

自适应多样性驱动的粒子群优化算法

张 俐

(江苏理工学院计算机工程学院, 江苏常州 213001)

摘 要: 针对粒子群优化算法在处理高维、多峰以及复杂优化问题时存在的种群多样性快速丧失与早熟收敛等固有缺陷, 本文提出一种自适应多样性驱动的粒子群优化算法 (Adaptive Diversity-Driven Particle Swarm Optimization, ADDPSO)。该算法从初始化策略、多样性量化、参数自适应、信息交互结构与扰动机制五个维度进行系统化重构, 以全面提升算法在复杂搜索空间中的全局探索能力、收敛精度与鲁棒性。首先, 针对初始种群分布不均问题, 设计基于 Logistic 混沌映射与均匀随机数的加权融合初始化策略, 通过 7:3 的比例兼顾序列遍历性与随机扰动性, 显著提升初始解在高维空间中的覆盖均匀性。随后, 引入基于平均欧氏距离的种群多样性量化指标, 并定义多样性比率作为演化过程的实时反馈信号, 为参数与策略的动态调整提供依据。在此基础上, 提出时间递减规则与多样性比率双驱动的参数自适应机制, 实现惯性权重与学习因子在探索与开发阶段间的平滑切换。进一步地, 为克服传统速度更新对单一全局最优解的过度依赖, 构建“个体认知-精英引导-种群分布”三层协同的速度更新架构。其中, 精英引导通过维护精英档案并采用基于适应度的概率化选择策略, 避免搜索方向趋同; 种群分布项则引入基于适应度偏差的全局协同机制, 通过均方根归一化与符号保留系数实现差异化引导。此外, 算法融合多样性感知的模拟退火机制与多策略自适应变异操作, 并引入双阈值接受准则, 在维持收敛趋势的同时主动注入多样性, 有效抑制早熟收敛。实验基于 CEC2017 测试集的 12 个高维复杂函数 (包括混合函数 F18、F19 与复合函数 F20~F29) 及齿轮系设计问题进行系统验证。结果表明, ADDPSO 在绝大多数函数上均取得最优或次优的平均值与标准差, 尤其在 F18、F20~F24、F27~F29 等高度复杂函数上, 其求解精度较主流 PSO 变体提升 1~4 个数量级, 且表现出更优的稳定性。在齿轮系设计问题中, ADDPSO 不仅稳定收敛至理论最优解, 显著优于对比算法, 充分验证其在工程优化中的可靠性与一致性。总之, ADDPSO 通过多层次、多机制的协同设计, 系统性地解决 PSO 在高维复杂优化中的多样性衰减与早熟收敛问题, 展现出优异的综合性能与实际应用潜力。

关键词: 粒子群优化算法; 种群多样性; 高维复杂; 参数自适应; 早熟收敛; 全局优化

基金项目: 符号计算与知识工程教育部重点实验室课题 (No.93K172023K08)

中图分类号: TP301.6

文献标识码: A

文章编号: 0372-2112(2025)12-4671-15

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

DOI: 10.12263/DZXB.20250823

Adaptive Diversity-Driven Particle Swarm Optimization

ZHANG Li

(College of Computer Engineering, Jiangsu University of Technology, Changzhou, Jiangsu 213001, China)

Abstract: To address the inherent limitations of particle swarm optimization (PSO) in handling high-dimensional, multi-modal, and complex optimization problems, such as rapid loss of population diversity and premature convergence, an adaptive diversity-driven particle swarm optimization (ADDPSO) algorithm is proposed. This algorithm is systematically reconstructed across five dimensions—initialization strategy, diversity quantification, parameter adaptation, information exchange structure, and perturbation mechanism—to comprehensively enhance its global exploration capability, convergence accuracy, and robustness within complex search spaces. First, to address uneven initial population distribution, a weighted fusion initialization strategy combining Logistic chaotic mapping and uniform random numbers is designed. The ratio at 7:3 balances sequence traversal and random perturbation, significantly improving the uniformity of initial solution coverage in high-dimensional spaces. Second, a population diversity metric based on average Euclidean distance is introduced, with the diversity ratio defined as a real-time feedback signal for evolutionary adjustments, which enables dynamic parameter and strategy tuning. Built upon this, a dual-driven parameter adaptation mechanism combining time-decreasing rules and diversi-

ty ratio is proposed, enabling smooth transitions between inertia weight and learning rate during exploration and exploitation phases. Furthermore, to overcome the excessive reliance of traditional velocity updates on a single global optimum, a three-tiered collaborative velocity update architecture “individual cognition-elite guidance-population distribution” is constructed. Elite guidance prevents search direction convergence by maintaining an elite archive and employing probability-based selection strategies based on fitness. While the population distribution component introduces a global coordination mechanism based on fitness deviation, achieving differentiated guidance through root-mean-square normalisation and sign-preserving coefficients. Additionally, the algorithm integrates diversity-aware simulated annealing with multi-strategy adaptive mutation operations and incorporates a dual-threshold acceptance criterion. This approach actively injects diversity while maintaining convergence trends, effectively suppressing premature convergence. Experimental validation was conducted on the CEC2017 test set comprising 12 high-dimensional complex functions (including mixed functions F18~F19 and composite functions F20~F29) and a gear system design problem. Results demonstrate that ADDPSO achieves optimal or near-optimal mean and standard deviation values on most functions. Particularly for highly complex functions such as F18, F20~F24, and F27~F29, their solution accuracy surpasses mainstream PSO variants by 1 to 4 orders of magnitude while exhibiting superior stability. In gear system design problems, ADDPSO not only converged stably to theoretically optimal solutions but also significantly outperformed comparison algorithms, fully validating its reliability and consistency in engineering optimization. In summary, through multi-level, multi-mechanism collaborative design, ADDPSO systematically addresses PSO’s diversity decay and premature convergence issues in high-dimensional complex optimization, demonstrating outstanding comprehensive performance and practical application potential.

Key words: particle swarm optimization algorithm; population diversity; high-dimensional complexity; parameter adaptation; premature convergence; global optimization

Foundation Item(s): Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education (No.93K172023K08)

1 引言

粒子群优化算法^[1] (Particle Swarm Optimization, PSO)由 Kennedy 和 Eberhart 于 1995 年提出,其通过模拟鸟群觅食等社会行为,利用个体与群体历史经验指导搜索过程.该算法结构简单、参数较少、收敛性能良好,已被广泛应用于特征选择^[2]、工程优化^[3]与路径规划^[4]等领域.然而,在处理高维多峰、非线性及复杂约束优化问题时,PSO 仍存在早熟收敛和易陷入局部最优等固有缺陷,严重限制其进一步应用与发展.

PSO 性能受限的根本原因在于种群多样性迅速下降,具体表现有:①初始种群分布不均,难以充分覆盖搜索空间;②参数设置缺乏动态适应性,无法响应不同搜索阶段的需求;③信息交互过度依赖全局历史最优解,导致搜索方向趋同;④缺乏有效的种群结构感知与多样性维持机制.

为提升种群多样性,研究者提出多种改进策略,主要包括以下四类:

(1)基于种群初始化的多样性增强方法.例如,Gao 等人^[5]提出结合混沌映射与反向学习策略,提高全局收敛性能.Gao 等人^[6]提出一种名为 ISPSO 的算法,该算法采用信息增益比的初始化机制,优化初始粒子的分布状态,提升粒子群优化算法的性能.

(2)基于种群拓扑结构的多样性增强方法.Sun 等人^[7]提出的 ADN-RSN-PSO 算法采用随机选择学习机

制提升多样性,并利用邻域结构加快收敛速度.Wang 等人^[8]提出的 RLLPSO 算法通过层级粒子群结构增强多样性,并结合强化学习动态调整层级与竞争策略.

(3)基于更新策略的多样性增强方法.Liang 等人^[9]提出综合学习粒子群优化算法 (CLPSO),使粒子能够从其他粒子的历史最优位置学习,从而扩展搜索空间并增强多样性.Li 等人^[10]将多样性度量用于指导进化过程,以缓解早熟收敛.

(4)混合策略的多样性增强方法.Liu 等人^[11]提出多样性粒子群优化算法 (DPSO),显著提升综合性能.Fu^[12]提出的 AAPSO 算法通过三级协同机制,解决 PSO 渐近效率低下的问题,实现种群多样性的动态平衡与搜索动量的持续维持.

然而,上述方法仍存在以下三方面局限:①改进策略多为单一机制,缺乏系统性与层次性集成;②参数调整多采用线性或固定规则,难以适应优化过程的动态特性;③对多样性度量、初始化质量及结构引导等基础环节关注不足,导致算法在应对“初始分布失衡”“参数固化”“搜索趋同”与“早熟收敛”等问题时效果有限.为了克服上述不足,本文提出一种自适应多样性驱动的粒子群优化算法 (Adaptive Diversity-Driven PSO, ADDPSO),从多样性度量、参数自适应调节与结构化信息交互等多个层面系统重构 PSO 的搜索机制.

本文的主要贡献包括:

(1)设计基于平均欧氏距离的种群多样性度量方

法,刻画出高维连续空间中粒子的分散程度.

(2)建立时间递减规则与多样性反馈相结合的双驱动参数自适应机制,实现惯性权重 w 与学习因子 (c_1 和 c_2) 的动态调节,以精准响应不同搜索阶段的需求.

(3)构建“个体认知-精英引导-种群分布”三层信息融合的速度更新机制,降低对全局最优解的过度依赖,提升搜索方向的多样性与引导有效性.

(4)引入多样性感知的模拟退火机制与多策略自适应变异算子,结合双阈值接受准则,在维持收敛性的同时有效抑制早熟收敛.

(5)提出速度边界与变异强度的动态调整策略,进一步提升算法在复杂环境下的鲁棒性与适应性.

2 粒子群优化算法

在 PSO 中,每个个体被称为“粒子”,代表搜索空间中的一个候选解.算法通过迭代更新粒子的位置与速度,引导种群逐步向全局最优解收敛.粒子 i 位置的更新与四个向量相关,分别是其历史最优位置 \mathbf{Pbest}_i , 全局最优位置 \mathbf{Gbest} , 当前位置 \mathbf{X}_i 以及当前速度 \mathbf{V}_i . 其中,设种群中包含 N 个个体,每个个体对应 dim 维的解向量,即个体 i 的解向量可表示为 $\mathbf{X}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,d})$, 其中 $x_{i,d}$ 为个体 i 在第 d 个维上的取值 ($i=1, 2, \dots, N; d=1, 2, \dots, \text{dim}$). 粒子 i 的位置更新公式如下:

$$\mathbf{X}_i(t+1) = \mathbf{X}_i(t) + \mathbf{V}_i(t+1) \quad (1)$$

$$\begin{aligned} \mathbf{V}_i(t+1) = & w \cdot \mathbf{V}_i(t) + c_1 \cdot r_1 \cdot (\mathbf{Pbest}_i - \mathbf{X}_i(t)) \\ & + c_2 \cdot r_2 \cdot (\mathbf{Gbest} - \mathbf{X}_i(t)) \end{aligned} \quad (2)$$

其中, w 为固定惯性权重, c_1 为个体学习因子, c_2 是社会学习因子, t 为当前迭代次数, $r_1 \in (0, 1)$, $r_2 \in (0, 1)$.

通常,粒子 i 的初始位置通常按以下方式初始化:

$$x_{i,d} = \text{lb}_d + r_3 \cdot (\text{ub}_d - \text{lb}_d) \quad (3)$$

其中, ub_d 和 lb_d 分别为第 d 维的上下边界, $r_3 \in (0, 1)$ 为随机数.

3 自适应多样性驱动粒子群优化算法

如第 1 节所述,研究基于种群多样性的 PSO 改进算法具有理论价值与应用前景^[13]. 传统 PSO 在处理高维多峰及复杂函数优化问题时,易出现早熟收敛与搜索效率低下的现象,其根源主要在于初始分布不均、多样性度量缺失、参数调节僵化以及信息交互机制单一. 本文围绕多样性量化、参数自适应与信息交互结构三个层面构建改进框架,重点针对以下两个关键问题提出解决方案.

首先,针对传统 PSO 中因固定参数与单一策略导致的搜索行为僵化问题,本文提出一种双驱动参数自适应机制与多策略变异框架. 以多样性比率 (divRatio)

作为反馈指标,实时调节惯性权重与学习因子,并在识别早熟风险时增强全局探索能力. 同时引入融合多种变异操作的自适应扰动策略,结合双阈值接受准则,在维持收敛趋势的同时主动注入多样性,缓解策略同质化带来的性能退化.

然后,针对信息交互机制缺乏结构化引导的问题,构建“个体认知-精英引导-种群分布”三层协同的速度更新架构. 通过精英解集和多精英引导机制降低对单一全局最优解的依赖,并设计基于适应度偏差的全局交互项,使不同粒子对搜索方向产生差异化贡献. 同时动态调整速度边界,以适应不同演化阶段对搜索步长的需求.

总之,ADDPSO 算法通过多样性感知机制、多层次信息整合与多策略变异操作,从结构上提升算法的全局收敛能力与鲁棒性.

3.1 符号说明

本节对算法中涉及的核心符号进行详细分类与说明. 符号主要涵盖种群参数、粒子状态、算法控制变量、多样性度量及扰动策略等方面.

3.1.1 种群基本参数

N 表示粒子群中的个体数量.

dim 表示优化问题的维度.

t 为当前迭代次数, t_{\max} 为最大迭代次数.

\mathbf{X}_i 为粒子 i 的当前位置向量,表示为 $\mathbf{X}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,d})$, 其中 $i=1, 2, \dots, N$.

\mathbf{V}_i 为粒子 i 的当前速度向量.

\mathbf{Pbest}_i 为粒子 i 历史最优位置.

\mathbf{Gbest} 为全局最优位置.

w 为惯性权重, c_1 为个体学习因子, c_2 为社会学习因子.

r_1, r_2, \dots 为均匀分布在区间 $[0, 1]$ 内的随机数.

ub 和 lb 分别表示变量上下边界.

lx_n 为 Logistic 混沌序列的第 n 项.

$f(\mathbf{X})$ 为解 \mathbf{X} 的适应度函数值.

3.1.2 多样性度量指标

$\text{dis}_{i,k}$ 为粒子 i 和粒子 k 之间的欧氏距离.

Div 为基于平均欧氏距离的种群多样性指标.

divRatio 为多样性比率.

3.1.3 其他辅助符号

M 为精英档案容量.

$\mathbf{E}(t)$ 为第 t 代精英档案集合.

f_{rms} 为种群适应度的均方根值.

TT 为模拟退火的当前温度.

CR 为降温系数.

Δf 为适应度变化量.

$\mathcal{N}(0, 1)$ 为标准正态分布.

$C(0, 1)$ 为柯西分布.

3.2 Logistic混沌与均匀分布混合的种群初始化

式(3)采用传统随机初始化方式时,在初始化阶段常难以充分覆盖整个搜索空间,不仅导致种群多样性不足,还会进一步制约算法的全局搜索能力.针对这一问题,为提升初始解的质量与分布均匀性,本文引入Logistic混沌与均匀随机数映射.该映射兼具优良的遍历性与伪随机性,能够生成分布范围广且结构复杂度高的初始序列.

3.2.1 Logistic混沌序列生成

Logistic映射是一种典型的混沌系统,其迭代公式如下:

$$lx_{n+1} = 4 \cdot lx_n \cdot (1 - lx_n) \quad (4)$$

当初始值 lx_0 在非周期点时,系统进入混沌状态,生成的序列在区间内均匀分布且对初值敏感,具备优良的遍历特性.

3.2.2 初始位置计算

为进一步增强初始种群的全局覆盖能力,避免纯混沌序列因系统动力学特性导致的局部聚集现象,本文采用混沌序列与均匀随机数加权融合的方法构建初始种群.第 i 个粒子个体在第 d 维的位置按下式计算:

$$x_{i,d} = lb_d + (ub_d - lb_d) \cdot (0.7 \cdot lx_d + 0.3 \cdot r_3) \quad (5)$$

该策略以7:3的比例融合混沌序列与随机扰动,在保证遍历性的同时增强初始解的随机性与空间覆盖能力.

3.3 基于平均欧氏距离的种群多样性度量方法

在进化算法中,种群多样性是维持算法全局搜索能力的关键指标.一旦多样性衰退,种群个体将趋于同质化,不仅易引发早熟收敛问题,还会限制算法在复杂多峰优化问题中的求解性能.当前,种群多样性测量体系可划分为两类核心方法.

(1)以曼哈顿距离^[11]为代表的连续空间评估方法.然而,该类方法的传统实现方式效率较低,难以满足实时性评估需求,进而严重影响算法的运行效率与优化性能.

(2)以互信息^[14]为代表的离散编码评估方法.该方法在离散优化问题中具有良好适用性,但在连续空间优化场景下,存在无法有效度量实值向量空间距离、对连续变量编码敏感性不足等缺陷.

针对现有种群多样性测量方法的上述不足,本文提出一种基于平均欧氏距离的种群多样性计算连续空间评估方法.

3.3.1 欧氏距离的定义

对于种群中任意两个不同个体 i 与 k (其中 $i \neq k$)之间的欧氏距离 dis_{ik} 定义为

$$dis_{ik} = \sqrt{\sum_{d=1}^{\dim} (x_{i,d} - x_{k,d})^2} \quad (6)$$

其中, $x_{i,d} - x_{k,d}$ 表示个体 i 与 k 在第 d 维度上的差异.该距离度量高维空间中两点的直线距离,平方和开根号操作可消除维度差异的符号影响,并强化较大偏差的权重,从而更真实反映个体间的整体差异.

3.3.2 平均欧氏距离计算

由于种群多样性可通过个体间距离的统计均值进行表征.基于此,本文定义的多样性指标Div为所有无序个体对之间欧氏距离的算术平均值,具体形式如下:

$$Div = \frac{2}{N(N-1)} \sum_{1 \leq i < k \leq N} dis_{ik} \quad (7)$$

其中, $1 \leq i < k \leq N$ 表示对所有无重复的个体对的欧氏距离求和,系数 $\frac{2}{N(N-1)}$ 为归一化因子,确保结果为算术平均值.当Div值越大时,种群内个体的分散程度越高,多样性越强;当Div值趋近于0时,个体高度同质化,多样性丧失.

3.4 基于多样性的PSO参数动态调整策略

传统PSO算法采用固定惯性权重 w 及学习因子 c_1 和 c_2 ,难以适应搜索过程的动态演化,易导致探索与开发失衡.尤其在多峰复杂问题中,静态参数难以有效协调全局搜索与局部收敛,易引发早熟或收敛停滞.为此,本文提出一种迭代进程与种群状态双驱动的自适应参数调控机制,结合非线性时间演化与多样性反馈,实现 w 、 c_1 和 c_2 的协同动态调整.

3.4.1 惯性权重与学习因子的非线性演化

为更合理匹配搜索阶段特性,设计如下非线性参数调整策略.惯性权重 w 与个体学习因子 c_1 采用加速递减模式:

$$w = w_{end} + (w_{start} - w_{end}) \cdot \left(1 - \frac{t}{t_{max}}\right)^2 \quad (8)$$

$$c_1 = c_{1,end} + (c_{1,start} - c_{1,end}) \cdot \left(1 - \frac{t}{t_{max}}\right)^2 \quad (9)$$

社会学习因子 c_2 采用递增模式:

$$c_2 = c_{2,start} + (c_{2,end} - c_{2,start}) \cdot \left(\frac{t}{t_{max}}\right)^2 \quad (10)$$

其中, $w_{min} = 0.2$, $w_{max} = 0.95$, $c_{1,min} = c_{2,min} = 0.2$, $c_{1,max} = c_{2,max} = 2.8$, t_{max} 为最大迭代次数.

上述设计在搜索初期赋予较大的 w 与 c_1 ,增强个体探索能力;随迭代推进, w 与 c_1 快速衰减, c_2 逐步上升,促进社会协同与局部开发,相较于线性策略更符合搜索阶段的动态需求.

3.4.2 多样性反馈的参数动态调节机制

为进一步提升参数调整对种群状态的响应能力,

引入多样性比率作为反馈信号:

$$\text{divRatio} = \frac{\text{Div}_{\text{cur}}}{\text{Div}_{\text{ini}}} \quad (11)$$

其中, Div_{cur} 为当前迭代种群多样性, Div_{ini} 为初始种群多样性. 当 $\text{divRatio} \leq 0.2$ 时候, 判定种群趋于早熟, 触发探索增强机制, 参数动态调整为

$$w = 1.2 \cdot w, \quad c_1 = 1.1 \cdot c_1, \quad c_2 = 0.9 \cdot c_2 \quad (12)$$

该策略通过增大惯性权重 w 与个体认知 c_1 能力, 削弱社会引导 c_2 作用, 促使粒子跳出局部吸引域, 恢复全局搜索活力.

3.5 基于多样性的分层引导速度更新机制

PSO 的速度更新策略仅依赖个体最优与全局最优, 缺乏对种群结构信息的感知, 易导致搜索方向单一、多样性丧失. 为此, 本文提出一种多尺度分层引导机制, 融合个体认知、精英引导与种群分布信息, 构建协同驱动的更新框架.

3.5.1 速度更新公式的三层结构

改进的速度更新方程包含三个层次:

$$\begin{aligned} V_i(t+1) = & w \cdot V_i(t) + c_1 \cdot r_1 \cdot (\mathbf{Pbest}_i - X_i(t)) \\ & + c_2 \cdot r_2 \cdot (\mathbf{Gbest} - X_i(t)) \\ & + c_3 \cdot r_3 \cdot (\mathbf{E}_k^*(t) - X_i(t)) + 0.5 \cdot \text{sum}_{\text{hso}} \end{aligned} \quad (13)$$

其中, 第一层为基本项, 维持个体记忆与社会学习; 第二层引入精英存档中的优质解 $\mathbf{E}_k^*(t)$ 以概率化方法进行定向引导; 第三层加入基于种群分布的全局交互项 sum_{hso} , 增强协同探索能力.

3.5.2 精英档案构建与概率化引导策略

从精英存档中选择优质解辅助引导, 避免仅依赖单一全局最优导致的局部收敛. 设第 t 代种群为 $\mathbf{X}(t) = \{X_1(t), X_2(t), \dots, X_N(t)\}$, 对应适应度值为 $f(\mathbf{X}(t)) = \{f(X_1(t)), f(X_2(t)), \dots, f(X_N(t))\}$. 精英档案 $\mathbf{E}(t) = \{E_1(t), E_2(t), \dots, E_M(t)\}$ (M 为精英档案容量), 对应适应度值为 $f(\mathbf{E}(t)) = \{f(E_1(t)), f(E_2(t)), \dots, f(E_M(t))\}$, 具体步骤如下.

精英存档机制通过候选解集的构建与筛选两个核心步骤, 实现对优质解的动态维护. 具体而言, 第一步先将当前种群与历史精英档案进行合并, 形成规模扩大的候选解集, 其数学表达如下:

$$\mathbf{T}(t) = \mathbf{X}(t) \cup \mathbf{E}(t) \quad (14)$$

其中, $\mathbf{T}(t)$ 为合并后的候选解集, 包含 $N+M$ 个解. 随后, 对候选解集 $\mathbf{T}(t)$ 按适应度值升序排序, 并在去重基础上选取最优的 M 个解组成新的精英档案, 具体公式如下:

$$\mathbf{E}(t+1) = \text{argmin}_{E' \in M} \{f(t) | t \in E'\} \quad (15)$$

其中, $E' \subseteq \mathbf{T}'(t)$, $\mathbf{T}'(t)$ 为去重后的候选解集, $\mathbf{E}(t+1)$ 为

更新后的精英档案, $M=5$.

然后, 为避免单一最优解主导引导过程, 采用基于适应度的概率化选择策略. 第 i 个精英解的选择权重为

$$\omega_i = \frac{\max(f(\mathbf{E}(t))) - f(\mathbf{E}_i(t))}{\sum_{i=1}^M (\max(f(\mathbf{E}(t))) - f(\mathbf{E}_i(t)))} \quad (16)$$

其中, $\max(f(\mathbf{E}(t)))$ 为精英档案中的最差适应度值. 第 i 个精英适度值归一化后的选择概率如下:

$$p_i = \frac{\omega_i}{\sum_{i=1}^M \omega_i} \quad (17)$$

其中, p_i 为第 i 个精英解被选中的概率, 满足 $\sum_{i=1}^M p_i = 1$ 且 $p_i \geq 0$. 通过轮盘赌方式选择引导个体 \mathbf{E}_i^* , 其被选中的概率为 p_i , 在强化优质解引导作用的同时保留搜索多样性.

3.5.3 基于适应度值的全局引导项

为增强种群层面的信息交互能力, 本文引入一种基于适应度偏差的全局引导机制. 该机制通过对适应度进行均方根归一化处理, 并采用符号保留的系数 (cc_i) 分配策略, 实现对粒子运动方向的精细化引导, 进而有效提升搜索过程的协同性与稳定性.

首先, 计算种群的均方根适应度 f_{rms} , 用于表征当前种群的整体性能水平, 具体公式如下:

$$f_{\text{rms}} = \sqrt{\frac{1}{N} \sum_{i=1}^N f(\mathbf{X}_i(t))} \quad (18)$$

随后, 定义各粒子相对于种群整体水平的偏差量 d_i :

$$d_i = f_{\text{rms}} - f(\mathbf{X}_i(t)) \quad (19)$$

并据此构建归一化引导系数 cc_i :

$$cc_i = \frac{\text{sign}(d_i) \cdot |d_i|}{\sum_{i=1}^N |d_i|} \quad (20)$$

其中, $\text{sign}(d_i)$ 为符号函数, 确保系数保留偏差的方向信息. 当 $f_{\text{rms}} > f(\mathbf{X}_i(t))$ 时, $cc_i > 0$, 表示第 i 个粒子优于种群平均水平, 其位置信息对其他个体具有正向引导作用; 反之则抑制其影响. 正偏差个体 (优于平均水平) 获得正向引导权重, 负偏差个体则被抑制. 全局信息项定义为

$$\text{sum}_{\text{hso}} = \sum_{i=1}^N (r_5 \cdot cc_i \cdot (\mathbf{X}_i - \mathbf{X}_j)) \quad (21)$$

其中, \mathbf{X}_i 为第 i 个粒子的位置, \mathbf{X}_j 为当前 j 个粒子位置. $r_5 \in (0, 1)$. 该机制融合种群结构信息, 实现差异化引

导,提升搜索协同性与稳定性.

3.5.4 速度边界多样性自适应机制

传统固定速度边界难以适应不同搜索阶段的需求.为此,设计基于多样性比率的动态边界:

$$dv_{\max,d} = v_{\max,d} \cdot (1 + 0.5 \cdot (1 - \text{divRatio})) \quad (22)$$

$$dv_{\min,d} = -dv_{\max,d} \quad (23)$$

其中, $v_{\max,d}$ 表示粒子在第 d 维上的最大速度值. divRatio 为多样性比率.当 divRatio 较高时,扩大速度边界以增强探索能力;当多样性降低时,收缩边界以提高局部开发精度.该机制显著提升算法在复杂优化问题中的鲁棒性.

3.6 自适应多样性模拟退火机制

为缓解早熟收敛,传统模拟退火通过接受劣解维持多样性,但其接受概率仅依赖固定降温机制,缺乏对种群多样性状态的感知能力.本文将多样性比率引入退火过程,构建自适应接受准则:

$$P(\Delta f) = \begin{cases} \exp\left(-\frac{\Delta f}{T \cdot (1 + 0.5(1 - \text{divRatio}))}\right), & \text{otherwise} \\ 1, & \text{if } f_{\text{new}} < f_{\text{old}} \end{cases} \quad (24)$$

其中, T 为当前温度, $\Delta f = f_{\text{new}} - f_{\text{old}}$. 当 divRatio 较低时,调节因子增大,退火温度等效升高,劣解接受概率提升,有效激发全局探索;反之则侧重局部开发.该机制实现探索与开发的自适应平衡.

3.7 多策略自适应变异机制

由第3.5节分析可知,自适应模拟退火机制仅能被被动接受新解,无法主动为种群注入多样性,因此需通过变异操作弥补这一不足.基于此,本文设计一种种群多样性的多策略自适应变异机制,该机制结合阶段感知与多样性反馈双重要素,旨在有效提升算法的全局搜索能力.

3.7.1 变异强度的自适应调整

为实现变异力度的动态调节,引入式(7)设计自适应变异强度控制机制:

$$\text{mutstrength} = \begin{cases} 0.8 - 0.5 \cdot \frac{t}{t_{\max}}, & \text{if } \text{curDiv} < 0.3 \cdot \text{iniDiv} \\ 0.4 - 0.3 \cdot \frac{t}{t_{\max}}, & \text{otherwise} \end{cases} \quad (25)$$

在种群多样性较低阶段采用较大变异强度,以增强逃逸能力;其他阶段逐步减弱,转向精细开发.

3.7.2 多策略变异算子的动态选择

为进一步提升搜索多样性与鲁棒性,设计三种特性互补的变异算子,并在每次变异操作时以等概率随

机选取一种,避免策略偏好,增强机制的自适应性.

(1)高斯变异(局部精细搜索)

$$\mathbf{X}_{\text{mut}} = \mathbf{X} + \text{mutstrength} \cdot \mathcal{N}(0, 1) \quad (26)$$

其中, $\mathcal{N}(0, 1)$ 表示标准正态分布采样,生成的扰动集中在当前解邻域,适用于局部精细搜索.

(2)柯西变异(全局探索)

$$\mathbf{X}_{\text{mut}} = \mathbf{X} + \text{mutstrength} \cdot \mathcal{C}(0, 1) \quad (27)$$

其中, $\mathcal{C}(0, 1)$ 为柯西分布,可产生大范围跳跃式扰动,有助于跳出局部吸引域,显著增强全局探索能力.

(3)差分进化变异(利用种群结构信息)

$$\mathbf{X}_{\text{mut}} = \mathbf{X}_{r_1} + 0.5 \cdot (\mathbf{X}_{r_2} - \mathbf{X}_{r_3}) \quad (28)$$

其中, \mathbf{X}_{r_1} 、 \mathbf{X}_{r_2} 和 \mathbf{X}_{r_3} 为从种群中随机选取的三个互异个体.该策略利用种群内部差异信息生成新解,具有方向引导性,有利于促进个体间协同进化与信息共享.

3.7.3 变异接受的双阈值机制

为平衡收敛性与多样性,设计双阈值接受规则.当变异解 \mathbf{X}_{mut} 的目标函数值 $f(\mathbf{X}_{\text{mut}})$ 优于当前解 $f(\mathbf{X}_i)$ (即 $f(\mathbf{X}_{\text{mut}}) < f(\mathbf{X}_i)$) 时,直接予以接受,确保进化过程的收敛性;当变异解略差但偏差较小 (即 $f(\mathbf{X}_i) < f(\mathbf{X}_{\text{mut}}) < 1.2 \cdot f(\mathbf{X}_i)$) 时,则以 10% 的概率予以接受.该机制在保证收敛趋势的同时,保留潜在优质解,有效防止过早收敛.

3.8 算法步骤

基于上述混沌初始化、多样性度量、动态参数调整、精英引导、模拟退火与多策略变异等机制,ADDPSO 算法的具体步骤如算法 1 所示.

3.9 算法复杂性分析

算法复杂度是评估群体智能优化算法计算效率的关键指标,直接决定算法在高维优化问题中的实际可用性^[15]. ADDPSO 算法在传统 PSO 框架基础上,引入多样性驱动机制、多层引导结构及多策略变异操作,虽在一定程度上增加计算开销,但通过模块化设计与层次化协同,整体复杂度仍处于可控范围.本节围绕各核心操作的计算成本展开详细分析,从而为算法在实际应用中的效率评估提供理论依据. ADDPSO 的计算复杂度主要来源于以下几个方面.首先,种群多样性计算基于平均欧氏距离度量,需遍历所有粒子对并计算其高维空间中的距离,其复杂度为 $O(N^2 \cdot \text{dim})$;其次,精英档案的维护涉及候选解集的合并、排序与筛选,其最坏复杂度为 $O((N+M) \log(N+M))$,其中 M 为精英档案容量;此外,速度与位置更新操作需基于“个体-精英-种群”三层引导机制更新每个粒子的状态,复杂度为 $O(N \cdot \text{dim})$;而模拟退火与多策略自适应变异部分粒子

算法 1 ADDPSO 算法

输入: 种群规模 N 、维度 dim 、变量上下边界 lb/ub 、精英存档容量 $M=5$ 、多样性阈值 $\text{diversity_threshold}=0.2$ 、模拟退火初始温度

$\text{TT}_{\text{init}}=3\ 000$ 、降温系数 $\text{CR}=0.995$

输出: 最优解 \mathbf{Gbest}

1. 根据式(5), 初始化种群 \mathbf{X} . 计算当前粒子群的适应度 $f(\mathbf{X})$ 并初始化种群速度 \mathbf{V}
2. 根据式(7), 计算初始种群多样性 Div_{ini}
3. 根据式(8)、式(9)、式(10), 分别计算参数 w 、 c_1 和 c_2 的值
4. 根据式(7)、式(11), 分别计算当前种群多样性 Div_{cur} 和多样性比率 divRatio
5. if $\text{divRatio} \leq 0.2$
 根据式(12), 计算参数 w 、 c_1 和 c_2 的值
6. 根据式(20), 全局种群系数 cc
7. 根据式(16)、式(17), 依据轮盘赌进行精英选择精英解
8. 根据式(21), 计算全局信息项
9. 根据式(13), 使用分层引导机制更新速度
10. 根据式(22)、式(23), 进行动态速度边界检查
11. 根据式(1), 进行位置更新
12. 根据式(25), 进行模拟退火选择
13. if $\text{rand} \leq 0.9$ THEN
 随机生成一个 $[0,1]$ 区间内的均匀随机数 r , 依据 r 的取值从以下 3 种变异策略中选择一种生成变异位置 X_{mut}
 若 $0 < r < 1/3$, 根据式(26), 进行高斯变异选择
 若 $1/3 < r < 2/3$, 根据式(27), 进行柯西变异选择
 若 $2/3 < r < 1$, 根据式(28), 进行差分进化变异
 使用双阈值接受机制决定是否接受变异解
14. 根据式(7)、式(15), 分别更新种群多样性和精英档案
 IF $\text{diversity_ratio} < \text{diversity_threshold}$
 $\text{TT} \leftarrow \text{TT} * (\text{CR} + 0.001)$
 ELSE
 $\text{TT} \leftarrow \text{TT} * \text{CR}$
 END IF

需执行接受判据与变异操作, 该过程复杂度为 $O(K \cdot N \cdot \text{dim})$, 其中 K 为变异策略数量; 参数自适应调整与轮盘赌选择操作复杂度较低, 仅为 $O(M+N)$. 因此, ADDPSO 算法的总计算复杂度可表示如下:

$$O\left(t_{\text{max}} \cdot \left[N^2 \cdot \text{dim} + (N+M) \log(N+M) + N \cdot \text{dim} \cdot (1+k) \right] \right) \quad (29)$$

尽管 ADDPSO 算法在传统 PSO 基础上引入多样性驱动与多层引导机制, 其复杂度仍与典型的 PSO 算法处于同一量级.

4 实验验证

4.1 参数测试

为验证自适应多样性驱动粒子群优化算法(ADDPSO)在高维复杂函数优化中的性能, 本文基于 CEC2017 测试

集(表1), 选取 2 个混合函数(F18 和 F19)与 10 个复合函数(F20~F29)进行测试. 混合函数通过组合多个基础函数片段构成, 用于评估算法在异质搜索空间中的泛化与适应能力; 复合函数具有强非线性、不可分性与旋转不变性, 可模拟高度复杂的实际优化场景.

表 1 CEC2017 部分测试套件

函数	编号	Functions
混合函数	F18	Hybrid Function 6 ($N=5$)
	F19	Hybrid Function 6 ($N=6$)
复合函数	F20	Composition Function 1 ($N=3$)
	F21	Composition Function 2 ($N=3$)
	F22	Composition Function 3 ($N=4$)
	F23	Composition Function 4 ($N=4$)
	F24	Composition Function 5 ($N=5$)
	F25	Composition Function 6 ($N=5$)
	F26	Composition Function 7 ($N=6$)
	F27	Composition Function 8 ($N=6$)
	F28	Composition Function 9 ($N=3$)
	F29	Composition Function 10 ($N=3$)

4.2 对比算法设置

实验在维度 $\text{dim}=100$ 的条件下进行, 将 ADDPSO 与 PSO 以及六种不同 PSO 变体算法, 进行综合性能对比. 具体的对比算法来源和参数设置如表 2 所示.

实验设置方面, 所有算法均采用统一参数, 种群规模为 50, 最大迭代次数为 1 000. 每个算法在每项测试函数上独立运行 30 次, 结果采用 Friedman 检验与 Wilcoxon 秩和检验(显著性水平 $\alpha=0.05$)进行统计显著性分析. 所有实验均在 Windows 10 系统下通过 Python 实现.

4.3 初始化种群参数设置

在 ADDPSO 算法中, 混沌序列与均匀随机数的融合比例是决定初始种群分布均匀性与空间遍历性的关键参数, 直接影响种群多样性的初始水平与全局搜索效率. 若 Logistic 混沌序列权重过高, 初始解易呈现局部聚集特征, 削弱种群多样性. 而均匀随机数占比过大, 则会降低解空间覆盖的系统性, 延缓算法收敛进程. 为充分发挥 Logistic 混沌序列的遍历性优势, 同时融合均匀随机数的扰动特性, 本研究通过系统实验确定最优融合比例. 实验基于 CEC2017 测试集中的混合函数(F18 和 F19)与复合函数(F20~F29), 对比分析 5 组融合比例(5:5、6:4、7:3、8:2、9:1)的优化性能, 并以平均值(mean)与标准差(std)作为评价指标, 具体结果如表 3 所示.

表 2 对比算法参数设置

算法	参数设置
PSO ^[1]	$w=0.9, c_1=2, c_2=2$
CLPSO ^[9]	$w \in [0.4, 0.9], Pc \in [0.05, 0.5]$
EAPSO ^[16]	$w \in [0, 1], \lambda_1 \in [0, 1], \lambda_2 \in [0, 1]$
ESLPSO ^[17]	$\phi=0.5, \eta=0.01, \beta=1.5,$ $T_c=50, T_m=50$
LDPSO ^[18]	$w \in [0.4, 0.9], c_1=2.5-0.5, c_2=0.5-2.5$ $R_{\max}=0.5, R_{\min}=0.1, G=15, \alpha=0.2$
MSCPSO ^[19]	$w \in [0.4, 0.9], c_g^1=2, c_s^1=1, c_s^2=2, C_1=C_2=1.5$
HCLPSO ^[20]	$w \in [0.79, 0.99], c_1 \in [2, 2.5], c_2 \in [0.5, 2], kg \in [2.5, 3]$
ADDP SO	$w \in [0.2, 0.95], c_1 \in [0.2, 2.8], c_2 \in [0.2, 2.8], T_{\text{mit}}=3\ 000, CR=0.995$

表 3 混沌序列与均匀随机数比例参数设置

函数	准则	7:3	5:5	6:4	8:2	9:1
F18	mean	9.45×10^7	1.21×10^8	1.04×10^8	1.98×10^7	1.28×10^8
	std	6.16×10^7	7.28×10^7	6.25×10^7	1.46×10^7	9.73×10^7
F19	mean	6.66×10^3	8.31×10^3	9.31×10^3	7.87×10^3	6.31×10^3
	std	9.28×10^2	8.96×10^2	2.01×10^3	7.23×10^2	5.23×10^2
F20	mean	1.13×10^4	1.71×10^4	1.19×10^4	1.33×10^4	1.59×10^4
	std	2.09×10^3	2.08×10^3	8.97×10^2	1.61×10^3	2.20×10^3
F21	mean	2.47×10^3	2.47×10^3	2.48×10^3	2.48×10^3	2.50×10^3
	std	3.61×10^0	2.14×10^1	2.60×10^1	2.81×10^1	5.51×10^0
F22	mean	1.38×10^4	1.42×10^4	1.24×10^4	1.96×10^4	1.94×10^4
	std	1.82×10^3	1.07×10^3	1.45×10^3	8.55×10^3	2.75×10^3
F23	mean	2.00×10^4	2.37×10^4	2.06×10^4	2.13×10^4	2.54×10^4
	std	9.37×10^2	1.97×10^3	1.60×10^3	3.00×10^3	2.33×10^3
F24	mean	3.78×10^3	3.86×10^3	3.98×10^3	4.10×10^3	4.12×10^3
	std	1.79×10^2	1.20×10^2	8.09×10^1	2.69×10^2	2.15×10^2
F25	mean	6.30×10^3	6.51×10^3	6.54×10^3	6.38×10^3	6.32×10^3
	std	5.06×10^1	2.69×10^2	4.28×10^1	1.99×10^2	7.61×10^1
F26	mean	3.53×10^3	3.70×10^3	3.71×10^3	3.52×10^3	3.52×10^3
	std	4.68×10^1	5.43×10^1	2.04×10^2	7.17×10^1	3.14×10^1
F27	mean	3.25×10^3	3.27×10^3	3.25×10^3	3.27×10^3	3.28×10^3
	std	3.79×10^1	2.74×10^1	8.71×10^1	2.14×10^1	1.08×10^1
F28	mean	9.42×10^7	2.97×10^8	1.03×10^8	3.24×10^8	2.75×10^8
	std	6.44×10^7	1.75×10^8	4.69×10^7	2.06×10^8	1.63×10^8
F29	mean	3.52×10^8	5.41×10^8	3.60×10^8	4.59×10^8	5.78×10^8
	std	2.33×10^8	2.28×10^8	2.34×10^7	1.47×10^8	1.38×10^8

实验结果表明,不同融合比例对算法性能具有显著影响。在多数测试函数中,7:3的融合比例表现出最优的综合性能,其均值与标准差均优于其他配置。例如,在F23函数中,该比例对应的均值(2.00×10^4)显著低于5:5与6:4比例,标准差(9.37×10^2)也优于9:1比例,体现出良好的收敛精度与稳定性。在F28函数中,7:3比例所得均值(9.42×10^7)较5:5比例显著提升,进一步验证其在快速聚焦高质量解区域方面的优势。

综上所述,7:3的融合比例能够在混沌序列的遍历性与均匀随机数的扰动性之间达成有效平衡,为高维多峰优化问题提供稳定可靠的初始化种群。

4.4 消融实验

为系统评估ADDP SO算法中各核心模块的贡献,通过逐步移除关键组件构建四个对比版本,即采用标准随机初始化的ADDP SO-CI、移除所有变异操作的ADDP SO-AM、采用固定参数($w=0.9, c_1=2, c_2=2$)的ADDP SO-PA,以及移除模拟退火劣解接受机制的ADDP SO-SA。实验以CEC2017测试集的混合函数(F18和F19)与复合函数(F20~F29)为基准,并以均值与标准差作为评价指标,具体结果如表4所示。

实验结果表明,ADDP SO在大多数测试函数中均表现出显著优势。例如,在F20函数上,ADDP SO所得均值(1.13×10^4)明显低于ADDP SO-PA,说明参数自适应机制在高维搜索中具有关键作用。在F23函数中,ADDP SO-AM的标准差上升至 1.17×10^4 ,反映出多策略变异对维持种群多样性的必要影响。在F25测试函数上,ADDP SO-CI所获得的平均结果为 6.55×10^3 ,其性能表现弱于ADDP SO,这表明混合初始化策略在提升初始种群分布均匀性方面具有关键作用。总体来看,ADDP SO-CI所得解的质量出现显著下降,从而充分印证本文所提出的混合混沌初始化策略的有效性与必要性。

综上所述,消融实验不仅量化各组成模块的独立贡献,也揭示ADDP SO层次化架构中的协同优化机制。具体而言,多样性维持机制与参数自适应策略共同奠定搜索方向有效调控的基础。而变异算子与退火操作则通过引入恰当的扰动,显著增强算法在高维复杂优化问题中的全局收敛能力与鲁棒性。

4.5 求解精度比较分析

为全面评估所提出的自适应多样性驱动粒子群优化算法(ADDP SO)在高维复杂问题上的收敛性能与鲁棒性,表5汇总与7种代表性算法(PSO、CLPSO、

表 4 消融实验

函数	准则	ADDP SO	ADDP SO- CI	ADDP SO- PA	ADDP SO- AM
F18	mean	9.45×10^7	6.52×10^7	4.68×10^{10}	3.48×10^{13}
	std	6.16×10^7	3.15×10^7	5.95×10^9	1.23×10^{13}
F19	mean	6.66×10^3	1.16×10^4	1.22×10^4	2.55×10^4
	std	9.28×10^2	4.83×10^2	8.27×10^2	2.42×10^3
F20	mean	1.13×10^4	1.70×10^4	6.15×10^4	2.30×10^5
	std	2.09×10^3	1.58×10^3	2.01×10^4	1.26×10^4
F21	mean	2.47×10^3	2.45×10^3	3.13×10^3	8.39×10^3
	std	3.61×10^0	1.52×10^1	1.43×10^2	4.29×10^3
F22	mean	1.38×10^4	1.32×10^4	6.94×10^4	1.27×10^5
	std	1.82×10^3	1.73×10^3	7.38×10^3	3.32×10^3
F23	mean	2.00×10^4	2.43×10^4	1.08×10^5	1.77×10^5
	std	9.37×10^2	3.49×10^2	1.18×10^4	1.17×10^4
F24	mean	3.78×10^3	4.03×10^3	8.90×10^3	2.76×10^4
	std	1.79×10^2	1.67×10^2	8.19×10^2	3.42×10^3
F25	mean	6.30×10^3	6.55×10^3	8.76×10^3	1.28×10^4
	std	5.06×10^1	3.26×10^2	1.83×10^3	2.90×10^3
F26	mean	3.53×10^3	3.55×10^3	4.15×10^3	4.11×10^3
	std	4.68×10^1	1.07×10^2	1.18×10^2	2.15×10^2
F27	mean	3.25×10^3	3.31×10^3	5.70×10^3	1.44×10^4
	std	3.79×10^1	1.98×10^1	6.70×10^2	8.96×10^2
F28	mean	9.42×10^7	2.28×10^8	3.72×10^{10}	1.16×10^{14}
	std	6.44×10^7	9.54×10^7	2.62×10^9	9.76×10^{13}
F29	mean	3.52×10^8	1.37×10^9	3.75×10^{10}	1.40×10^{14}
	std	2.33×10^8	8.34×10^8	5.48×10^9	1.33×10^{14}

EAPSO、ESLPSO、LDPSO、MSCPSO、HCLPSO) 在 CEC2017 测试集的 12 个高维函数(包括混合函数 F18 和 F19 及复合函数 F20~F29)上独立运行 30 次后的统计结果. 表中列出平均值、标准差和最优解(best), 并以粗体标注各项最优指标. 其中, 平均值反映算法的整体寻优能力, 最优解体现其峰值性能潜力, 标准差则衡量结果的稳定性, 数值越小表明搜索过程越可靠.

由表 5 可知, ADDPSO 在大多数测试函数上均显著优于对比算法, 表现出优异的求解精度和稳定性. 在 12 个测试函数中, ADDPSO 在 9 个函数(F18、F20、F21、F22、F23、F24、F27、F28、F29)上均取得最优的均值与最优解, 且其标准差普遍为最低. 例如, 在复合函数 F20 上, ADDPSO 的均值为 1.13×10^4 , 相比次优算法 EAPSO (2.47×10^4) 有明显提升, 这表明其不仅收敛精度更高, 结果的一致性也更好. 在 F25 和 F26 上, 尽管 HCLPSO 的均值略优, 但 ADDPSO 表现出显著更低的标准差, 体现出更优异的稳定性与抗波动能力.

在混合函数 F18 和 F19 上, ADDPSO 表现尤为突出. 对于 F18, 其均值达到 9.45×10^7 , 较次优算法 EAPSO (6.27×10^{10}) 提高近三个数量级, 说明该算法能有效处理

多模态与异构搜索空间带来的挑战. 在 F19 上, ADDPSO 在最优解指标上取得最好结果 (5.83×10^3), 验证了其在平衡种群多样性与局部开发机制方面的有效性.

在更为复杂的复合函数 F20~F29 中, 由于该类函数具有强旋转不变性、多峰性及非线性等特点, 对算法的全局探索能力提出更高要求. ADDPSO 在 10 个复合函数中的 8 个(F20~F24、F27~F29)上均取得最小均值. 以 F22 为例, ADDPSO 均值为 1.38×10^4 , 相比 CLPSO (6.46×10^4) 有明显降低, 标准差为 1.82×10^3 也低于其他主流算法. 在 F28 上, ADDPSO 均值达 9.42×10^7 , 较 HCLPSO (1.80×10^{11}) 提升四个数量级, 并同时获得最优解 (2.71×10^7), 充分证明其在高维复杂环境中逃离局部极值、维持长期探索的能力具有显著优势.

综上所述, ADDPSO 通过引入自适应多样性驱动机制, 在种群进化过程中实现探索与开发的动态平衡, 有效缓解传统 PSO 算法早熟收敛与搜索方向单一等问题. 实验结果表明, 该算法在处理高维复杂结构的优化问题时, 在收敛精度与鲁棒性方面均优于现有主流 PSO 改进算法.

4.6 收敛性分析

为进一步验证算法收敛性能, 本文将所提 ADDPSO 算法与标准粒子群优化(PSO)算法及 6 种典型 PSO 变体算法的种群收敛性进行对比分析, 结果如图 1 所示. 在图 1 中, 横坐标代表迭代次数, 范围从 0 到最大迭代次数 ($t_{\max} = 1000$). 纵坐标轴代表适应度值, 采用对数刻度 (\log_{10}) 表示, 展示算法在迭代过程中适应度值的变化.

从图 1 可以看出, ADDPSO 在绝大多数测试函数上均展现出更优的收敛速度与最终求解精度, 显示出系统而稳健的优化性能. 在迭代初期, 其基于 Logistic 混沌映射与均匀分布混合的种群初始化策略有效提升初始解的分布均匀性与空间遍历性, 结合精英引导的层次化搜索机制, 能够迅速定位潜在的高质量解区域. 在迭代中后期, 依托多样性反馈的双驱动参数调节机制与多策略自适应变异操作, 显著延缓种群多样性的衰退进程, 有效抑制早熟收敛现象.

尽管在如 F25 和 F26 等复合函数上, ADDPSO 与部分先进算法(如 EAPSO 和 HCLPSO)的性能差距并不显著, 表明这些算法在特定问题结构下仍具有一定竞争力. 然而, 在面对高维、多峰及强旋转的混合和复合函数测试集(F18、F20~F24、F27~F29)时, ADDPSO 的优势尤为突出. 以 F20 为例, 该函数具有高度复杂的局部极值分布与极小的适应度差异, 对算法的全局探索和局部开发平衡能力提出极高要求. 多数对比算法在迭代中期便陷入局部最优, 而 ADDPSO 凭借其多样性感知

表5 运行结果对比

函数	准则	ADDP SO	PSO	CLPSO	EAPSO	ESLPSO	LDPSO	MSCPSO	HCLPSO
F18	mean	9.45×10^7	3.67×10^{12}	2.35×10^{12}	6.27×10^{10}	2.93×10^{16}	6.64×10^{10}	2.92×10^{13}	3.34×10^{11}
	std	6.16×10^7	4.78×10^{12}	1.00×10^{12}	5.78×10^{10}	9.19×10^{15}	7.35×10^9	2.10×10^{13}	2.77×10^{11}
	best	1.96×10^7	1.60×10^{11}	1.11×10^{12}	1.77×10^{10}	1.70×10^{16}	5.69×10^{10}	6.91×10^{12}	1.42×10^{10}
F19	mean	6.66×10^3	1.79×10^4	2.10×10^4	6.86×10^3	4.77×10^4	1.12×10^4	2.58×10^4	6.13×10^3
	std	9.28×10^2	1.44×10^3	2.97×10^3	1.39×10^3	3.69×10^3	1.85×10^3	5.37×10^3	2.18×10^2
	best	5.83×10^3	1.59×10^4	1.68×10^4	4.91×10^3	4.25×10^4	8.63×10^3	2.12×10^4	5.88×10^3
F20	mean	1.13×10^4	7.13×10^4	2.34×10^5	2.47×10^4	2.78×10^5	1.61×10^5	2.95×10^5	5.10×10^4
	std	2.09×10^3	2.29×10^4	1.17×10^4	6.82×10^3	6.67×10^2	9.82×10^3	5.37×10^4	5.39×10^3
	best	8.43×10^3	4.89×10^4	2.24×10^5	1.51×10^4	2.77×10^5	1.48×10^5	2.52×10^5	4.64×10^4
F21	mean	2.47×10^3	5.85×10^3	8.78×10^3	2.78×10^3	3.12×10^4	3.50×10^3	3.59×10^3	2.84×10^3
	std	3.61×10^0	1.16×10^3	5.44×10^3	4.92×10^1	0.00×10^0	1.72×10^2	3.58×10^2	1.22×10^1
	best	2.47×10^3	4.99×10^3	3.98×10^3	2.74×10^3	3.12×10^4	3.37×10^3	3.31×10^3	2.83×10^3
F22	mean	1.38×10^4	5.44×10^4	6.46×10^4	2.21×10^4	1.33×10^5	1.21×10^5	1.63×10^5	4.00×10^4
	std	1.82×10^3	1.77×10^4	2.57×10^3	1.16×10^3	1.15×10^{-3}	2.66×10^3	1.59×10^4	2.02×10^3
	best	1.24×10^4	3.00×10^4	6.22×10^4	2.09×10^4	1.33×10^5	1.18×10^5	1.40×10^5	3.72×10^4
F23	mean	2.00×10^4	7.92×10^4	7.57×10^4	3.31×10^4	1.94×10^5	1.55×10^5	1.54×10^5	4.53×10^4
	std	9.37×10^2	1.14×10^4	2.70×10^3	4.23×10^3	0.00×10^0	2.27×10^3	5.58×10^4	8.59×10^2
	best	1.88×10^4	6.71×10^4	7.30×10^4	2.73×10^4	1.94×10^5	1.53×10^5	7.59×10^4	4.41×10^4
F24	mean	3.78×10^3	8.72×10^3	2.63×10^4	3.93×10^3	5.39×10^4	1.73×10^4	1.83×10^4	4.63×10^3
	std	1.79×10^2	8.31×10^2	9.84×10^2	1.65×10^2	2.63×10^3	4.52×10^3	1.49×10^3	1.99×10^2
	best	3.62×10^3	7.55×10^3	2.51×10^4	3.76×10^3	5.17×10^4	1.40×10^4	1.68×10^4	4.44×10^3
F25	mean	6.30×10^3	3.74×10^4	6.90×10^3	6.87×10^3	3.94×10^5	1.69×10^4	3.14×10^4	6.12×10^3
	std	5.06×10^1	4.19×10^3	1.00×10^1	6.43×10^2	2.66×10^4	5.32×10^3	1.18×10^4	6.97×10^1
	best	6.23×10^3	3.26×10^4	6.88×10^3	6.23×10^3	3.57×10^5	1.07×10^4	1.48×10^4	6.02×10^3
F26	mean	3.53×10^3	5.71×10^3	4.00×10^3	3.49×10^3	1.34×10^4	4.34×10^3	4.21×10^3	3.43×10^3
	std	4.68×10^1	6.18×10^2	8.89×10^1	7.66×10^1	1.78×10^3	3.48×10^2	9.12×10^1	4.98×10^1
	best	3.47×10^3	4.86×10^3	3.90×10^3	3.40×10^3	1.10×10^4	3.85×10^3	4.10×10^3	3.36×10^3
F27	mean	3.25×10^3	7.68×10^3	7.53×10^3	4.26×10^3	3.60×10^4	9.65×10^3	8.89×10^3	5.17×10^3
	std	3.79×10^1	7.01×10^2	2.16×10^2	4.69×10^2	2.11×10^3	7.39×10^2	1.81×10^2	1.85×10^2
	best	3.21×10^3	6.79×10^3	7.23×10^3	3.63×10^3	3.32×10^4	8.68×10^3	8.70×10^3	5.00×10^3
F28	mean	9.42×10^7	2.07×10^{14}	6.59×10^{12}	1.39×10^9	4.99×10^{16}	4.14×10^{11}	6.98×10^{14}	1.80×10^{11}
	std	6.44×10^7	2.78×10^{14}	3.00×10^{12}	7.81×10^8	2.92×10^{16}	5.17×10^{11}	9.73×10^{14}	1.40×10^{11}
	best	2.71×10^7	1.21×10^{11}	2.62×10^{12}	3.89×10^8	1.18×10^{16}	3.70×10^{10}	9.38×10^{12}	5.34×10^9
F29	mean	3.52×10^8	9.37×10^{10}	2.31×10^{13}	5.08×10^{10}	1.60×10^{16}	2.62×10^{11}	1.47×10^{14}	3.34×10^{10}
	std	2.33×10^8	1.96×10^{10}	1.01×10^{13}	3.42×10^{10}	1.17×10^{16}	1.04×10^{11}	1.68×10^{14}	1.92×10^{10}
	best	1.45×10^8	7.57×10^{10}	9.73×10^{12}	5.77×10^9	7.42×10^{15}	1.77×10^{11}	2.41×10^{13}	1.21×10^{10}

的模拟退火机制、多策略变异框架与动态边界控制,表现出卓越的逃离局部极值能力和收敛鲁棒性。

综上所述,ADDP SO从多样性度量、参数自适应、结构化信息交互和多策略扰动等多个层面系统应对 PSO 存在问题(参数僵化、搜索方向单一和种群早熟收敛)。实验结果表明,该算法在高维复杂优化问题中具有更优的整体性能和更高的求解成功率,显著优于现有主流 PSO 变体。

4.7 多样性分析

收敛性体现算法逼近最优解的能力,而种群多样性是衡量粒子群优化算法探索能力的关键指标,直观反映算法在全局探索与局部开发之间的平衡状态^[21]。较高的多样性意味着粒子在解空间中分布广泛,有助于探索未知区域并增强跳出局部最优的能力;而多样性下降则表明粒子趋于聚集,算法转向局部开发为主的精细搜索阶段。因此,多样性随迭代的动态变化可作为分析算法搜

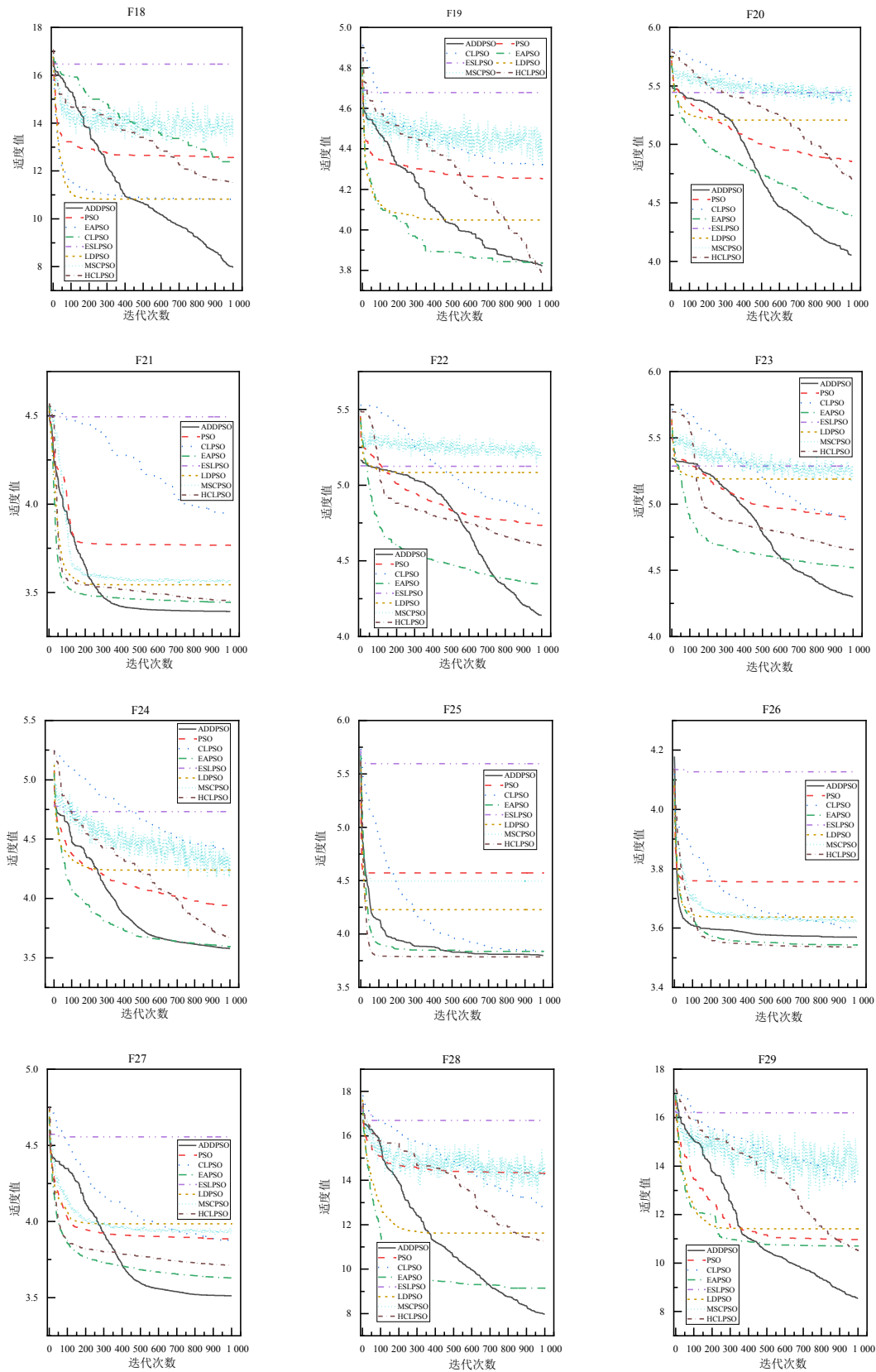


图1 算法收敛性

索行为的重要依据. 种群多样性度量公式如下:

$$\text{diversity} = \frac{1}{N \cdot \text{dim}} \sum_{i=1}^N \text{disc}_i \quad (30)$$

其中, $\text{disc}_i = \sum_{j=1}^{\text{dim}} |x_{i,j} - m_j|$ 为第 i 个第 j 维粒子 $x_{i,j}$ 与第 j 维

中位数 (m_j) 的绝对偏差和. 本文提出的 ADDPSO 与 PSO 以及六种不同 PSO 变体算法的种群多样性对比结果如图 2 所示. 纵坐标是种群多样性度量值, 展示其在优化过程中种群多样性的变化. 横坐标代表迭代次数, 范围从 0 到最大迭代次数 ($t_{\max} = 1000$).

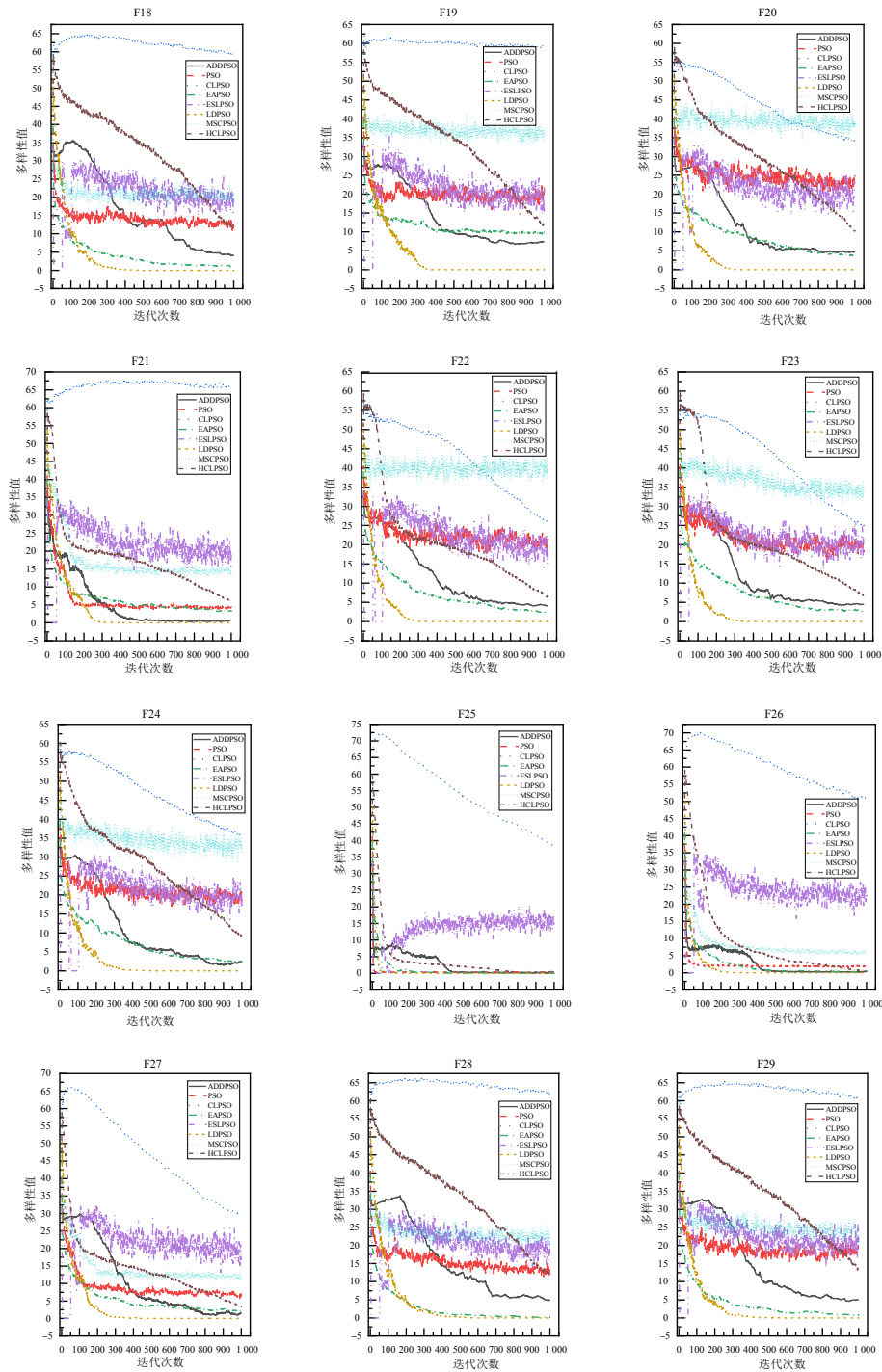


图2 算法种群多样性

根据图2,从种群多样性的动态演化特性来看,传统改进算法如CLPSO、EAPSO和LDPSO等虽借助个体学习或leader引导机制在一定程度上维持种群多样性,其曲线亦呈现一定波动,但该类调节多依赖于固定规则,属于被动响应,难以适应解空间结构在搜索过程中的动态变化.相比之下,ADDPSO算法表现出显著且规律性的多样性波动,其优势源于一种主动式多样性调控机制,通过基于平均欧氏距离的divRatio实时反馈种群同质化风险,并据此动态调整惯性权重与个体学习因子以增强探索能力.同时,结合多策略自适应变异与多样性感知模拟退火,在迭代中后期随机施加高斯或柯西变异,并基于divRatio动态调节退火温度,从而主动引入差异化个体,实现全局探索与局部开发间的平衡.在演化后期,当对比算法多样性普遍低于0.1并陷入停滞时,ADDPSO仍可维持合理多样性水平,这得益于其速度边界动态调整机制,有效避免搜索震荡与收敛停滞.尤其在复杂多峰环境中,ADDPSO通过三层速度更新机制实现多样性的恢复,显示出优异跳出局部最优的能力.

综上所述,ADDPSO算法通过“感知-反馈-调控”的闭环管理框架,有效解决了传统PSO变体中多样性维持与收敛速度之间的平衡难题,为处理高维多峰复杂优化问题提供了可靠的解决方案.

4.8 统计结果分析

为系统评估本文提出的ADDPSO性能,并在高维复杂优化问题中验证其统计显著性,本研究基于CEC2017测试集(涵盖混合函数F18、F19与复合函数F20~F29),采用Wilcoxon秩和检验及Friedman检验,对ADDPSO与七种主流PSO改进算法进行严格的统计比较分析.

Wilcoxon秩和检验是一种非参数统计方法,用于判断两组独立样本是否来自同一分布.本研究设定显著性水平 $\alpha=0.05$,若检验 p 值不超过该阈值,则拒绝原假设,认为算法性能存在显著差异.从表6分析结果表明,ADDPSO算法在与PSO、CLPSO、ESLPSO、LDPSO及MSCPSO对比中,正秩和均为0,负秩和均为78,且 p 值均低至 4.88×10^{-4} ,远低于0.05.这说明在全部测试函数上,ADDPSO均显著优于上述算法.与EAPSO相比,ADDPSO的负秩和为77($p=9.77 \times 10^{-4}$),表明其在绝大多数测试函数上性能更优,仅在极个别问题(如F19)略逊.此外,在与当前性能较强的HCLPSO对比中,ADDPSO仍取得负秩和71(正秩和7), p 值为 9.28×10^{-3} ,达到显著水平($p < 0.05$).特别在复合函数F20~F24上,ADDPSO表现出更优的优化性能.

为评估ADDPSO与其他算法的综合性能差异,本

表6 不同算法的Wilcoxon对比

算法	正秩和	负秩和	p 值
ADDPSO VS PSO	0	78	4.88×10^{-4}
ADDPSO VS CLPSO	0	78	4.88×10^{-4}
ADDPSO VS EAPSO	1	77	9.77×10^{-4}
ADDPSO VS ESLPSO	0	78	4.88×10^{-4}
ADDPSO VS LDPSO	0	78	4.88×10^{-4}
ADDPSO VS MSCPSO	0	78	4.88×10^{-4}
ADDPSO VS HCLPSO	7	71	9.28×10^{-3}

研究进一步采用Friedman排序检验进行多算法比较分析.如图3所示,ADDPSO的平均排序值显著低于所有对比算法,稳居首位.该结果表明,在高维多峰、非线性及旋转不变性等复杂测试函数上,ADDPSO的整体性能具有统计学显著优势.此排序结果与其在全局探索和局部开发的平衡能力形成机理互证,印证ADDPSO算法在多场景下的鲁棒性.

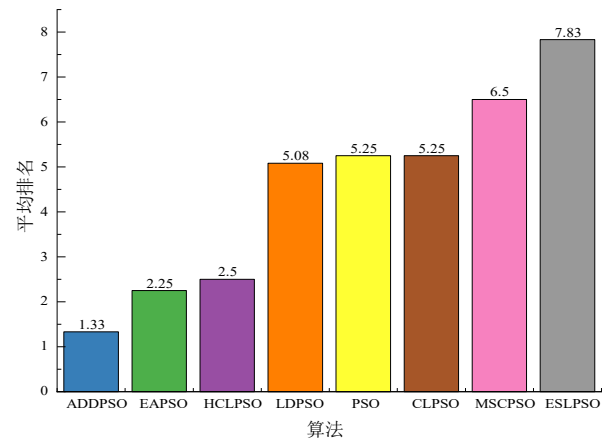


图3 不同算法的Friedman排序

4.9 齿轮系设计问题

齿轮系设计问题(Gear Train Design, GTD)是机械工程中一类典型的无约束离散优化问题,其目标为最小化输出轴与输入轴的角速度比.该问题的设计变量为四个齿轮的齿数,分别记为 $n_A=x_1$ 、 $n_B=x_2$ 、 $n_C=x_3$ 和 $n_D=x_4$.GTD的数学模型具体如下:

$$\min f(x) = \left(\frac{1}{6.931} - \frac{x_2 x_3}{x_1 x_4} \right)^2 \quad (30)$$

其中, $x_1, x_2, x_3, x_4 \in \{12, 13, 14, \dots, 60\}$.

为验证ADDPSO在离散优化问题中的有效性,本节将其与ESLPSO、LDPSO、MSCPSO、HCLPSO、EAPSO、

PSO 及 CLPSO 等主流算法进行对比. 如表 7 所示, 所有算法均能找到理论最优解, 表明在解质量方面算法间差异不显著.

然而, 在稳定性与鲁棒性方面, ADDPSO 表现出显著优势. 其所得解的标准差远低于其他算法, 说明算法

在多次运行中结果波动极小; 同时, ADDPSO 在最优值、最差值等统计指标上均一致趋近于 0, 验证其良好的收敛一致性. 相比之下, ESLPSO 等算法虽也达到理论最优, 但均值和标准差较大, 表明优化过程存在显著的不稳定性.

表 7 齿轮系设计中不同算法对比结果

	ADDPSO	ESLPSO	LDPSO	MSCPSO	HCLPSO	EAPSO	PSO	CLPSO
x_1	50	26	56	52	57	51	60	51
x_2	18	13	26	24	18	22	37	24
x_3	22	14	22	17	27	17	14	18
x_4	53	21	54	48	46	47	58	50
mean	1.05×10^{-28}	6.17×10^{-2}	4.64×10^{-19}	3.47×10^{-6}	1.28×10^{-12}	3.10×10^{-12}	8.85×10^{-12}	8.25×10^{-12}
std	2.10×10^{-28}	7.56×10^{-2}	8.66×10^{-19}	5.18×10^{-6}	1.17×10^{-12}	3.76×10^{-12}	1.73×10^{-11}	1.06×10^{-11}
best	0.00×10^0	1.14×10^{-2}	0.00×10^0	1.02×10^{-8}	2.63×10^{-16}	1.50×10^{-15}	1.45×10^{-18}	1.58×10^{-15}
worst	5.26×10^{-28}	2.09×10^{-1}	2.19×10^{-18}	1.37×10^{-5}	3.03×10^{-12}	8.01×10^{-12}	4.35×10^{-11}	2.59×10^{-11}

5 结论

为解决粒子群优化算法在处理高维多峰复杂优化问题时, 因种群多样性丧失而易陷入早熟收敛以及收敛效率降低的问题, 本文提出一种自适应多样性驱动的粒子群优化算法(ADDPSO). 该算法在粒子群优化算法框架基础上, 从多样性度量机制、参数自适应策略、信息交互结构以及扰动机制等多个维度进行系统性重构, 显著增强算法在复杂搜索空间中的全局探索能力与鲁棒性. 本文的主要贡献包括: 第一, 提出了混合初始化策略与基于平均欧氏距离的多样性度量方法, 为全局探索奠定了分布基础并实现了种群状态的量化评估; 第二, 建立了时间递减与多样性反馈相结合的双驱动参数自适应机制, 实现了对惯性权重与学习因子的协同调节, 并构建了“个体-精英-种群”三层信息融合的速度更新方式, 减少了对单一全局最优的依赖; 第三, 引入多样性感知的模拟退火机制与多策略自适应变异算子, 结合动态速度边界与变异策略, 有效抑制了早熟收敛并增强了算法在复杂环境下的适应性.

基于 CEC2017 测试集的实验结果表明, ADDPSO 在多数高维复杂函数(混合函数 F18、F19 与复合函数 F20~F29)上均显著优于主流 PSO 变体算法. 此外, 在齿轮系设计这一典型优化问题中, ADDPSO 稳定收敛, 且其优化结果的标准差低至 2.10×10^{-28} , 显著优于对比算法, 展现出工程适用性与收敛一致性.

未来工作将围绕算法计算效率的进一步提升展开, 同时拓展 ADDPSO 在特征选择、物联网数据收集优化、燃料电池系统参数估计等实际工程场景中的应用, 并结合约束处理机制以应对更复杂的优化问题.

参考文献

- [1] KENNEDY J, EBERHART R. Particle swarm optimization[C]//Proceedings of ICNN'95 - International Conference on Neural Networks. Piscataway: IEEE, 2002: 1942-1948.
- [2] HE J L, QU L D, WANG P, et al. An oscillatory particle swarm optimization feature selection algorithm for hybrid data based on mutual information entropy[J]. Applied Soft Computing, 2024, 152: 111261.
- [3] 王毅, 李晓梦, 耿国华, 等. 基于直觉模糊熵的混合粒子群优化算法[J]. 电子学报, 2021, 49(12): 2381-2389.
WANG Y, LI X M, GENG G H, et al. Hybrid particle swarm optimization algorithm based on intuitionistic fuzzy entropy[J]. Acta Electronica Sinica, 2021, 49(12): 2381-2389. (in Chinese)
- [4] 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)
- [5] GAO W F, LIU S Y, HUANG L L. Particle swarm optimization with chaotic opposition-based population initialization and stochastic search technique[J]. Communications in Nonlinear Science and Numerical Simulation, 2012, 17(11): 4316-4327.
- [6] GAO J R, WANG Z Q, JIN T, et al. Information gain ratio-based subfeature grouping empowers particle swarm opti-

- mization for feature selection[J]. Knowledge-Based Systems, 2024, 286: 111380.
- [7] SUN W, LIN A P, YU H S, et al. All-dimension neighborhood based particle swarm optimization with randomly selected neighbors[J]. Information Sciences, 2017, 405: 141-156.
- [8] WANG F, WANG X J, SUN S L. A reinforcement learning level-based particle swarm optimization algorithm for large-scale optimization[J]. Information Sciences, 2022, 602: 298-312.
- [9] LIANG J J, QIN A K, SUGANTHAN P N, et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(3): 281-295.
- [10] LI C, ZHAI Y H, PALADE V, et al. Diversity-based adaptive differential evolution algorithm for multimodal optimization problems[J]. Swarm and Evolutionary Computation, 2025, 93: 101869.
- [11] LIU Z G, JI X H, YANG Y, et al. Multi-technique diversity-based particle-swarm optimization[J]. Information Sciences, 2021, 577: 298-323.
- [12] FU W Y. Adaptive-acceleration-empowered collaborative particle swarm optimization[J]. Information Sciences, 2025, 721: 122621.
- [13] HOUSSEIN E H, GAD A G, HUSSAIN K, et al. Major advances in particle swarm optimization: Theory, analysis, and application[J]. Swarm and Evolutionary Computation, 2021, 63: 100868.
- [14] LI A D, XUE B, ZHANG M J. Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies[J]. Applied Soft Computing, 2021, 106: 107302.
- [15] YAN J L, HU G, JIA H M, et al. GPSOM: Group-based particle swarm optimization with multiple strategies for engineering applications[J]. Journal of Big Data, 2025, 12(1): 114.
- [16] ZHANG Y Y. Elite archives-driven particle swarm optimization for large scale numerical optimization and its engineering applications[J]. Swarm and Evolutionary Computation, 2023, 76: 101212.
- [17] 齐铖, 谢军伟, 王雪, 等. 基于精英引导的社会学习粒子群优化算法[J]. 西北工业大学学报, 2024, 42(5): 948-958.
QI C, XIE J W, WANG X, et al. A novel elite guidance-based social learning particle swarm optimization algorithm[J]. Journal of Northwestern Polytechnical University, 2024, 42(5): 948-958. (in Chinese)
- [18] ZHAO S C, ZHOU H, ZHOU H. Leaders-driven particle swarm optimizer[J]. Expert Systems with Applications, 2025, 292: 128595.
- [19] TANG Y, HUANG K C, TAN Z P, et al. Multi-subswarm cooperative particle swarm optimization algorithm and its application[J]. Information Sciences, 2024, 677: 120887.
- [20] ZHAO Y L, WU F, PANG J H, et al. New heterogeneous comprehensive learning particle swarm optimizer enhanced with low-discrepancy sequences and conjugate gradient method[J]. Swarm and Evolutionary Computation, 2025, 93: 101848.
- [21] ZHANG L, CHEN X B. Elite-driven grey wolf optimization for global optimization and its application to feature selection[J]. Swarm and Evolutionary Computation, 2025, 92: 101795.

作者简介



张 俐 男, 1977年11月出生于陕西省汉中市. 江苏理工学院计算机工程学院副教授. 2019年毕业于北京邮电大学计算机学院(国家示范性软件学院), 获工学博士学位. 主要研究方向为智能优化算法、强化学习以及特征选择.

E-mail: zhangli_3913@163.com