

基于动态惯性权重与莱维飞行的改进粒子群算法

冯若雨

河北地质大学信息工程学院, 河北 石家庄

收稿日期: 2025年7月15日; 录用日期: 2025年8月14日; 发布日期: 2025年8月21日

摘要

针对粒子群算法在解决高维优化问题时存在的易陷入局部最优的问题, 文中引入了具有下降趋势的动态变化因子, 提出了一种融合莱维飞行机制的带有动态惯性权重的改进粒子群优化算法(WLPSO)。该算法在位置更新的惯性权重部分引入了一个自适应值, 来平衡局部搜索与全局探索能力, 同时引入莱维飞行策略增强算法跳出局部最优的能力。在搜索过程中, 莱维飞行的长步长特性有效引导粒子群在最优解空间进行高效探索, 显著提升了收敛速度。改进后的粒子群算法与其他3种优化算法在9个经典测试函数上进行仿真实验, 结果表明改进的粒子群算法在收敛速度和收敛精度方面的综合表现都优于其它算法。

关键词

粒子群算法, 自适应动态惯性权重, 莱维飞行, 全局最优

Improved Particle Swarm Algorithm Based on Dynamic Inertia Weight and Levi's Flight

Ruoyu Feng

School of Information Engineering, Hebei GEO University, Shijiazhuang Hebei

Received: Jul. 15th, 2025; accepted: Aug. 14th, 2025; published: Aug. 21st, 2025

Abstract

Aiming at the problem that the particle swarm optimization algorithm is prone to fall into local optimum when solving high-dimensional optimization problems, a dynamic change factor with a

downward trend is introduced in this paper, and an improved particle swarm optimization algorithm with dynamic inertia weight (WLPSO) integrating the Levi flight mechanism is proposed. This algorithm introduces an adaptive perturbation value in the inertia weight part of position update to balance the local search and global exploration capabilities. Meanwhile, it introduces the Levi flight strategy to enhance the algorithm's ability to escape from local optima. During the search process, the long step size characteristic of Levi's flight effectively guides the particle swarm to conduct efficient exploration in the optimal solution space, significantly improving the convergence speed. The improved particle swarm optimization algorithm was subjected to simulation experiments with three other optimization algorithms on nine classical test functions. The results show that the comprehensive performance of the improved algorithm in terms of convergence speed and convergence accuracy is superior to that of the other algorithms.

Keywords

Particle Swarm Algorithm, Adaptive Dynamic Inertia Weight, Levi's Flight, Global Optimum

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

粒子群算法(Particle swarm optimization, PSO)在 1995 年 Eberhart 博士和 Kennedy 博士受鸟类捕食启发而提出[1], 由于粒子群算法的实现容易、精度高以及收敛快等特点, 在实际应用中展现了很好的优越性。虽然粒子群算法在解决高维优化问题时具有较强的收敛能力, 但该算法在寻优过程当中, 粒子容易陷入局部最优, 导致问题求解的结果不太理想。这一缺陷, 逐渐引起了学术界的重视。

在理论研究方面, Clerc 引入了收敛因子, 为算法参数选择进行数学分析, 提供理论依据, 保证算法的收敛性能[2]; Ozcan 和 Mohan 指出粒子在离散时间状态下为一个连续的正弦波形, 粒子运动本质上是连续正弦波形的跃迁过程, 以此来寻找最优解[3]; 国内学者金鑫磊、马龙华、吴铁军等通过建立均方收敛的充分条件, 为算法的稳定性研究奠定了重要理论基础[4]。

在应用领域方面, 国内学者刘国海、万亚连、沈跃等在标准 PSO 基础上, 通过在惯性权重值上引入方向角, 使惯性权重随迭代次数的增加而非线性减小, 解决路径规划问题[5]; 陈攀、孙鉴、吴佳伟等加入反向学习和正弦余弦算法两个过程提高粒子的寻优能力, 解决云计算任务调度问题[6]; 刘虹伶和时维国, 提出了一种结合灰狼优化算法的改进粒子群优化算法, 通过引入反向学习和混沌映射对最优粒子进行扰动, 求解环境经济调度问题[7]; 吴鹏、叶宝林、吴维敏等利用混沌运动增强全局搜索能力以克服局部最优, 实现对交通信号的控制[8]; 段艳和潘峰提出了一种融合改进莱维飞行策略的混沌粒子群算法, 通过引入改进的混沌序列来初始化粒子群并利用莱维飞行来调整粒子的速度和位置[9]。

针对粒子群算法易陷入局部最优的问题, 本文提出了一种融合莱维飞行机制的带有动态惯性权重的改进粒子群算法(WLPSO)。自适应动态惯性权重的值符合指数衰减函数, 使得惯性权重值前期注重全局搜索快速缩小搜索范围, 后期进行精细搜索找到最优解。鉴于粒子易陷入局部最优, 又将位置更新不变的粒子或收敛前期的粒子, 进行莱维飞行的扰动, 扰动方向由粒子全局最优解方向决定。采用 9 个经典的测试函数对所提出的算法进行验证, 并与其他 3 种优化算法进行了对比。结果表明, 改进后的算法在寻优质量、稳定性、收敛速度等方面更具优越性。

2. PSO 算法

PSO 算法是一种受自然界群体智能行为启发的随机搜索算法，其灵感来源于鸟群协同觅食的生物现象。该算法模拟了鸟群在未知环境中寻找食物的过程：假设在一个搜索区域内存在单一食物源，虽然每只鸟都无法直接获知食物的具体位置，但它们能够通过感知当前距离食物的远近来进行协作搜索。在此过程中，群体中的个体通过追踪距离食物最近的同伴所在区域，逐步向最优解靠近。

PSO 算法中，所有的粒子构成种群，并且每一个粒子都可能是所研究优化问题的最优解。粒子在搜索空间中根据个体的最优位置和群体最优位置更新速度和位置，从而寻找最优解[10]。设该种群由 N 个粒子组成，记为集合 $I = \{1, 2, 3, \dots, N\}$ 。设有一个 D 维的搜索空间，其中第 i 个粒子就表示为一个 D 维的向量，其粒子的速度也是一个 D 维的向量，分别记为：

$$X_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{iD}), i \in I$$

$$V_i = (v_{i1}, v_{i2}, v_{i3}, \dots, v_{iD}), i \in I$$

令 $P_{best} = (p_{i1}, p_{i2}, \dots, p_{iD})$ 表示粒子 i 到目前为止的最优个体位置， $G_{best} = (g_{i1}, g_{i2}, \dots, g_{iD})$ 表示种群整体的全局最优位置。找到个体最优值和全局最优值后，第 i 粒子在第 $k+1$ 次迭代时，根据公式(1)和公式(2)来更新自己的速度和位置：

$$V_i^{(k+1)} = wV_i^{(k)} + c_1r_1(P_{best}^{(k)} - X_i^{(k)}) + c_2r_2(G_{best}^{(k)} - X_i^{(k)}) \quad (1)$$

$$X_i^{(k+1)} = X_i^{(k)} + V_i^{(k+1)} \quad (2)$$

其中， w 是惯性权重， r_1, r_2 都是 D 维向量，其每个分量都是 $[0, 1]$ 范围内的均匀随机数； c_1 表示粒子向自身历史最优位置逼近的加速度系数； c_2 表示粒子向群体的全体最优位置逼近的加速度系数。

3. 改进的粒子群算法

本文提出一种融合自适应动态惯性权重和莱维飞行的改进粒子群算法(WLPSO)。

3.1. 自适应动态惯性权重

标准粒子群优化算法的位置更新机制包含三个关键组成部分：粒子当前惯性方向、个体历史最优位置和群体全局最优位置。其中，惯性权重参数对算法的搜索行为具有决定性影响，当参数值较大时，算法中粒子更依赖当前速度，倾向于大范围探索的全局搜索，而当参数值较小时，粒子则更依赖个体最优和全局最优的引导，倾向于精细搜索的局部开发，从而提升算法的搜索效率。然而，由于标准粒子群算法存在着参数的变化，因此它在计算粒子位置时存在较大的不确定性，从而限制了粒子的准确搜索能力。

为了提高搜索速度，Shi 和 Eberhart 博士先后在 1998 年和 1999 年提出了 PSO-LDIW 算法[11]-[13]，该算法采用了线性下降的惯性权重：

$$w = w_{\max} - (w_{\max} - w_{\min}) \times \frac{iter}{\max iter} \quad (3)$$

其中， w_{\max} 和 w_{\min} 分别表示惯性权重的最大值和最小值， $iter$ 和 $\max iter$ 分别表示当前迭代次数和最大迭代次数。然而，简单线性变化的惯性权重策略难以动态适应复杂的优化过程，导致算法存在搜索精度不足、收敛稳定性欠佳等问题。

针对这一局限性，本文提出一种改进的指数衰减自适应权重调节机制。通过在算法中引入动态权重因子，使惯性权重能够根据搜索状态智能调整。该改进策略的数学表达为：

$$w = w_{\min} - (w_{\max} - w_{\min}) \times e^{-k \times \frac{iter}{\max iter}} \quad (4)$$

其中, w_{\max} 和 w_{\min} 分别表示惯性权重的最大值和最小值, $iter$ 和 $\max iter$ 分别表示当前迭代次数和最大迭代次数, k 表示衰减函数的衰减速率。

这种自适应调节机制通过粒子群的分布特征和收敛程度分析该算法的开发程度, 依据这些实时反馈信息, 自动平衡全局探索与局部开发, 使得算法在搜索初期能保持较强的全局探索能力, 而在接近最优解区域时则自动增强局部精细搜索的强度。

3.2. 莱维飞行策略

从数学本质上分析, 莱维飞行是一种具有短距离搜索兼长程跳跃的高斯随机过程[14], 其核心特征表现为短距离精细搜索与长距离随机跳跃的有机结合, 实现了全局探索与局部开发的动态平衡。智能优化算法领域, 步长控制参数通常设置在 0.01 到 1 之间, 以确保算法在搜索空间中进行有效探索。张海阔团队研究[15]表明通过将莱维飞行的步长参数设置为 0.01, 能够在粒子群优化算法中实现两个关键突破: 其一, 该参数设置有效抑制了粒子群的过早聚集现象, 维持了种群的多样性; 其二, 通过适度调控搜索范围, 使算法在保持局部搜索能力的同时, 显著增强了全局探索性能。相关研究进一步揭示: 较小的步长参数有利于保持种群多样性, 而较大的参数值则能拓展搜索空间, 两者协同作用可显著降低算法陷入局部最优的可能性。

为便于算法实现与工程应用, 普遍采用文献[16]提出的标准化建模方法对莱维分布进行数值模拟, 该方法的数学表达为:

$$\left\{ \begin{array}{l} Levy = \frac{\mu}{|v|^{\frac{1}{\beta}}}, \mu \sim N(0, \sigma_{\mu}^2), v \sim N(0, \sigma_v^2) \\ \sigma_{\mu} = \left\{ \begin{array}{l} \Gamma(1+\beta) \times \sin\left(\frac{\pi \times \beta}{2}\right) \\ \Gamma[(1+\beta)/2] \times \beta \times 2^{\frac{\beta-1}{2}} \end{array} \right\} \\ \sigma_v = 1 \end{array} \right. \quad (5)$$

其中, $0 < \beta < 2$, 通常取 1.5; μ, v 均服从正态分布; Γ 是标准伽马函数; $Levy(\lambda)$ 符合莱维分布的随机路径。本文提出的改进算法中, 动态惯性权重机制已经能有效保证精细搜索, 因此莱维飞行策略的主要功能为: 当粒子群出现停滞现象或在迭代初期时, 跳出局部最优, 增加发现全局最优解的机会。基于此设计理念, 算法将莱维飞行的 β 参数设定为 1.2。

由于莱维飞行方向选择确定遵循随机且各向同性的原则, 其运动方向在搜索空间中呈均匀分布, 即在所有可能的方向上均匀随机选择, 确保探索过程不会对任何特定方向产生偏好, 这种完全随机的方向选择机制从概率上保证了搜索轨迹能够均匀覆盖整个解空间。但在改进的粒子群算法中需要粒子有向全局最优位置靠拢的趋势, 所以用步长乘以 $(X^{(k)} - G_{best}^{(k)})$ 形成方向性引导。当粒子远离全局最优时, $(X^{(k)} - G_{best}^{(k)})$ 较大, 莱维步长被放大, 帮助粒子跳出局部最优; 粒子接近全局最优时, $(X^{(k)} - G_{best}^{(k)})$ 较小, 莱维步长衰减, 转为局部精细搜索。那么, 莱维飞行的公式则变为:

$$Levy = \frac{\mu}{|v|^{\frac{1}{\beta}}} \otimes (X^{(k)} - G_{best}^{(k)}) \quad (6)$$

针对这一改进，粒子群中的粒子位置更新公式变为：

$$X^{(k+1)} = X^{(k)} + V^{(k+1)} + Levy \quad (7)$$

但为了避免过早收敛，希望该改进的粒子群算法能够采用动态激活策略来优化莱维飞行机制。在迭代初期阶段，当粒子群尚未充分探索搜索空间时，主动激活莱维飞行的长跳跃特性，以增强算法的全局探索能力；随着迭代过程的推进，当检测到粒子群出现停滞现象，即群体最优解连续多代未改变时，选择性触发莱维飞行机制以帮助种群跳出局部最优；在算法收敛后期，则逐步降低莱维飞行的激活频率，使算法能够专注于最优解邻域内的精细搜索。这种基于迭代阶段和种群状态的双重判定机制，既有效避免了过早收敛问题，又确保了算法在全局探索和局部开发之间实现动态平衡。

3.3. 改进粒子群算法的构建

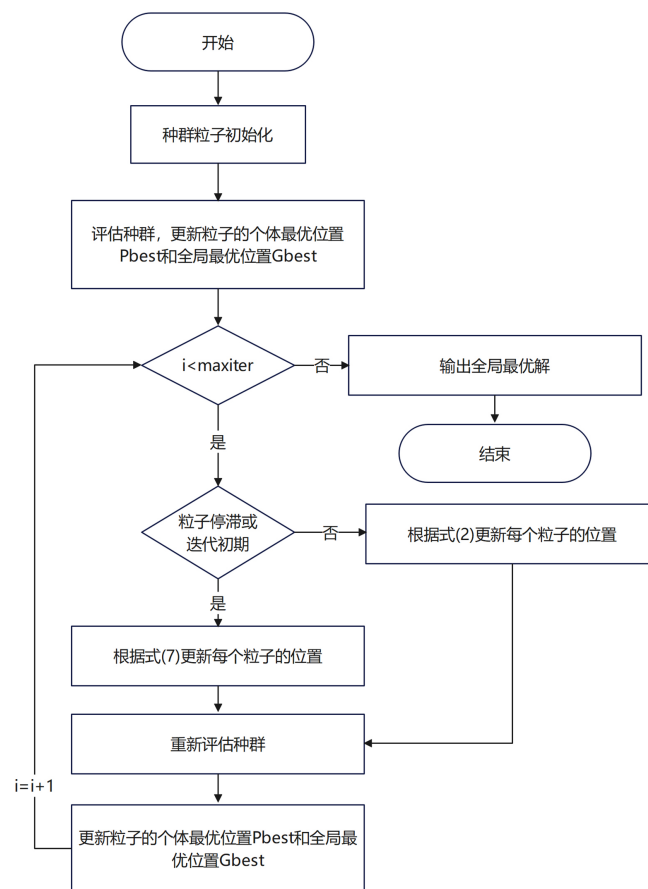


Figure 1. The flowchart of the WLPSO algorithm

图 1. 基于动态惯性权重与莱维飞行的改进粒子群算法流程图

本文提出一种带有自适应动态惯性权重和莱维飞行的粒子群优化(WLPSO)算法。

基于动态惯性权重与莱维飞行的改进粒子群算法的具体步骤如下：

步骤 1 设置种群参数；

步骤 2 在定义域内随机初始化粒子种群；

步骤 3 评估整个种群，更新粒子的个体最优位置 Pbest 和全局最优位置 Gbest；

步骤 4 判断是否满足粒子停滞或迭代前期条件，若不满足则使用式(2)更新每个粒子的位置；若满足，

则使用式(7)更新每个粒子的位置:

步骤 5 进行边界处理;

步骤 6 重新评估种群, 计算粒子的个体最优位置 P_{best} 和全局最优位置 G_{best} ;

步骤 7 判断是否满足终止条件, 若不满足则返回步骤 4; 若满足, 则算法终止。

改进后的基于动态惯性权重与莱维飞行的改进粒子群算法的流程图如图 1 所示。

3.4. 参数选择

改进后的粒子群算法有四个权重参数, 其中权重 w 是用来控制惯性影响的, 由式(4)决定。加速度系数 c_1, c_2 的参数设定是由 Clerc 和 Kennedy (2002) [17] 提出的 PSO-CK 算法决定, 其中 $c_1 = c_2 = 1.49$ 。莱维飞行中步长控制参数选用张海阔团队的研究参数 0.01。

4. 仿真实验与结果分析

为验证 WLPSO 算法的优化性能, 本文选取三种典型优化算法作为对比, 所有算法均采用统一实验设置: 粒子规模 30 个, 最大迭代次数 1000 次, 为确保实验结果的可靠性, 每个测试函数均独立进行 30 次, 以消除随机因素带来的偏差。通过多轮独立实验的统计分析, 迭代曲线分析算法的收敛速度和搜索效率, 平均值和标准差评估该算法的收敛精度和鲁棒性, 从而客观验证 WLPSO 算法的有效性和优越性。

4.1. 测试函数

Table 1. Benchmark function

表 1. 基准函数

函数	名字	公式	搜索空间
$f_1(x)$	Schwefel 1.2	$F_1(x) = \sum_{i=1}^D \left(\sum_{j=1}^i x_j^2 \right)^2$	$[-100, 100]$
$f_2(x)$	Schwefel 2.22	$F_2(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i $	$[-10, 10]$
$f_3(x)$	Schwefel 2.21	$F_3(x) = \max_i \{ x_i \}$	$[-100, 100]$
$f_4(x)$	Step	$F_4(x) = \sum_{i=1}^D (\lfloor x_i + 0.5 \rfloor)^2$	$[-100, 100]$
$f_5(x)$	Rastrigin	$F_5(x) = \sum_i x_i^2 - 10 \cos(2\pi x_i) + 10$	$[-5.12, 5.12]$
$f_6(x)$	Griewank	$F_6(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]$
$f_7(x)$	Sphere	$F_7(x) = \sum_{i=1}^D x_i^2$	$[-100, 100]$
$f_8(x)$	Schwefel 2.26	$F_8(x) = -\sum_{i=1}^D x_i \sin(\sqrt{ x_i })$	$[-500, 500]$
$f_9(x)$	Rosenbrock	$F_9(x) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2]$	$[-30, 30]$

本文采用 9 个标准测试函数对改进粒子群算法进行性能验证, 测试函数如表 1。实验设置如下: 在高

维搜索空间中, 初始化 30 个粒子进行优化搜索, 最大迭代次数设为 1000 次。为消除随机性干扰, 每个测试函数独立运行 30 次并取平均结果作为最终性能指标。在每次实验开始时, 随机生成粒子的初始位置和速度, 计算各粒子的适应度值, 并据此确定个体最优位置和群体全局最优位置。

4.2. 实验结果与分析

为了验证改进的粒子群算法在高维搜索空间的优越性, 本文选用标准粒子群优化算法(PSO)、蒲公英优化器(DO)和差分进化算法(DE)三种典型优化算法进行对比, 采用平均值和标准差作为评价指标。对于相同的测试函数, 算法输出的平均值直接反映其收敛精度, 数值越小表明解的质量越高; 而标准差则用于衡量算法的稳定性, 数值越小代表算法鲁棒性越强。实验结果如表 2 和图 2 所示。

表 2 表明, 与其他三种算法相比, WLPSO 算法在标准测试函数 $F_1(x) \sim F_4(x)$ 和 $F_7(x) \sim F_9(x)$ 上的平均值和标准差最小。虽然 WLPSO 在 $F_5(x)$ 和 $F_6(x)$ 上的平均值和标准差不是最小, 但其标值明显小于标准 PSO 算法, 这表明改进的粒子群算法在搜索这九个标准测试函数时具有较高的收敛精度和稳定性。

Table 2. Comparison of four algorithms on the test function
表 2. 四种算法在测试函数上的对比表

函数	评价指标	PSO	DO	DE	WLPSO
$f_1(x)$	平均值	8.4463e+02	1.1037e-09	5.6108e+00	1.0586e-55
	标准差	1.3503e+02	1.2624e-09	7.9348e+00	1.1671e-55
$f_2(x)$	平均值	7.0891e+00	9.4466e-15	5.2856e-84	4.6059e-11
	标准差	1.3360e-15	4.1550e-12	7.4749e-84	1.0041e-01
$f_3(x)$	平均值	7.6863e+00	1.6164e-22	1.7528e+01	1.2855e-07
	标准差	3.1157e-01	3.0223e-08	3.9144e+00	2.1307e-22
$f_4(x)$	平均值	3.3615e+02	7.6453e-12	3.1691e-02	1.5407e-32
	标准差	1.3074e-32	9.2806e-13	4.4818e-02	1.0627e+01
$f_5(x)$	平均值	5.2733e+01	1.8207e+02	6.1048e+01	2.5316e+02
	标准差	2.7966e+01	1.5230e+01	2.8141e+00	1.6157e+01
$f_6(x)$	平均值	8.0068e+00	6.0816e+01	1.8494e-02	3.6980e-03
	标准差	6.3814e-01	4.5150e+00	1.2109e-02	5.2298e-03
$f_7(x)$	平均值	6.6395e+02	6.5731e+03	8.3030e-17	2.3749e-04
	标准差	1.1390e+02	2.1570e+02	3.3586e-04	1.0742e-16
$f_8(x)$	平均值	-5.6026e+03	-4.0937e+03	-2.8121e+03	-7.6534e+03
	标准差	5.9044e+02	3.3313e+02	2.4578e+02	2.2452e+02
$f_9(x)$	平均值	5.3486e+04	5.1005e+01	4.1125e+02	3.1658e+06
	标准差	3.9861e+03	3.8908e+01	4.5841e+02	1.0418e+06

图 2 表明, 在相同的实验环境下, 四种算法都有较好的收敛, 但 WLPSO 的优势更加明显, 其在九个标准测试函数上平均适应度曲线均位于最下方。由于引入了自适应惯性权重和莱维飞行机制, 在迭代初期, 粒子能够快速缩小搜索范围向最优解方向进行搜索, 迭代后期又有较好的局部搜索和跳出局部最优的能力。这表明 WLPSO 算法在搜索过程中较好地平衡了全局搜索和局部搜索的能力, 降低了陷入局

部最优的可能性，实现了收敛速度和收敛精度的提升。

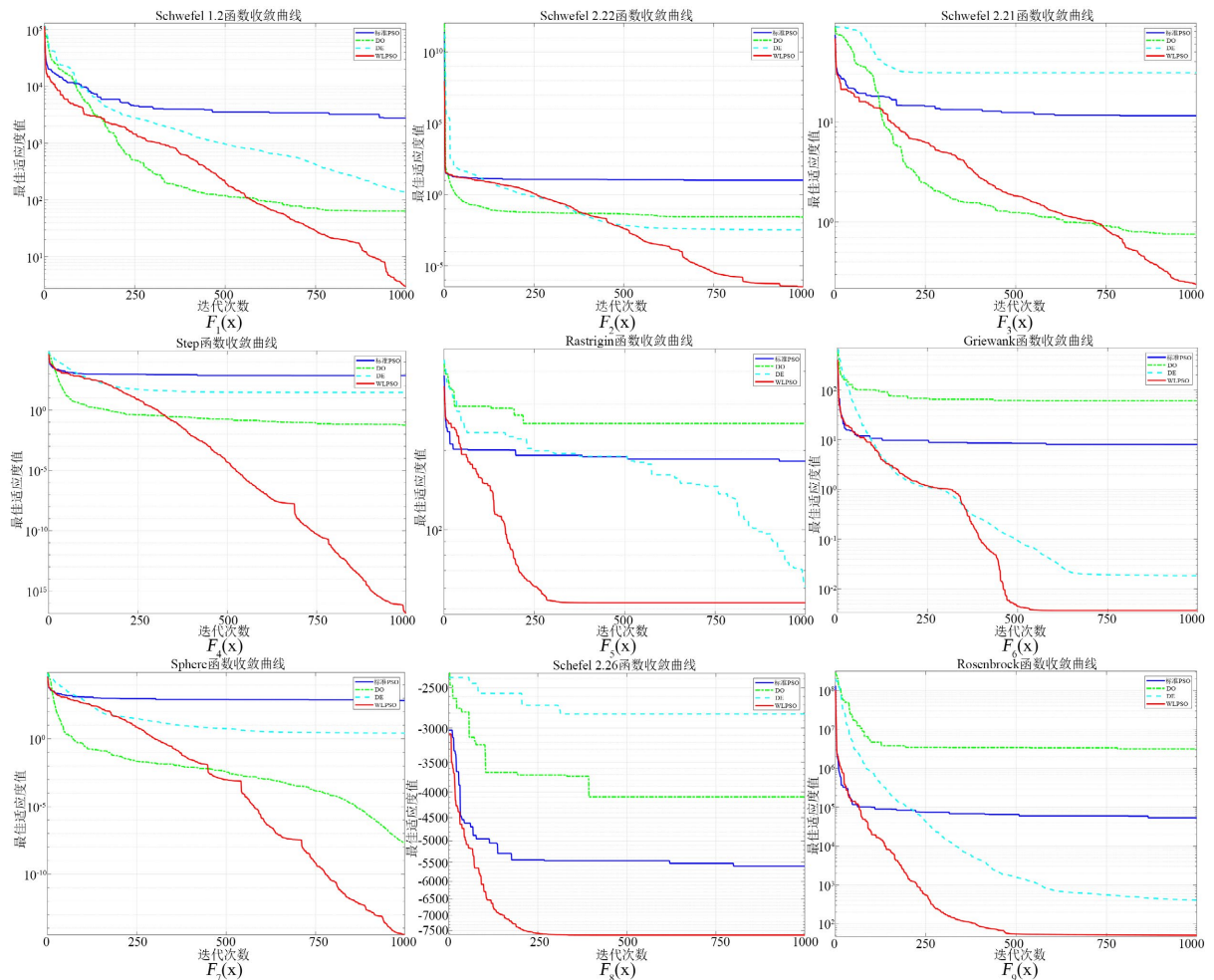


Figure 2. Comparison of four algorithms on the test function

图 2. 四种算法在测试函数上的对比图

5. 结束语

针对传统粒子群优化算法(PSO)在复杂优化问题中易陷入局部最优、收敛精度不足等固有缺陷，本文提出了一种融合自适应惯性权重与莱维飞行机制的改进粒子群算法(WLPSPSO)。该算法通过非线性自适应惯性权重调节策略，根据迭代过程动态平衡全局探索与局部开发能力，在搜索初期保持较大权重增强全局探索性，在后期逐步减小权重提高局部搜索精度；引入的莱维飞行机制通过其特有的短距离精细搜索与长距离随机跳跃相结合的特性，有效增强了算法跳出局部最优的能力。综上所述，该算法在寻优能力上得到了显著提升并具有高收敛精度和强鲁棒性。

参考文献

- [1] Eberhart, R. and Kennedy, J. (1995) A New Optimizer Using Particle Swarm Theory. MHS'95. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, 4-6 October 1995, 39-43. <https://doi.org/10.1109/mhs.1995.494215>
- [2] Clerc, M. and Kennedy, J. (2002) The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional

- Complex Space. *IEEE Transactions on Evolutionary Computation*, **6**, 58-73. <https://doi.org/10.1109/4235.985692>
- [3] Ozcan, E. and Mohan, C.K. (1999) Particle Swarm Optimization: Surfing the Waves. *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, Washington, 6-9 July 1999, 1939-1944. <https://doi.org/10.1109/cec.1999.785510>
- [4] 金欣磊, 马龙华, 吴铁军, 等. 基于随机过程的 PSO 收敛性分析[J]. 自动化学报, 2007(12): 1263-1268.
- [5] 刘国海, 万亚连, 沈跃, 等. 基于改进粒子群算法的高地隙无人喷雾机对不规则凸田块的全覆盖作业路径规划[J]. 华南农业大学学报, 2025, 46(3): 390-398.
- [6] 陈攀, 孙鉴, 吴佳伟, 等. 基于改进粒子群的云计算任务调度算法[J]. 科学技术与工程, 2025, 25(12): 5045-5057.
- [7] 刘虹伶, 时维国. 基于灰狼优化的改进粒子群算法求解环境经济调度问题[J]. 电机与控制应用, 2024, 51(11): 97-109.
- [8] 吴鹏, 叶宝林, 吴维敏, 等. 基于改进混沌粒子群算法的交通信号控制[J]. 计量学报, 2024, 45(12): 1876-1884.
- [9] 段艳, 潘峰. 融合改进莱维飞行的混沌粒子群算法[J]. 长江信息通信, 2025, 38(1): 67-69+72.
- [10] 张文成, 周穗华, 吴志东, 等. 改进策略的粒子群算法及其实验测试[J]. 计算机工程与设计, 2012, 33(10): 3945-3949.
- [11] Shi, Y. and Eberhart, R.C. (1998) Parameter Selection in Particle Swarm Optimization. In: *Lecture Notes in Computer Science*, Springer, 591-600. <https://doi.org/10.1007/bfb0040810>
- [12] Shi, Y. and Eberhart, R. (1998) A Modified Particle Swarm Optimizer. 1998 IEEE International Conference on Evolutionary Computation Proceedings. *IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, Anchorage, 4-9 May 1998, 69-73. <https://doi.org/10.1109/iccc.1998.699146>
- [13] Shi, Y. and Eberhart, R.C. (1999) Empirical Study of Particle Swarm Optimization. *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, Washington, 6-9 July 1999, 1945-1950. <https://doi.org/10.1109/cec.1999.785511>
- [14] Iacca, G., dos Santos Junior, V.C. and Veloso de Melo, V. (2021) An Improved Jaya Optimization Algorithm with Lévy Flight. *Expert Systems with Applications*, **165**, Article 113902. <https://doi.org/10.1016/j.eswa.2020.113902>
- [15] 张海阔, 孟秀云. 基于改进粒子群算法的无人机航迹规划[J]. 飞行力学, 2024, 42(2): 29-35.
- [16] Jensi, R. and Jiji, G.W. (2016) An Enhanced Particle Swarm Optimization with Levy Flight for Global Optimization. *Applied Soft Computing*, **43**, 248-261. <https://doi.org/10.1016/j.asoc.2016.02.018>
- [17] Clerc, M. and Kennedy, J. (2002) The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Transactions on Evolutionary Computation*, **6**, 58-73. <https://doi.org/10.1109/4235.985692>