在每个维度上均匀分布样本点。LHS 的均匀性确保了样本覆盖整个参数空间, 增强了探索性能和种群多样性, 避免算法陷入局部最优, 从而为算法提供了更广泛的搜索起始点。总的来说, LHS 是一种有效的多维空间采样技术, 可以提高优化算法的性能和效率。LHS 的具体操作步骤如下:

- 1) 确定样本数 N ;
- 2) 将单位区间 $[0, 1]$ 划分为 n 等份, 均匀覆盖搜索空间;
- 3) 在每个子区间 $\left[\frac{i}{n}, \frac{i+1}{n}\right]$ 内随机抽取样本点;
- 4) 对各维度的样本位置随机置换, 确保均匀分布且不重复。

如图 1 所示, 通过对比实验验证 LHS 的有效性, 结果显示 LHS 初始化相比 PKO 算法的随机初始化能产生更均匀的种群分布, 直方图直观展示了两者的区别, 表明 LHS 能在搜索空间内实现更均匀的数量分布。

2.2 融入鱼鹰定位捕鱼机制

原始 PKO 算法通过随机因子和适应度比较在“探索”和“开发”阶段更新粒子位置, 但易导致开发阶段过早收敛, 限制全局搜索能力。因此引入 OOA 中的定位捕鱼机制, 通过适应度比较选择目标, 增强局部和全局搜索能力。算法中个体(鱼鹰)的定位和搜索机制的公式为:

$$FP_i = \{X_k \mid k \in \{1, 2, \dots, N\} \Delta F_k < F_i\} \cup \{X_{best}\} \quad (12)$$

式中: FP 为第 i 条鱼鹰鱼群位置的集合; X_{best} 为最优解。

鱼鹰随机发现目标并进攻, 通过式(13)、(14)迭代模拟

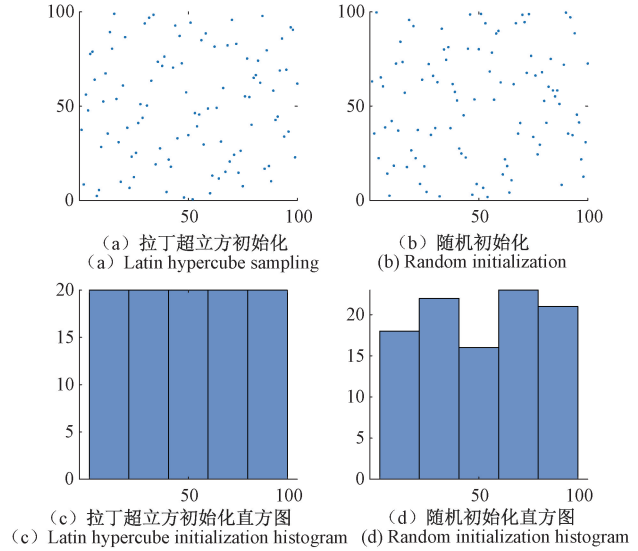


图 1 LHS 与随机初始化对比

Fig. 1 Comparison between LHS and random initialization

鱼鹰向目标移动。如果新位置提高了目标函数值, 则根据式(15)替换鱼鹰的原始位置。

$$x_{i,j}^{p1} = x_{i,j} + rand \times (SF_{i,j} - I_{i,j} \times x_{i,j}) \quad (13)$$

$$x_{i,j}^{p1} = \begin{cases} x_{i,j}^{p1}, & lb_j \leq x_{i,j}^{p1} \leq ub_j \\ lb_j, & x_{i,j}^{p1} < lb_j \\ ub_j, & x_{i,j}^{p1} > ub_j \end{cases} \quad (14)$$

$$X_i = \begin{cases} X_i^{p1}, & X_i^{p1} < F_i \\ X_i, & \text{其他} \end{cases} \quad (15)$$

式中: $x_{i,j}^{p1}$ 是基于鱼鹰算法的第一阶段的第 i 只鱼鹰的新位置, $x_{i,j}^{p1}$ 表示第 j 维, F_i^{p1} 表示目标函数值, $SF_{i,j}$ 表示为在 j 维第 i 只鱼鹰选择的鱼, $I_{i,j} \in \{1, 2\}$ 的随机数。

2.3 坠落机制

PKO 算法使用固定公式确定步长, 缺乏动态适应性, 容易陷入早熟收敛。本文引入坠落机制, 根据迭代次数动态缩小步长, 平衡了全局探索与局部开发能力, 模拟班翠鸟在飞行和觅食过程中坠落或被捕食的现象, 通过坠落步长和位置建立坠落机制模型, 以保持种群数量不变。模型为:

$$X_{i,j}^{t+1} = r_1 X_{i,j}^t - r_2 X_{i,j}^t + r_3 X_{step} \quad (16)$$

式中: r_1 、 r_2 和 r_3 为区间 $(0, 1)$ 的随机数, X_{step} 表示为班翠鸟坠落步长, 其模型为式(17)。

$$\begin{cases} X_{step} = (U - L) \exp(-C_2 t / T_{max}) \\ W_f = 0.1 - 0.05 t / T_{max} \\ C_2 = 2W_f \times n \end{cases} \quad (17)$$

式中: U 和 L 表示第 i 维度上限和下限的取值, n 表示问题的维度, C_2 为步长因子, 步长因子受迭代次数影响, 班翠鸟坠落概率 W_f 从 0.1 逐渐减少到 0.05, 说明算法在逼近最优解时搜索风险逐渐降低。

2.4 变异率终止条件

1)变异率动态变化

变异率 μ 是影响算法探索性的关键参数,表示算法在搜索空间中随机变动的程度。其值通常随算法运行逐渐减小,以模拟从探索阶段到开发阶段的过渡。变异率模型如下:

$$\mu(t) = \max(0.01, 1 - (t/T_{\max})(1 - \theta)) \quad (18)$$

式中: t 为当前迭代次数, $T_{\max}$ 为最大迭代次数, θ 为预设变异率的最小阈值, $t/T_{\max}$ 为当前迭代所占比例。

2)终止条件

变异率终止条件通过判断当前变异率是否低于设定的最小阈值来决定是否提前终止算法。当 μ 小于 θ 阈值时,算法提前停止。

$$\mu(t) < \theta \quad (19)$$

变异率调节探索与开发的平衡。较高的变异率增强探索性,有助于广泛搜索解空间;较低的变异率增强开发性,倾向于局部优化。因此,随着迭代进程,变异率逐渐减小,促进初期快速探索,后期专注于精细优化。

2.5 算法流程

步骤 1)设置参数, $BF=8$ 、式 (7);初始化种群,使用 LHS 生成式 (1);

步骤 2)适应度计算, $PKO_Fitness(i)$ 、 $PKO_Fitness(j)$;

步骤 3)排序适应度值;选择最优个体使用式 (7);

步骤 4)迭代过程,设定迭代次数 $t=1$,直到达到 Max_Iter ;

更新变异率 μ :使用式 (18);

探索阶段: $if\ rand < 0.8$, 选择一个随机个体, 计算式 (9)的第一个式子,使用式 (8)更新位置;

开发阶段: $if\ rand \geq 0.8$,使用式 (9)的第 3 个式子更新位置;

使用式 (14)边界处理;

适应度值更新;

使用式 (15)将位置更新;

更新全局最优解;

步骤 5)更新选择概率用式 (11)的第 3 个式子,使用式 (13)更新位置;

步骤 6)计算更新步长,使用式 (17)第 1 个式子,使用式 (16)更新位置;

步骤 7) $if\ \mu(t) < \theta$, 输出最优解。

3 仿真实验及结果分析

3.1 仿真实验准备

实验环境: 64 位 Windows 11 操作系统,处理器为 AMD Ryzen 9 7945HX with Radeon Graphics 2.50 GHz,运行内存 6GB,编程环境为 MATLAB R2023a 进行实验仿真。种群数量设为 30,最大迭代次数为 1 000 次,通过计算最优值 (Best)、标准差 (Std)、平均值 (Average)、中值

(Median)、差值 (Diff)以及 Friedman 检验作为指标来验证本文提出的 IPKO 算法的有效性及其优越性。

3.2 泛化能力测试仿真

首先提高训练和测试准确率及其差距评估算法对已知数据的拟合能力与对未知数据的预测能力,较小的训练、测试差距表明更优的泛化性能。其次记录不同特征维度下运行时间,分析算法计算效率及其扩展性,运行时间较短表明算法更高效。最后考察特征维度对准确率、运行时间及内存消耗的影响,评估算法在高维数据下的适应能力。由表 1 和 2 可得 IPKO 在测试集上的准确率 (0.47~0.53)整体高于 PKO (0.40~0.505),且训练与测试准确率更接近,尤其在 1 000 维度时表现最佳(测试准确率 0.53)。相比之下,PKO 的训练准确率较高 (约 0.52),但测试准确率波动较大,高维时表现下降 (0.485),显示一定过拟合现象。在运行时间上,IPKO 在所有维度均优于 PKO,尤其在 1000 维度时效率更高。如图 2(a)所示,IPKO 在不同维度下运行时间均比 PKO 短。如图 2(b)所示,两种算法的内存消耗差异不显著。如图 3 所示,IPKO 相比于 PKO 在高维情况下表现出更优的泛化能力。综上,IPKO 适合高维特征及泛化能力要求较高的场景,而 PKO 更适合训练拟合要求较高的任务。

表 1 IPKO 不同维度下仿真数据表

Table 1 Simulation data of IPKO for different dimensions

维度	IPKO 测试准确率	IPKO 训练准确率	时间/ms
500	0.47	0.466 25	0.211 7
1 000	0.53	0.507 50	0.170 0
2 000	0.46	0.530 00	0.171 4

表 2 PKO 不同维度下仿真数据表

Table 2 Simulation data of PKO for different dimensions

维度	PKO 测试准确率	PKO 训练准确率	时间/ms
500	0.400	0.521 25	0.273 1
1 000	0.505	0.528 75	0.175 4
2 000	0.485	0.516 25	0.178 2

3.3 最优适应度的影响分析

优化算法参数(种群大小和最大迭代次数)、优化问题参数(目标函数、维度及搜索空间边界),以及实验设计(参数组合与单参数敏感性分析)。种群大小设置为 [50,200,500,1 000],最大迭代次数为 [500,1 000,2 000],以评估不同参数组合对算法性能的影响; Griewank 函数的上下界设置为 $LB=-600$; $UB=600$,实验包括 12 种参数组合和单参数变化(固定最大迭代次数或种群大小),分析在系统研究参数配置对算法收敛性和最优适应度的影响。

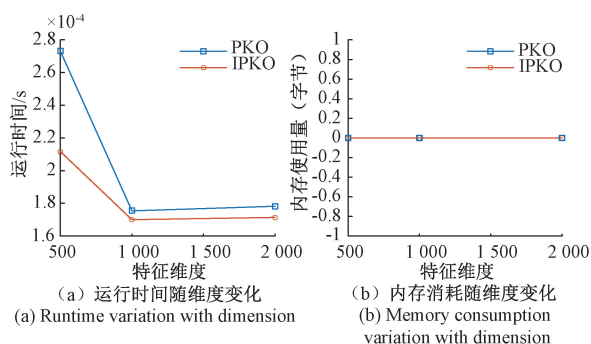


图 2 高维空间下算法性能检验

Fig. 2 Performance evaluation of algorithms in high-dimensional space

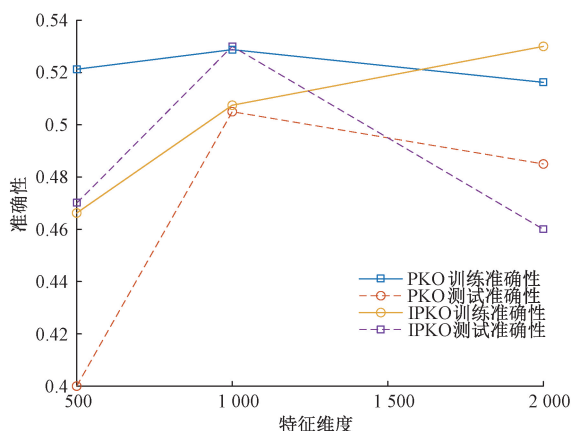


图 3 训练与测试准确性对比

Fig. 3 Training and testing accuracy comparison

由图 4 可知,种群大小和最大迭代次数的变化对算法的最优适应度影响较小,说明算法对这两个参数的敏感性较低,表现出较强的鲁棒性和稳定性。由图 5 可知,该三维散点图分析了种群大小和最大迭代次数对算法性能的影响。结果表明,这两个参数对适应度值的影响较小,适应度在参数组合变化时表现出较强的稳定性,也能说明算法对这两个参数的敏感性较低。

3.4 收敛性分析

在 12 种测试函数的基础上,进一步验证了 IPKO 算法相较于其他对比算法在收敛性能方面的优势。图 6 展示了 5 种算法的收敛曲线,以直观对比其性能差异。

通过对单峰函数(F1-F4)、多峰函数(F5-F8)及复合函数(F9-F12)的收敛曲线和收敛速度分析,IPKO 算法在所有测试函数中均表现出卓越的收敛性能。单峰函数测试中,IPKO 算法展现出快速收敛能力,能够在早期迭代中迅速达到最低适应度值,显著优于其他算法。在多峰函数测试中,IPKO 算法表现出较强的跳出局部最优能力,尽管部分对比算法(如 PSA)在后期迭代中显示出一定的改进潜力。复合函数测试进一步验证了 IPKO 算法的鲁棒性和适应复杂优化问题的能力,其收敛速度和最终适应度值始终

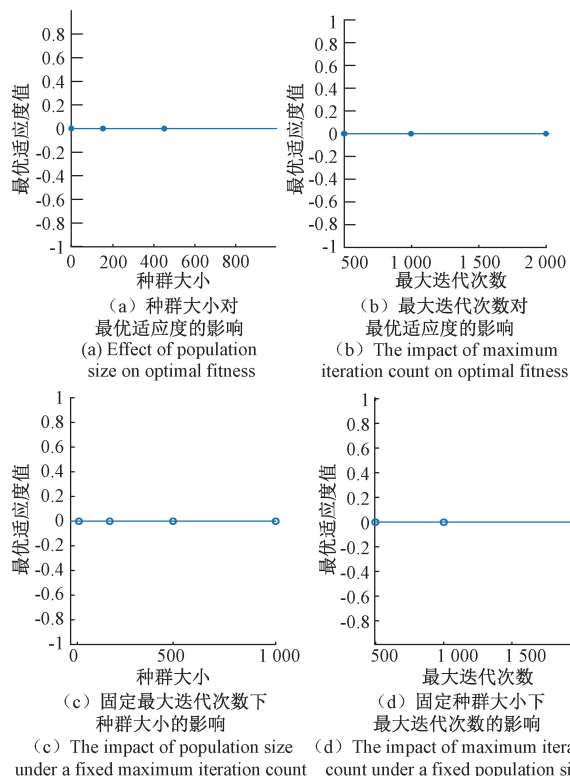


图 4 种群大小与最大迭代次数对算法最优适应度的影响分析

Fig. 4 Analysis of the impact of population size and maximum iterations on the optimal fitness value of the algorithm

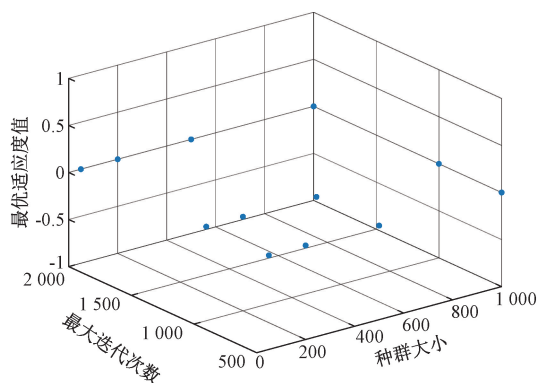


图 5 参数组合对 IPKO 性能的影响

Fig. 5 Impact of parameter combinations on IPKO performance

优于其他算法。综上,IPKO 算法在收敛速度、全局优化能力及求解精度方面均表现出显著优势,适用于多种类型的优化问题。

为验证所提 IPKO 的有效性,采用表 3 的 12 个标准测试函数来测试算法的性能,将多策略改进的 IPKO 与近几年出现的其他群智能算法进行对比分析,对比的算法包括原算法 PKO、雪消融优化算法(snow ablation optimizer, SAO)^[18]、鸽群优化算法(pigeon-inspired optimization, PIO)^[19]、PSA 搜索算法^[20]、鹦鹉优化算法^[21](parrot optimizer, PO)。

表 3 标准测试函数详细信息

Table 3 Detailed information of standard test functions

函数类型	函数代号	函数名称	定义域及维度	最优值
单峰函数	F1	Schwefel	$[-10, 10]^{30}$	0
	F2	Rosenbrock	$[-30, 30]^{30}$	0
	F3	Step	$[-100, 100]^{30}$	0
	F4	Quartic	$[-1.28, 1.28]^{30}$	0
多峰函数	F5	Griewank	$[-500, 500]^{30}$	$-418.98 \times dim$
	F6	Ackley	$[-32, 32]^{30}$	0
	F7	Schwefel 2.13	$[-50, 50]^{30}$	0
	F8	Schwefel 2.20	$[-50, 50]^{30}$	0
复合函数	F9	Schwefel 2.22	$[-5, 5]^4$	0.148 4
	F10	Pathological	$[0, 1]^3$	-3
	F11	Zakharov	$[0, 10]^4$	-1
	F12	Penholder	$[0, 10]^4$	-1

3.5 IPKO 算法的全面性能评估及统计显著性验证

在单峰函数(F1-F4)、多峰函数(F5-F8)及复合型函数(F9-F12)的测试中,如表 4 所示,IPKO 算法在收敛精度、稳定性和全局优化能力方面表现出显著优势。单峰函数测试中,IPKO 的 Best 值接近理论最优值,且 Std 值几乎为 0,展现出卓越的全局搜索能力和极高的收敛稳定性。在多峰函数中,IPKO 同样在 Best 值和 Std 值上优于其他算法,能够有效跳出局部最优解,特别是在复杂问题(如 F6)中接近理论最优解。同时,复合型函数测试验证了 IPKO 的鲁棒性和复杂性适应性,其高维优化问题中的收敛速度和求解精度显著优于其他算法,并保持稳定的结果。综上,IPKO 算法凭借其快速、精准的收敛能力以及高度一致的性能,展现出在多种优化问题中全面优于其他算法的显著优势。

图 7 箱线图展示了各算法在测试函数上的性能分布,反映了结果的范围和分布差异。可以看出,IPKO 在多个函数上表现出更集中的结果分布且更接近最优,而其他算

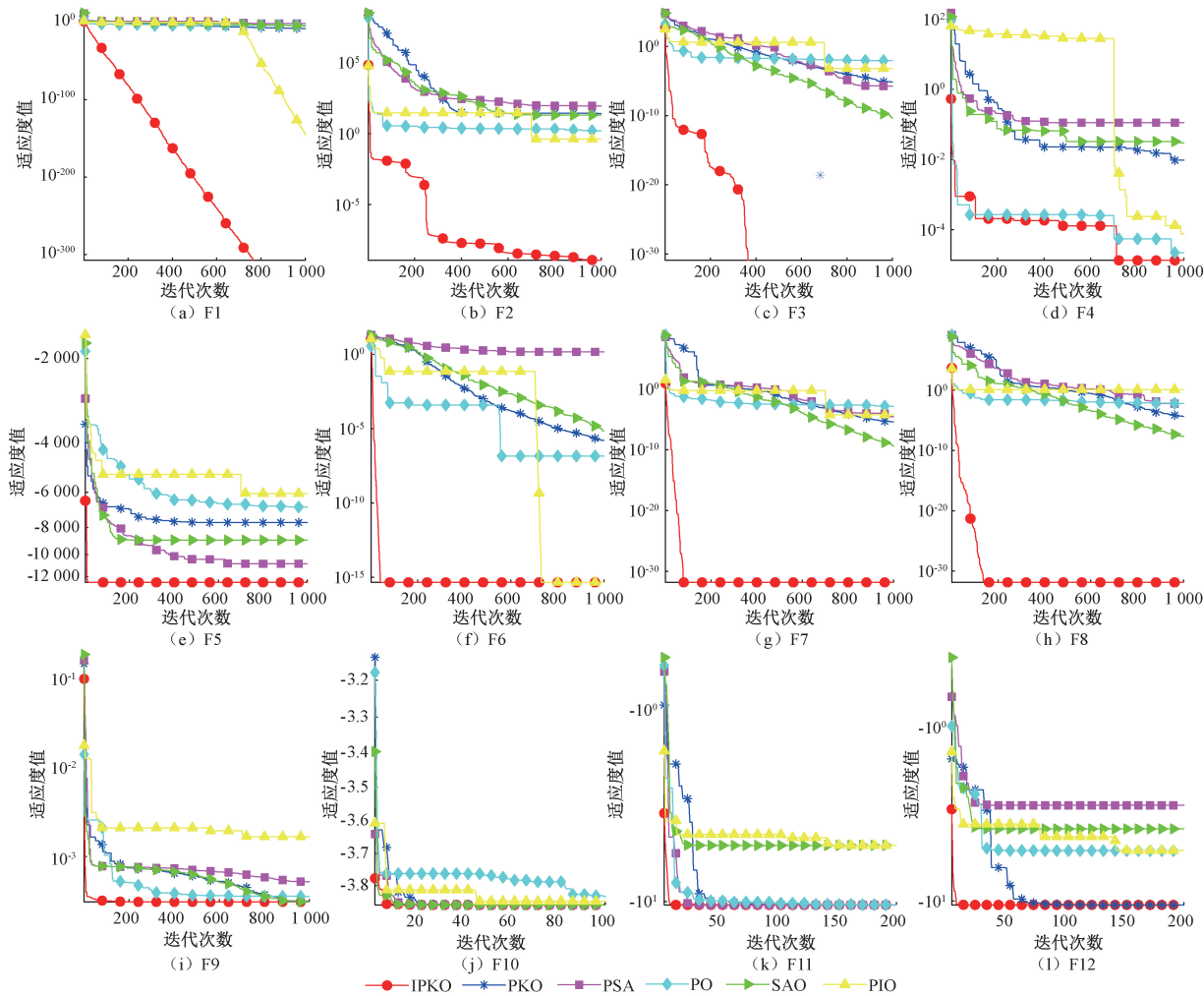


图 6 IPKO 算法与对比算法的收敛曲线图

Fig. 6 Convergence curves of IPKO algorithm and comparison algorithms

表 4 各个算法在 12 个测试函数的求解结果
Table 4 Optimization results of different algorithms on 23 test functions

函数代号	指标	IPKO	PKO	PSA	PSA	PSA
F1	Best	3.51×10^{-201}	0.000 415	7.88×10^{-09}	6.72×10^{-09}	0.002 154
	Std	0	0.042 782	0.000 107	6.67×10^{-06}	0.002 599
	Average	1.71×10^{-196}	0.020 363	6.52×10^{-05}	4.20×10^{-06}	0.004 825
	Median	8.28×10^{-199}	0.001 146	2.37×10^{-06}	1.66×10^{-06}	0.004 244
	Diff	8.46×10^{-196}	0.096 875	0.000 248	1.60×10^{-05}	0.007 965
F2	Best	0	28.413 5	0.240 27	0.153 88	28.788 1
	Std	7.31×10^{-08}	68.813 4	1.539 1	6.016 1	29.536 5
	Average	3.29×10^{-08}	93.529 6	1.640 1	6.669 1	58.867 5
	Median	9.69×10^{-22}	78.065 7	0.763 04	4.270 2	59.965 5
	Diff	1.64×10^{-07}	176.515	3.684 9	14.006 4	89.576 5
F3	Best	0	0.000 252	0.015 348	0.017 448	0.001 553
	Std	2.46×10^{-13}	0.023 881	0.032 871	0.021 861	0.003 448
	Average	1.10×10^{-13}	0.019 266	0.056 298	0.042 121	0.004 347
	Median	0	0.017 99	0.045 283	0.037 508	0.002 318
	Diff	5.51×10^{-13}	0.058782	0.10194	0.074438	0.008518
F4	Best	7.01×10^{-05}	0.024 947	3.60×10^{-05}	2.45×10^{-05}	0.023 79
	Std	8.50×10^{-05}	0.042 259	0.000 167	3.58×10^{-05}	0.029 999
	Average	0.000 164	0.057 629	0.000 203	7.66×10^{-05}	0.057 828
	Median	0.000 172	0.044 44	0.000 144	8.12×10^{-05}	0.053 922
	Diff	0.000 257	0.131 1	0.000 437	0.000 123	0.095 81
F5	Best	-12 569.5	-7567.61	-7442.55	-7207.21	-9714.83
	Std	1.73×10^{-11}	349.808 1	817.091 8	708.039 3	690.957 7
	Average	-12 569.5	-6 957.37	-6651.14	-6482.69	-8959.86
	Median	-12 569.5	-6 849.3	-6606.01	-6700.55	-8780.99
	Diff	-12 569.5	-6 729.59	-5409.5	-5414.6	-8064.29
F6	Best	4.44×10^{-16}	0.005 919	6.08×10^{-14}	4.44×10^{-16}	0.004 793
	Std	0	10.904 9	4.24×10^{-06}	2.80×10^{-05}	0.010 716
	Average	4.44×10^{-16}	8.004 9	3.19×10^{-06}	2.46×10^{-05}	0.016 176
	Median	4.44×10^{-16}	0.107 82	1.80×10^{-05}	1.81×10^{-05}	0.019 868
	Diff	4.44×10^{-16}	19.950 9	1.05×10^{-05}	6.55×10^{-05}	0.029 588
F7	Best	1.57×10^{-32}	0.002 309	0.000 494	0.000 683	3.48×10^{-05}
	Std	0	8.435 4	0.002 866	0.002 358	0.045 162
	Average	1.57×10^{-32}	7.504 7	0.003 459	0.003 01	0.024 55
	Median	1.57×10^{-32}	3.810 5	0.003 256	0.003 123	4.51×10^{-05}
	Diff	1.57×10^{-32}	16.767 7	0.007 344	0.006 605	0.104 05
F8	Best	1.35×10^{-32}	0.024 115	0.026 938	0.009 929	0.000 167
	Std	0	16.273 1	0.009 645	0.027 239	0.004 81
	Average	$1.35\text{E} \times 10^{-32}$	11.227 5	0.033 636	0.041 655	0.005 932
	Median	1.35×10^{-32}	0.854 43	0.028 612	0.036 739	0.004 375
	Diff	1.35×10^{-32}	36.810 7	0.049 907	0.077 38	0.012 985
F9	Best	0.000 307	0.000 336	0.000 31	0.000 308	0.000 319
	Std	9.36×10^{-07}	0.000 412	0.000 398	0.000 39	0.008 801
	Average	0.000 308	0.000 935	0.000 515	0.000 581	0.004 631

续表 4
Continuation table 4

函数代号	指标	IPKO	PKO	PSA	PSA	PSA
F10	Median	0.000 308	0.001 223	0.000 323	0.000 418	0.000 668
	Diff	0.000 31	0.001 223	0.001 224	0.001 238	0.020 363
	Best	-3.862 8	-3.862 8	-3.862 4	-3.862 5	-3.862 8
	Std	0	0	0.001 463	0.008 118	0
	Average	-3.862 8	-3.862 8	-3.861 1	-3.857 2	-3.862 8
	Median	-3.862 8	-3.862 8	-3.861 4	-3.86 1	-3.862 8
	Diff	-3.862 8	-3.862 8	-3.858 8	-3.843 1	-3.862 8
	Best	-10.402 9	-10.402 9	-5.087 6	-10.402 7	-10.402 9
F11	Std	0	2.377 1	5.55×10^{-05}	2.910 6	2.377 1
	Average	-10.402 9	-9.339 9	-5.087 5	-8.276	-6.150 7
	Median	-10.402 9	-10.402 9	-5.087 5	-10.399 5	-5.087 7
	Diff	-10.402 9	-5.087 7	-5.087 5	-5.087 6	-5.087 7
	Best	-10.536 4	-10.536 4	-10.536 2	-10.535 4	-10.536 4
F12	Std	8.88×10^{-16}	1.26×10^{-15}	2.417 2	2.960 6	2.623 5
	Average	-10.536 4	-10.536 4	-9.452 4	-8.371 5	-5.951 5
	Median	-10.536 4	-10.536 4	-10.531 8	-10.53 1	-5.128 5
	Diff	-10.536 4	-10.536 4	-5.128 3	-5.128 3	-3.835 4

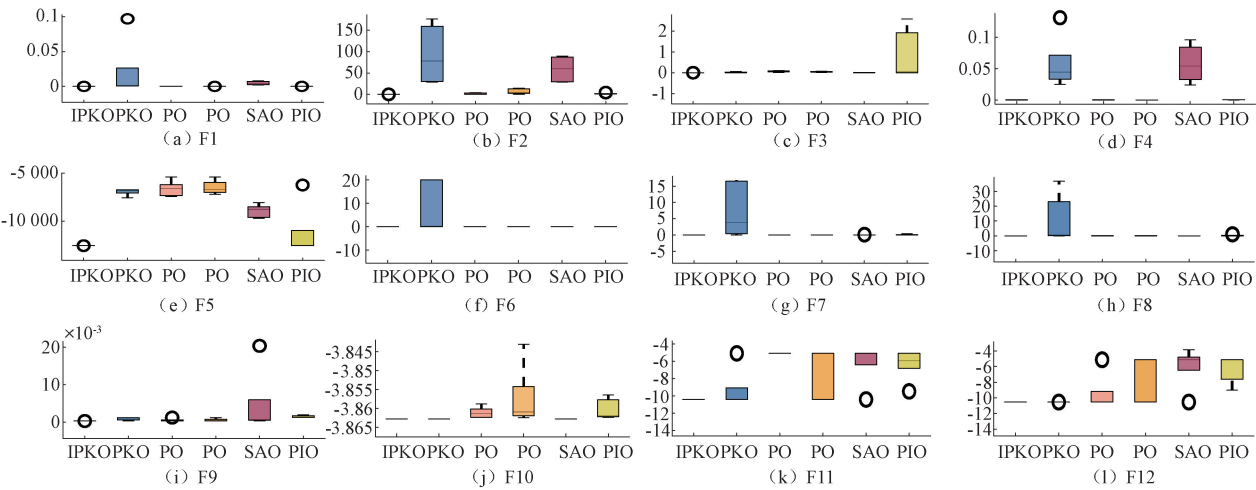


图 7 各算法在试函数上的箱线图

Fig. 7 Boxplot of algorithm performance on test functions

法则表现出较大的分布范围和偏差。进一步突出了 IPKO 算法在全局优化任务中的稳定性和优越性。

从统计学角度看,表 5 可知 Friedman 检验结果进一步验证了 IPKO 的优势,其平均排名(1.391 304)显著优于其他算法,且 p 值($5.916 2 \times 10^{-7}$)表明该优势具有统计显著性。整体而言,IPKO 在多种测试函数中展示了广泛的适用性和稳定性,有效解决了复杂的全局优化问题,展现了卓越的优化性能和强大的适应能力。

表 5 各算法 Friedman 值排名

Table 5 Friedman ranking of different algorithms

算法	Friedman	排名
IPKO	1.250 000	1
PKO	5.500 000	5
PSA	3.833 333	4
PO	2.833 333	3
SAO	5.500 000	5
PIO	2.083 333	2

4 IPKO 用于求解微电网中优化调度问题

4.1 微电网优化调度问题

微电网系统主要包括光伏发电 (photovoltaic, PV)、风力发电 (wind turbine, WT)、柴油发电 (diesel engine, DE)、微型燃气轮机 (microturbine, MT)、以及用于削峰填谷的储能蓄电池 (storage battery, BT) 构成。考虑微电网系统的运行成本和环境成本,建立经济优化调度模型。该模型涵盖发电成本、设备维护成本、电能交易成本以及污染物排放引发的环境成本。优化目标是最小化微电网系统的日总运行成本,同时满足相关约束条件。微电网的数学模型如下:

1) 目标函数

(1) 运行成本 C_{OM_F}

$$C_{BESS} = \sum_{i=49}^{72} 0.026 |x_i| \quad (20)$$

$$C_{DE} = \sum_{i=49}^{96} (0.128x_i^3 + 0.00011x_i^2 + 0.180x_i + 6) \quad (21)$$

$$C_{MT} = \sum_{i=97}^{120} \left(\frac{0.0293x_i + (2.55/9.8)x_i}{0.0753x_i^3 - 0.3095x_i^2 + 0.4174x_i + 0.1068} \right) \quad (22)$$

$$C_{Grid} = C_{grid_buy} - C_{grid_sell} \quad (23)$$

$$C_{OM_F} = C_{DE} + C_{MT} + C_{BESS} + C_{Grid} \quad (24)$$

(2) 环境保护成本 C_{en}

$$C_{DE_{en}} = 264.282 \sum_{i=73}^{96} x_i \quad (25)$$

$$C_{MT_{en}} = 18.254 \sum_{i=97}^{120} 6x_i \quad (26)$$

$$C_{Grid_{en}} = 98.196 \sum_{i=121}^{144} (P_{load,i} - generation_i) \quad (27)$$

$$C_{en} = C_{DE_{en}} + C_{MT_{en}} + C_{Grid_{en}} \quad (28)$$

$$\min f = C_{OM_F} + C_{en} \quad (29)$$

2) 约束条件

$$0 \leq x_n \leq PV_n, n \in [1, 24] \quad (30)$$

$$0 \leq x_n \leq WT_{n-24}, n \in [25, 48] \quad (31)$$

$$-30 \leq x_n \leq 30, n \in [49, 72] \quad (32)$$

$$6 \leq x_n \leq 30, n \in [73, 96] \quad (33)$$

$$3 \leq x_n \leq 30, n \in [97, 120] \quad (34)$$

$$-60 \leq x_n \leq 60, n \in [121, 144] \quad (35)$$

4.2 仿真参数

采用本文提出的 IPKO 算法进行微电网的经济优化调度。设置不同的负荷环境 (如高峰和低谷) 后,按照表 6 分时电价对微电网进行多目标优化调度。

4.3 仿真结果分析

光伏、风力和负荷功率预测如图 8 所示,并网、柴油发电、储能系统、微燃机出力如图 9 所示,光伏、风电、负荷的出力如

图 10 所示,微电网分布式电源功率平衡如图 11 所示。

表 6 分时电价表

Table 6 Time-of-use electricity pricing table

分时阶段	时段划分	电价/(元/kWh)	
		购电	售电
峰时段	10:00~15:00, 18:00~22:00	1.35	0.36
平滑段	08:00~09:00, 16:00~18:00	0.82	0.36
谷时段	01:00~07:00, 23:00~06:00	0.38	0.36

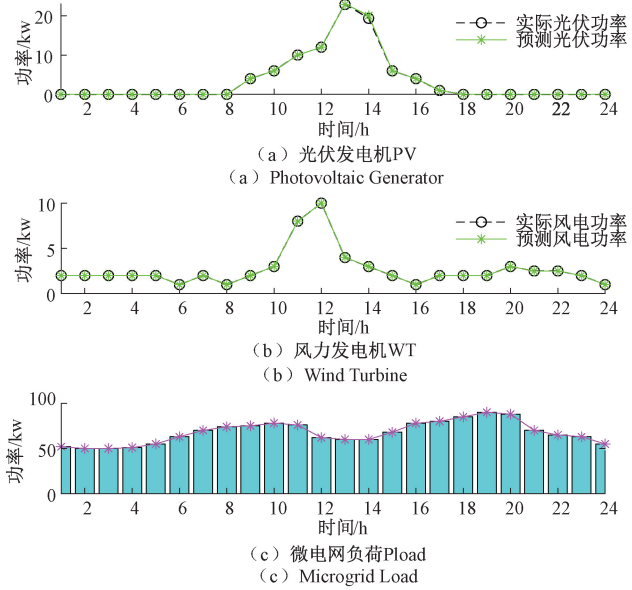


图 8 光伏、风电和微电网负荷功率预测曲线

Fig. 8 Prediction curves of photovoltaic, wind power, and microgrid load power

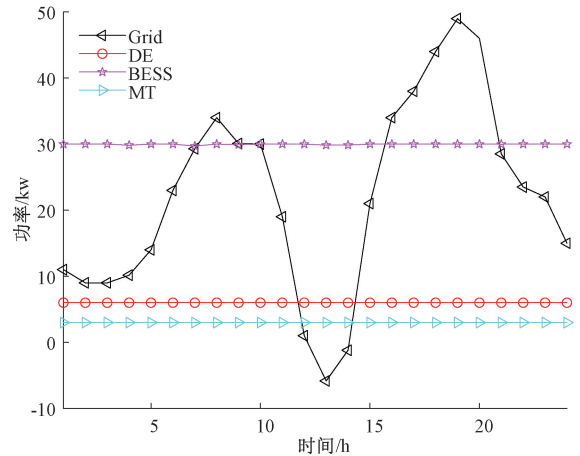


图 9 大电网、柴油发电、储能系统、微燃机出力曲线

Fig. 9 Output curves of the large power grid, diesel power generation, energy storage system, and microturbine

如表 7 所示,对比了 PKO、PSA、PO 算法与本文所提 IPKO 算法在相同工作条件下的优化结果。IPKO 算法在

优化后的最低成本相比于 PKO、PSA、PO 分别降低了约 1.92%、7.38%、9.65%，表明 IPKO 优化调度问题上具有更优秀和稳定的性能。

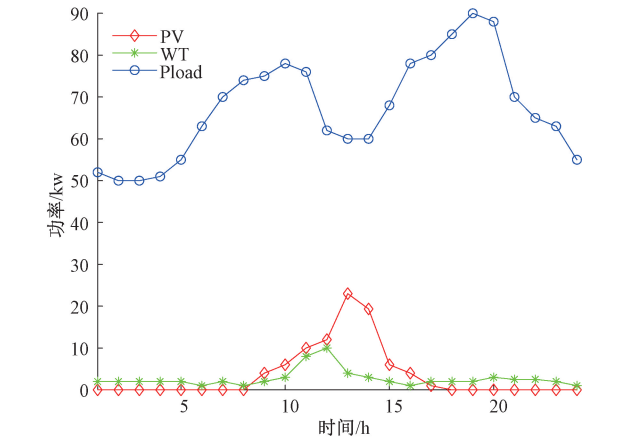


图 10 光伏、风电、负荷的出力曲线

Fig. 10 Power output curves of photovoltaic, wind power, and load

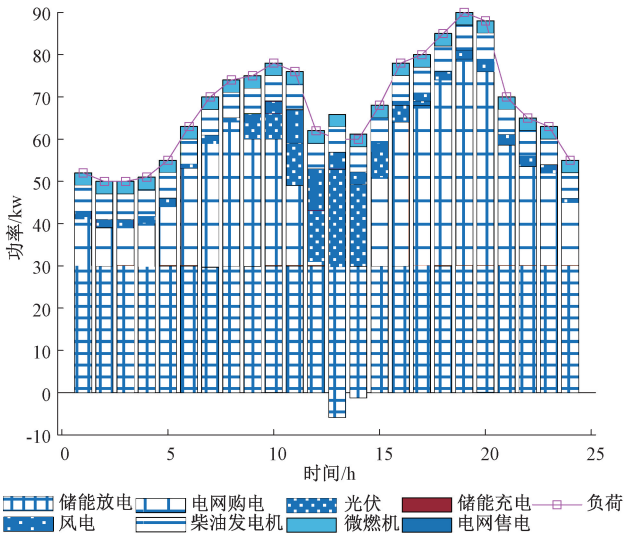


图 11 功率平衡图

Fig. 11 Power balance

表 7 各算法最低总成本对比结果

Table 7 Comparison of the minimum total cost for different algorithms	
算法	优化后的最低总成本/元
IPKO	44 386.359 2
PKO	45 254.682 7
PSA	47 923.808 4
PO	49 126.881 3

5 结 论

为提升 PKO 算法的优化性能,本文提出了一种基于多策略改进的 IPKO 算法。具体地,采用 LHS 方法,以避

免在高维问题中出现样本分布过度集中或忽略潜在优良区域的情况,从而降低陷入局部最优的风险。同时,引入鱼鹰算法中的定位捕鱼机制,以增强对最优区域的探索能力,并提高逃避局部最优的效果。此外,融合坠落机制以提升搜索能力和稳定性,从而有效防止过早收敛。最后,通过变异率终止条件的自适应调整,在全局探索与局部开发之间实现动态平衡,从而进一步提升解的质量并优化搜索效率。

通过 3 个实验,分别评估了算法在不同特征维度下的准确性差异、运行时间和内存消耗,从而分析了其拟合能力、预测能力、计算效率及高维适应性。此外,还考察了种群大小和最大迭代次数对算法性能的影响。为验证改进策略的有效性,采用了 12 个标准测试函数进行验证,并将 IPKO 与其他群体智能优化算法进行对比。通过 Fridman 检验,实验结果表明,IPKO 在收敛精度、收敛速度及鲁棒性方面均优于原始 PKO 及其他对比算法。最后,将 IPKO 应用于微电网经济优化调度问题,验证了其在实际问题中的可靠性与适用性。未来,考虑进一步提升 IPKO 的收敛性能,并探索其在更复杂实际问题中的应用潜力。

参考文献

[1] BOUAOUDA A, HASHIM F A, SAYOUTI Y, et al. Pied kingfisher optimizer: A new bio-inspired algorithm for solving numerical optimization and industrial engineering problems[J]. Neural Computing and Applications, 2024, DOI: 10.1007/s00521-024-09879-5.

[2] HEIDARI A, MIRALILI S, FARIS H, et al. Harris hawks optimization: Algorithm and applications[J]. Future Generation Computer Systems, 2019, 97: 849-872.

[3] MIRJALILI S, LEWIS A. The whale optimization algorithm[J]. Advances in Engineering Software, 2016, 95: 51-67.

[4] MIRJALILI S. SCA: A sine cosine algorithm for solving optimization problems[J]. Knowledge-Based Systems, 2016, 96: 120-133.

[5] ZHAO Z S, FENG X, LIN Y Y, et al. Evolved neural network ensemble by multiple heterogeneous swarm intelligence[J]. Neurocomputing, 2015, 149: 29-38.

[6] MIRJALILI S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm [J]. Knowledge-Based Systems, 2015, 89(7): 228-249.

[7] 全年,李燕青,申宏威,等.基于改进黑洞算法的微电网优化调度[J].电力科学与技术学报,2021,36(5): 113-119.

TONG N, LI Y Q, SHEN H W, et al. Microgrid

- optimization scheduling based on improved black hole algorithm[J]. Journal of Electric Power Science and Technology, 2021, 36(5): 113-119.
- [8] 李希俊,郝飞,李杰,等.基于非合作博弈的多微电网协同运行策略[J/OL]. 电气工程学报, 1-11[2024-08-09]. <http://kns.cnki.net/kcms/detail/10.1289.TM.20240809.1123.008.html>.
- LI X J, HAO F, LI J, et al. Cooperative operation strategy of multiple microgrids based on non-cooperative game [J/OL]. Journal of Electrical Engineering, 1-11[2024-08-09]. <http://kns.cnki.net/kcms/detail/10.1289.TM.20240809.1123.008.html>.
- [9] 张林,郭辉,姚李孝.基于改进蝙蝠算法的微电网优化研究[J]. 电网与清洁能源, 2021, 37(4): 122-126.
- ZHANG L, GUO H, YAO L X. Microgrid optimization research based on improved bat algorithm[J]. Power Grid and Clean Energy, 2021, 37(4): 122-126.
- [10] 邢毓华,任甜甜.改进 MOPSO 在微电网优化调度中的应用研究[J]. 太阳能学报, 2024, 45(6): 191-200.
- XING Y H, REN T T. Application research of improved MOPSO in microgrid optimization scheduling[J]. Solar Energy Journal, 2024, 45(6): 191-200.
- [11] 蒋翱微,刘文红.基于改进蜣螂优化算法的无人机三维路径规划[J]. 电子测量技术, 2024, 47(13): 128-135.
- JIANG AO H, LIU W H. Three-dimensional path planning of UAV based on improved dung beetle optimization algorithm [J]. Electronic Measurement Technology, 2024, 47(13): 128-135.
- [12] 王雨虹,王志中,付华,等.多策略改进麻雀算法与 Bi-LSTM 的变压器故障诊断研究[J]. 仪器仪表学报, 2022, 43(3): 87-97.
- WANG Y H, WANG ZH ZH, FU H, et al. Research on transformer fault diagnosis based on multi-strategy improved sparrow algorithm and Bi-LSTM[J]. Chinese Journal of Scientific Instrument, 2022, 43(3): 87-97.
- [13] 万怡华,张雪梅.混合多策略改进蜣螂算法的避障路径规划[J]. 电子测量技术, 2024, 47(2): 69-78.
- WAN Y H, ZHANG X M. Hybrid multi-strategy improved dung beetle algorithm for obstacle avoidance path planning [J]. Electronic Measurement Technology, 2024, 47(2): 69-78.
- [14] 徐明,王风富,龙文.多策略改进的徒步优化算法及其应用[J/OL]. 电子测量技术, 1-15[2025-02-09]. <http://kns.cnki.net/kcms/detail/11.2175.TN.20250109.1346.031.html>.
- XU M, WANG F F, LONG W. Multi-strategy improved walking optimization algorithm and its application [J/OL]. Electronic Measurement Technology, 1-15[2025-02-09]. <http://kns.cnki.net/kcms/detail/11.2175.TN.20250109.1346.031.html>.
- [15] 杨宇鸽,郝杨杨,王逸文.基于威布尔飞行和警戒机制的沙猫群优化算法及应用[J]. 计算机工程与应用, 2025, 61(2): 145-157.
- YANG Y G, HAO Y Y, WANG Y W. Sand cat swarm optimization algorithm based on Weibull flight and vigilance mechanism and its applications [J]. Computer Engineering and Applications, 2025, 61(2): 145-157.
- [16] DEGHANI M, TROJOVSKY P. Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems[J]. Frontiers in Mechanical Engineering, 2023, 8: 1126450.
- [17] 李一铭,王跟成.基于坠落机制的混沌麻雀算法 AGV 路径规划[J]. 制造技术与机床, 2023(1): 102-108.
- LI Y M, WANG G CH. AGV path planning based on the chaotic sparrow algorithm with falling mechanism [J]. Manufacturing Technology and Machine Tools, 2023(1): 102-108.
- [18] DENG L, LIU S. Snow ablation optimizer: A novel metaheuristic technique for numerical optimization and engineering design [J]. Expert Systems with Applications, 2023, 225: 120069.
- [19] DUAN H, QIAO P. Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning [J]. International Journal of Intelligent Computing and Cybernetics, 2014, 7(1): 24-37.
- [20] GAO Y. PSA-based search algorithm: A novel metaheuristic algorithm based on PSA algorithm[J]. Expert Systems with Applications, 2023, 232: 120886.
- [21] LIAN J, HUI G, MA L, et al. Parrot optimizer: Algorithm and applications to medical problems [J]. Computers in Biology and Medicine, 2024, 172: 108064.

作者简介

何印,硕士研究生,主要研究方向为智能算法与微电网优化调度。

E-mail:15985038560@163.com

孔玲玲(通信作者),副教授,主要研究方向为电网中电力电子技术和电气工程教育教学。

E-mail:Kongling0104@163.com

郑哲明,硕士研究生,主要研究方向为风电预测。

E-mail:18016697736@163.com