

基于精英引导的社会学习粒子群优化算法

齐铖¹, 谢军伟¹, 王雪², 冯为可¹, 张浩为¹

(1.空军工程大学 防空反导学院, 陕西 西安 710051; 2.滨州职业学院 信息工程学院, 滨州 256600)

摘要: 为了改进经典粒子群算法(PSO)过早收敛和全局搜索能力不足的缺点,提出了一种基于精英引导的社会学习粒子群优化算法(ESLPSO)。在 ESLPSO 中,提出了一种分层拓扑结构的搜索方法。这一策略根据粒子的适应度表现将粒子分化为最优的精英粒子和其余的平民粒子,革新了传统种群迭代搜索的更新样本,由此加强了整个种群演化信息的引导作用。采用 Cubic 混沌初始化赋予了初始粒子群体在搜索空间内的广域覆盖能力。设计了精英粒子引导的社会学习策略,通过增加态叠加的不确定性更好地利用了种群演化的多维信息。在此基础上,结合极值扰动迁移机制激励粒子经历新的搜索路径和区域,增加种群的多样性,平衡种群在搜索过程中的探索和开发能力。基于 12 个涵盖单峰、多峰以及旋转多峰的基准测试函数集对所提算法的性能进行了验证。此外,ESLPSO 与其他 8 种 PSO 改进算法的比较结果表明,ESLPSO 在解决不同类型函数方面表现出了优秀的搜索性能,具有高效的求解稳定性和优异的求解结果。

关键词: 粒子群优化; 社会学习; Cubic 混沌; 极值扰动

中图分类号: TP182

文献标志码: A

文章编号: 1000-2758(2024)05-0948-11

优化问题长期以来一直是一个基础研究课题,其在农业生产管理、军事指挥运筹、大规模复杂性网络、资源分配、任务分配等领域中都有着巨大的应用潜力,吸引了广泛的探讨和研究^[1-5]。进化算法是高效求解复杂优化问题最常用的一类算法,包括遗传算法^[6]、禁忌搜索^[7]和模拟退化^[8]等,其中,粒子群优化(particle swarm optimization, PSO)算法凭借结构简单、收敛迅速、计算成本低以及通用性强等优点脱颖而出^[9-10]。

PSO 算法是于 1995 年由 Kenny 和 Eberhart 受到人工生命结果的启发,通过模拟鸟群觅食过程中的迁徙和群聚行为而提出的一种基于群体智能的全局随机搜索算法^[11]。与其他进化算法相类似,PSO 算法基本思想也是基于“种群”和“进化”的生物社会背景概念,粒子运动源于个体之间的协作与竞争活动的规律性,利用种群中信息的社会共享产生从

无序到有序的演化运动,实现在复杂空间对最优解的搜索^[12-13]。目前已经被广泛应用于多种类型问题的求解,如高维度优化问题、多模优化问题、多目标优化问题和离散超空间优化问题等,在资源管理、电力系统等领域也被广泛应用^[14]。

同时 PSO 算法也有着不可忽视的缺陷和不足^[15]。一方面是难以平衡粒子多样性和算法收敛速度,使得群体演化过程容易陷入局部最优;另一方面,粒子仅认知学习粒子最优和种群最优的位置信息进行更新,而忽略了其他有效信息,导致搜索策略执行效率低下。实际上,PSO 算法的收敛速度及精度与搜索策略和相应的控制参数等因素密切相关^[16]。针对以上问题,诸多学者对 PSO 算法进行了优化和改进,相关研究可以概括为四方面:控制参数、种群拓扑结构、学习策略和融合算法。

1) PSO 算法惯性权重和加速度系数等控制参数的自适应调整

文献[17]向原始粒子群算法中引入了随时间递减的惯性权重系数,大大提高了粒子群算法的性能,一定程度上实现了粒子进化中搜索和开发之间的平衡。文献[18]定义了一个 PSO 算法的广义模

收稿日期: 2023-09-10

基金项目: 中国科学技术协会青年人才托举项目(2021-JCJQ-QT-018)
资助

作者简介: 齐铖(1998—),博士研究生

通信作者: 谢军伟(1972—),教授 e-mail: qc_afeu@163.com

型,该模型采用带有压缩系数的随机系数,有效提升了算法的收敛度。文献[9]定义了自适应粒子群优化(adaptive particle swarm optimization, APSO)算法实时框架下粒子进化的搜索、开发、收敛和跳出 4 种状态,并据此实现了对惯性权值和加速度系数等算法参数的自适应控制,增强了算法的局部搜索能力。文献[19]引入了时变加速度系数,在避免搜索前期早熟收敛的同时提高了优化末尾的收敛精度。文献[20]提出了一种基于 sigmoid 函数的加权策略来自适应调整加速度系数,在其中同时考虑了粒子全局最佳位置和到自身最佳位置的距离,最终显著提升了收敛速度。

2) PSO 算法中粒子分布邻域拓扑结构的改进
种群拓扑结构的设计对于算法性能有着关键作用,例如全连接结构、环形结构等各种拓扑结构均有不同的影响^[21]。文献[22]采用一种综合学习策略,粒子的每个维度独立地选择范例学习,更加有效地利用种群信息,显著提高了 PSO 的性能。文献[23]将动态学习策略分为普通粒子的开发行为和通信粒子的探索行为,促进粒子之间的信息交换。文献[24]依据粒子间距离信息动态调整粒子个体邻域,并结合切换学习策略以便充分利用整个蜂群的种群演化信息。

3) PSO 算法中粒子进化学习策略的开发
文献[22]提出一种采用综合学习策略的变体 PSO 算法,进化粒子使用所有其他粒子的历史最佳信息更新粒子的速度,保留了种群的多样性,避免了早熟收敛。文献[25]将动物世界中的社会学习机制引入到粒子群算法中,赋予粒子向任意示范粒子学习的能力,提升了算法的搜索效率。文献[26]采用了一种动态分段的混合学习策略,分别利用基于动态分段的均值学习策略(DSMLS)和多维综合学习策略(MDCLS)实现种群协同进化和收敛速度精度的提升。

4) PSO 算法与其他优化技术的交叉融合
人们尝试将 PSO 算法与其他算法相结合而形成一种性能更为优异的混合粒子群算法^[27]。例如,遗传算法改进的粒子群(PSO-genetic algorithm, PSO-GA)算法^[28]、遗传算法改进的粒子群算法(PSO-simulated annealing, PSO-SA)算法^[29]、差分进化算法改进的粒子群(PSO-differential evolution, PSO-DE)算法^[29]等。除此之外,文献[10]提出了基于小生境技术的粒子群优化算法,提高了粒子群算

法求解多峰优化问题的能力;文献[30]结合混沌运动遍历性改善了粒子优化算法容易陷入局部极值的问题,提高了算法的收敛精度和速度;文献[31]专门在粒子群算法中引入局部搜索策略,实现了在局部最优位置对潜在较优解的搜索,打破了粒子在局部最优的停滞,并增加了收敛速度。

总而言之,以上 4 种 PSO 的改进方式都是通过改变种群迭代方式或者增强算法的种群多样性,从而强化算法的全局搜索能力、缓解早熟的问题,最终提高算法的收敛性能。虽然现有的粒子群优化算法改进研究主要集中于状态初始化、种群进化拓扑或粒子更新机制等单一环节的改进设计,但这些改进算法在某些环节中仍然存在优化瓶颈。因此,为了提高优化效果,需要从整体上协调粒子进化的各个环节,并设计基于粒子进化全环节的改进技术。

为此,同时从以上总结的 4 个改进角度出发,提出一种新的分层组织的精英引导社会学习粒子群优化算法(elite guidance-based social learning particle swarm optimization, ESLPSO)。首先依据粒子适应度表现动态划分精英粒子和平民粒子改进算法的拓扑结构,为不同粒子分配定制调优方法,使粒子在应对复杂情况时发挥不同的作用。同时引入社会学习机制拓展种群演化过程中的多维信息,并基于精英粒子引导的种群集体经验驱动平民粒子生成更加有潜力的子代粒子。利用多维种群信息改善了基本 PSO 算法对于种群最优和全局最优粒子的过度依赖。此外,为平衡种群在搜索过程中的探索和开发能力,在算法中融合扰动操作,在平民粒子社会学习策略中增加极值扰动机制赋予子代粒子态叠加的不确定性,以平衡种群的搜索和开发能力。

1 粒子群算法模型

1.1 经典粒子群算法

粒子群优化算法是一种基于迭代的全局随机搜索算法,它将问题的若干个潜在解抽象为粒子,通过群体在搜索空间的飞行行为模拟迭代优化问题。粒子种群规模为 N ,每个粒子代表 D 维搜索空间中的一个潜在解决方案。第 t 次迭代中第 i 个粒子的位置矢量和速度矢量分别表示为 $\mathbf{X}_i^t = (\mathbf{X}_{i,1}^t, \mathbf{X}_{i,2}^t, \dots, \mathbf{X}_{i,D}^t)$ 与 $\mathbf{V}_i^t = (\mathbf{V}_{i,1}^t, \mathbf{V}_{i,2}^t, \dots, \mathbf{V}_{i,D}^t)$ 。在群体进化过程中,粒子群算法会不断生成和更新第 i 个粒子的最

佳历史位置 $p_{best,i}^t = (p_{best,i,1}^t, p_{best,i,2}^t, \dots, p_{best,i,D}^t)$, 以及群体最优位置 $g_{best}^t = (g_{best,1}^t, g_{best,2}^t, \dots, g_{best,D}^t)$ 。经典粒子群算法中粒子位置与速度的更新策略为

$$\mathbf{X}_{i,j}^{t+1} = \mathbf{X}_{i,j}^t + \mathbf{V}_{i,j}^{t+1} \quad (1)$$

$$\mathbf{V}_{i,j}^{t+1} = \omega \cdot \mathbf{V}_{i,j}^t + c_1 \cdot u_1^t \cdot (p_{best,i,j}^t - \mathbf{X}_{i,j}^t) + c_2 \cdot u_2^t \cdot (g_{best,j}^t - \mathbf{X}_{i,j}^t) \quad (2)$$

式中, 惯性权重系数 ω 决定粒子对上一次速度的继承比例; c_1 和 c_2 分别为自身加速常数和社会加速常数, 影响粒子在搜索空间内局部或者全局影响的比重, c_1 控制自我认知部分, 体现粒子向该粒子的历史最佳位置飞行动作, c_2 控制社会认知部分, 指导粒子与种群之间的信息; u_1^t 和 u_2^t 是均匀分布在 $[0, 1]$ 之间的随机数。为每个粒子确定的速度值设置速度限制 $[V_{\max}, V_{\min}]$ 以防止粒子急剧变化。

显然, 在经典粒子群架构下粒子进化迭代分别是由原始惯性速度、粒子自我认知记忆以及种群社会的信息交互三部分共同控制, 拥有结构简单、求解快捷的优点, 在处理复杂优化问题上具有一定的优势。但 PSO 算法结构单一且参数固定, 在优化后期粒子飞行速度容易急剧降低, 导致收敛速度变慢, 产生早熟收敛和难以摆脱局部极值的问题。

1.2 典型的 PSO 改进算法

本文选择 2 个典型的 PSO 变体作为介绍, 这些 PSO 变体启发了本文提出的算法。

1) 综合学习粒子群优化器 (comprehensive learning particle swarm optimizer, CLPSO)

CLPSO 通过使用所有其他粒子的历史最佳信息更新粒子的速度以维持种群的多样性, 避免早熟收敛问题^[22]。该学习策略中使用了以下速度更新方程

$$\mathbf{V}_{i,j}^{t+1} = \omega \cdot \mathbf{V}_{i,j}^t + c \cdot u_1^{t+1} \cdot (p_{best,fi(d),j}^t - \mathbf{X}_{i,j}^t) \quad (3)$$

式中, ω, c, u_1^{t+1} 与基本粒子群算法中的定义一致。不同的是, CLPSO 将粒子学习对象扩大到整个种群, 粒子将根据其随机数与学习概率决定粒子学习。

该算法将粒子学习对象的范围扩大到整个种群, 范例粒子的最佳历史位置信息更加全面地反映了整个种群, 为保持种群多样性提供强有力的保障。但是, 其舍弃了全局最优粒子这一维关键信息, 导致收敛速度和搜索精度具有局限性。

2) 自调节自进化粒子群优化器 (self-regulating and self-evolving particle swarm optimizer, SRPSO)

文献[32]提出一种自调节自进化的粒子位置

更新策略以加快收敛速度、增加多样性和防止过早收敛。当搜索过程中粒子的 $p_{best,i}^t$ 未能提高时, 结合反转思想赋予粒子一个反向速度更新位置

$$\mathbf{X}_{i,j}^{t+1} = \mathbf{X}_{i,j}^t + c_3 \mathbf{R}_3 (-\mathbf{V}_{i,j}^{t+1}) \quad (4)$$

式中: c_3 为控制粒子飞行的正常数; $\mathbf{R}_3$ 为对角元素随机分布于 $[0, 1]$ 的 $D \times D$ 矩阵。在此基础上, 结合自进化行为突变操作生成新的全局最优粒子。粒子最优位置和全局最优位置均采用贪婪更新思想, 分别对一般粒子和全局最优粒子的更新进行改进, 所提出的这种分层处理机制有效提升了算法的收敛性能和可靠性。

2 基于精英引导的社会学习 PSO 算法

2.1 Cubic 立方混沌初始化

种群初始粒子位置的散布程度及其在搜索空间中的均匀覆盖程度, 将直接影响整个算法的收敛速度和寻优效率^[20]。在面向大范围搜索空间和小样本粒子种群时随机初始化策略表现较为乏力, 相较而言, 混沌强烈的无规则非线性运动具有良好的遍历性和随机性^[33]。因此, 本文利用图 1 中均匀性更好的 Cubic 混沌映射赋予粒子无规则非线性运动状态, 有效保证了初始化粒子位置的随机程度, 提高了粒子初始化在解空间内的均匀覆盖程度, 提升了算法初始阶段的搜索效率^[34]。具体而言, Cubic 立方混沌映射公式为

$$\mathbf{y}_{i+1,j} = \rho \cdot \mathbf{y}_{i,j} (1 - \mathbf{y}_{i,j}^2) \quad (5)$$

式中, ρ 为混沌控制参数, 所产生的变量值 $\mathbf{y}_i = (y_{i,1}, y_{i,2}, \dots, y_{i,D})$ 为随机在 $[-1, 1]^D$ 内生成的 D 维向量。当 $\rho = 2.595, y_0 = 0.3$ 时, Cubic 映射具有较好的混沌遍历性, 具备在一定范围内对所有状态不重复遍历的能力。在此基础上, 将 Cubic 映射产生的变量值映射到种群个体并作为初始粒子位置

$$\mathbf{X}_i^1 = \mathbf{X}_{lb} + \frac{(\mathbf{X}_{ub} - \mathbf{X}_{lb})(\mathbf{y}_i + 1)}{2} \quad (6)$$

式中, $\mathbf{X}_{ub}$ 与 $\mathbf{X}_{lb}$ 表示粒子初始化的上界和下界。

2.2 精英引导的社会学习机制

如前所述, 传统的粒子群优化算法进化机制仅利用单个粒子的 p_{best} 和全局种群的 g_{best} 进行平行四边形矢量加法运算来更新每个粒子, 不能有效利用其他更有前途的粒子或者种群有效信息来提高进化

效果和缓解早熟收敛问题^[35]。针对这一缺陷,本文引入一种精英粒子引导的社会学习机制,在向适应度占优的粒子学习的同时,把整个群体的演化信息也拓展为粒子学习部分,更加全面地利用了粒子进化信息。

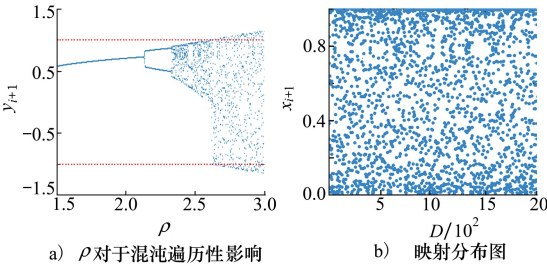


图 1 Cubic 混沌映射示意图

首先,为了提高群体算法高效优化求解的鲁棒性,兼顾单模态优化问题和复杂密集的多模态优化问题上算法种群规模的合理设置,保证在问题维数较高的情况下相应扩大种群规模^[36],将生成种群的大小设置成为关于搜索维度的函数,表示为

$$N = M + \lfloor D/10 \rfloor \tag{7}$$

式中: M 为基本种群规模; D 为初始种群的维数; $\lfloor \cdot \rfloor$ 为下取整函数。由初始粒子计算适应度值,并在此基础上粒子进行排序和进化学习运动。

按照粒子适应度表现对粒子从劣到优排序以区分精英粒子与平民粒子,并根据粒子表现设定学习概率赋予粒子不同的学习意愿。具体而言,适应度表现越好的粒子向其他较优粒子学习的能力越弱。另外,为了保证粒子在应对高维度、变量耦合的优化问题时保持更好的多样性,需要使得问题维度升高时粒子学习意愿降低。因此,粒子采用以下概率学习进化

$$P_i^t = \left(1 - \frac{i-1}{N}\right)^{\phi \cdot \log_2(\lceil M/D \rceil)} \tag{8}$$

式中, $\lceil \cdot \rceil$ 为上取整函数,常数 $\phi < 1$ 与 $\log_2(\cdot)$ 函数用于平滑 $\lceil M/D \rceil$ 。显然,粒子学习概率与粒子排序编号呈反比,适应度越优粒子的社会学习能力越弱;当处理大规模高维复杂问题时,减小粒子学习以保持更好的种群多样性。

在更新机制中,每一个粒子都要向优质的示范体粒子学习,但粒子学习的方向只能是趋于提升种群的适应度表现。为赋予粒子学习机制的随机性,随机学习概率 $P_i(t) \in (0,1)$ 位于区间 $[0, P_i^t]$ 时才采用社会学习策略更新粒子^[25]。

$$\mathbf{X}_{i,j}^{t+1} = \begin{cases} \mathbf{X}_{i,j}^t + \Delta \mathbf{X}_{i,j}^{t+1}, & P_i^t < P_i^L \\ \mathbf{X}_{i,j}^t, & \text{其他} \end{cases} \tag{9}$$

式中, $\Delta \mathbf{X}_{i,j}^{t+1}$ 为学习偏移量定义为

$$\Delta \mathbf{X}_{i,j}^{t+1} = r_1^t \cdot \Delta \mathbf{X}_{i,j}^t + r_2^t \mathbf{I}_{i,j}^t + r_3^t \varepsilon \mathbf{W}_{i,j}^t \tag{10}$$

式中,收缩因子 r_1^t, r_2^t 和 r_3^t 为每一次迭代中产生的 $[0,1]$ 范围内的随机数, $\varepsilon = \eta \times D/M$ 为集体行为学习因子; η 为学习率,按照经验取值为 0.01 时在各维度下算法能取得较好收敛效果。其中 $\mathbf{I}_{i,j}^t$ 和 $\mathbf{W}_{i,j}^t$ 具体为:

$$\begin{cases} \mathbf{I}_{i,j}^t = \mathbf{X}_{k,j}^t - \mathbf{X}_{i,j}^t \\ \mathbf{W}_{i,j}^t = \mathbf{C}_j^t - \mathbf{X}_{i,j}^t \end{cases} \tag{11}$$

式中, $\mathbf{C}_j^t$ 为当前种群中所有粒子第 j 维的平均位置,更广泛地利用了样本集合提供的种群集体信息; $k \in (i, N]$ 表示种群中相对于第 i 个粒子的较优粒子。

如图 2 所示,更新后的粒子位置的确定因素可以划分为三部分:粒子历史经验 $\mathbf{X}_{i,j}^t$ 、精英粒子 k 的学习行为 $\mathbf{X}_{k,j}^t$ 、种群集体行为 $\varepsilon \cdot \mathbf{W}_{i,j}(t)$ 。 $\Delta \mathbf{X}_{i,j}^t$ 与规范粒子群中的惯性分量类似,传递粒子的历史经验信息;精英粒子引导的学习行为打破了大多数粒子群单一学习群体最优个体的局限性,使得粒子每一维度随机学习当前种群中的范例($\mathbf{X}_{k,j}^t$, 其中 $i \leq k < m$),避免过度依赖于当前种群的最优粒子,有效增强了粒子进化的多样性;此外,种群的平均行为对所有粒子诱发群体级的从众现象,将 $\mathbf{C}_j^t$ 作为种群集散度的反映从而更加全面地挖掘种群信息,有效缓解早熟收敛的问题。

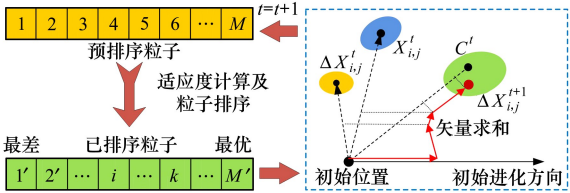


图 2 精英粒子引导的社会学习机制

综上所述,在自适应调整种群规模的基础上,通过图 2 中的循环机制,更好利用了种群学习的有效信息,增加了粒子进化进程中状态迭加的不稳定性,以达到算法收敛和粒子多样性之间的良好平衡,能够在不需要额外微调参数的情况下稳健地执行各种搜索维度尺寸的问题。

2.3 极值扰动因子

实验表明,种群中的粒子在进化进程时会停滞在粒子极值本身或者种群极值处,在高密度的搜索飞行之后难以突破局部极值,往往导致粒子陷入局部最优影响最终的优化求解效率^[37-38]。然而单纯增加种群样本数量仍然难以打破饱和限制,需要从搜索方式上摆脱粒子聚集到自身极值和群体全局极值决定的极值停滞点,增强粒子的进化潜力。

为此,在粒子搜索飞行进化过程中添加迁移机制激励粒子经历新的搜索路径和区域,通过随机扰动来摆脱在局部极值的停滞局面,降低早熟收敛现象发生的概率^[18],从而再次恢复种群的多样性、增强种群的可进化潜力。本文将粒子在当前最优位置的进化停滞步数 t 作为极值扰动触发条件,同时对 $p_{best,i,j}^t$ 与 $g_{best,j}^t$ 进行随机扰动,并在学习偏移量中补充极值扰动算子。 $\mathbf{I}_{i,j}^t$ 和 $\mathbf{W}_{i,j}^t$ 增加极值扰动算子后的具体形式为

$$\begin{cases} \mathbf{I}_{i,j}^t = r_4^{t_e > T_e} \mathbf{X}_{k,j}^t - \mathbf{X}_{i,j}^t \\ \mathbf{W}_{i,j}^t = r_5^{t_m > T_m} \mathbf{C}_j^t - \mathbf{X}_{i,j}^t \end{cases} \quad (12)$$

式中: t_e 与 t_m 分别表示精英粒子和群体学习行为的进化停滞步数; T_e 与 T_m 分别为精英粒子和群体学习行为增加扰动的进化停滞步数阈值。 $r_4^{t_e > T_e}$ 与 $r_5^{t_m > T_m}$ 分别表示带条件的 $U(0,1)$ 分布的随机函数,有

$$\begin{aligned} r_4^{t_e > T_e} &= \begin{cases} 1 & t \leq T_e \\ u(0,1) & t > T_e \end{cases} \\ r_5^{t_m > T_m} &= \begin{cases} 1 & t_m \leq T_m \\ u(0,1) & t_m > T_m \end{cases} \end{aligned} \quad (13)$$

由于均匀随机函数 r_4 和 r_5 的影响,演化过程中各学习分量的长度略有变化。

3 实验验证与对比分析

为了充分验证 ESLPSO 算法的优化性能和稳定性,将该算法与一些先进的 PSO 改进算法在基准函数测试集上进行了一系列比较。文中所有仿真实验均在 Interl Core i7 2.60 GHz CPU, 7 GB 运行内存的 64 位 Windows 10 的计算机上运行,并使用 Matlab 2019a 软件进行实验验证。为消除随机影响,所有实验结果均为 30 次独立重复运行后的平均值。

3.1 基准函数和对比算法

表 1 中总结了本文用于算法测试的 12 个基准函数,涵盖了许多不同类型的问题。表 2 中给出了

基准测试函数,包括函数名称、搜索空间和范围、特征。其中单峰函数(U)只有一个局部最优或者一个全局最优,常被用来测试粒子进化的开发能力;多峰函数(M)与多峰旋转函数(R)容易使探索性能乏力的算法陷入局部最优。所有算法运行的终止条件均为最大评价次数(MaxFEs)。

表 1 基准函数测试集

No.	函数名	搜索空间	特征
1	Sphere	$[-100,100]^D$	U
2	Schwefel 2.22	$[-10,10]^D$	U
3	Rosenbrock	$[-10,10]^D$	U
4	Schwefel 2.26	$[-500,500]^D$	M
5	Rastrigin	$[-5.12,5.12]^D$	M
6	Ackley	$[-32,32]^D$	M
7	Griewank	$[-600,600]^D$	M
8	Penalized1	$[-50,50]^D$	M
9	Penalized2	$[-50,50]^D$	M
10	Rotated Ackley	$[-32,32]^D$	R
11	Rotated Griewank	$[-600,600]^D$	R
12	Step	$[-100,100]^D$	U

最后,文中选取了其他 7 种 PSO 算法与本文方法进行性能比较,具体的算法来源和参数设置如表 2 所示。

表 2 8 种算法的参数设置

算法名称	参数设置
Breed PSO ^[39]	$\omega = 0.9 \sim 0.4, c_1 = c_2 = 1.494\ 45,$ $b_c = 0.8, b_s = 0.1$
XPSO ^[35]	$[\mu_1, \mu_2, \mu_3] = [1.35, 1.35, 1.35], \sigma = 0.1,$ $p = 0.5, \eta = 0.2, S_{GB} = 10, S_{max} = 5$
ASAPSO ^[37]	$\omega = 0.9 \sim 0.4$
AHAPSO ^[38]	$\omega = 0.8 \sim 0.2, c_1 = c_2 = 2.5 \sim 1.25,$ $\mu = 0.95, \gamma = 1\%, p_h = 0.8, p_b = 0.8$
SLPSO ^[40]	$\alpha = 0.5, \beta = 0.01$
AWPSO ^[20]	$a = 0.000\ 035 \times m, b = 0.5, c = 0,$ $d = 1.5, c_1 = c_2 = 1.494\ 45, \omega = 0.8 \sim 0.4$
DNSPSO ^[24]	$\xi(k) = \{1, 2, 3, 4\}, \omega = 0.5 \times E_f + 0.4$
ESLPSO	$\phi = 0.5, \eta = 0.01, \beta = 1.5,$ $T_e = 50, T_m = 50$

仿真实验中所有的基准函数的种群大小 M 均为 100,搜索空间的维度 D 为 30。每次运行的最大

函数计算次数 (MaxFEs) 最大迭代次数为 5 000。

3.2 求解精度比较及分析

为了全面评估所提出的 ESLPSO 的收敛精度,表 3 给出了所有算法在基准函数上运行 30 次后的

平均值 (mean)、最优解 (beat value) 和标准差 (Std. Dev.) 的信息,并粗体显示最优的结果数值。其中标准差反映了结果之间的不一致性程度,可以通过检查标准差来衡量算法搜索的稳健性^[20]。

表 3 运行结果对比

函数	准则	Breed PSO	XPSO	ASAPSO	AHAPSO	SLPSO	AWPSO	DNSPSO	ESLPSO
f_1	Mean	1.09×10^{-3}	6.09×10^{-181}	8.52×10^{-80}	6.53×10^{-112}	1.37×10^{-121}	2.66×10^{-2}	9.82×10^{-100}	0.00
	Best value	2.73×10^{-4}	2.08×10^{-229}	1.93×10^{-141}	1.36×10^{-149}	5.90×10^{-118}	9.94×10^{-3}	7.00×10^{-104}	0.00
	Std. Dev.	1.31×10^{-3}	0.00	2.69×10^{-79}	1.60×10^{-111}	1.06×10^{-117}	1.27×10^{-2}	1.85×10^{-99}	0.00
f_2	Mean	1.42×10^{-1}	5.25×10^{-75}	7.81×10^{-26}	1.46×10^{-54}	4.00×10^{-66}	2.30×10^{-1}	3.42×10^{-50}	2.65×10^{-160}
	Best value	3.20×10^{-2}	1.53×10^{-85}	6.90×10^{-63}	1.49×10^{-66}	5.09×10^{-73}	1.30×10^{-1}	1.27×10^{-51}	8.05×10^{-171}
	Std. Dev.	3.23×10^{-3}	1.36×10^{-74}	1.59×10^{-25}	3.13×10^{-54}	1.05×10^{-65}	7.19×10^{-2}	5.72×10^{-50}	1.96×10^{-161}
f_3	Mean	3.36×10^1	2.85	2.69×10^1	2.65×10^1	5.68	4.72×10^1	8.68×10^{-2}	1.07
	Best value	2.84×10^1	2.44×10^{-3}	2.68×10^1	2.85×10^1	1.48×10^{-2}	2.82×10^1	6.04×10^{-2}	6.16×10^{-2}
	Std. Dev.	1.94×10^1	3.35	6.18×10^{-3}	1.11×10^{-1}	9.97	2.99×10^1	1.83×10^{-2}	1.75
f_4	Mean	5.97×10^3	3.17×10^3	1.62×10^3	1.88×10^3	1.63×10^3	6.88×10^3	4.23×10^3	1.29×10^3
	Best value	5.00×10^3	2.47×10^3	1.28×10^3	1.28×10^3	1.28×10^3	6.14×10^3	2.28×10^3	8.29×10^2
	Std. Dev.	6.76×10^2	3.95×10^2	2.74×10^2	4.96×10^2	2.74×10^2	6.21×10^2	1.34×10^3	1.29×10^3
f_5	Mean	3.32×10^1	1.58×10^1	1.44×10^1	1.66×10^1	1.95×10^1	2.76×10^1	2.47×10^2	0.00
	Best value	1.89×10^1	3.83	0.00	0.00	1.29×10^1	2.19×10^1	2.08×10^2	0.00
	Std. Dev.	9.20	3.83	1.39×10^1	1.00×10^1	2.74	3.19	2.92×10^1	0.00
f_6	Mean	3.60	6.86×10^{-15}	1.47×10^{-14}	7.99×10^{-15}	6.93×10^{-15}	5.91×10^{-1}	2.66×10^{-15}	0.00
	Best value	3.22	6.22×10^{-15}	2.66×10^{-15}	8.88×10^{-16}	2.66×10^{-15}	9.22×10^{-2}	2.66×10^{-15}	0.00
	Std. Dev.	3.44×10^{-1}	2.14×10^{-15}	2.70×10^{-14}	8.52×10^{-15}	3.89×10^{-15}	4.83×10^{-1}	0.00	0.00
f_7	Mean	2.92×10^{-5}	1.11×10^{-16}	0.00	0.00	0.00	3.63×10^{-4}	0.00	0.00
	Best value	7.40×10^{-6}	1.11×10^{-16}	0.00	0.00	0.00	1.61×10^{-4}	0.00	0.00
	Std. Dev.	2.31×10^{-5}	2.60×10^{-32}	0.00	0.00	0.00	1.09×10^{-4}	0.00	0.00
f_8	Mean	6.70×10^{-1}	1.57×10^{-32}	1.57×10^{-32}	1.57×10^{-32}	1.57×10^{-32}	3.21×10^{-1}	2.58×10^{-5}	1.57×10^{-32}
	Best value	6.49×10^{-3}	1.57×10^{-32}	1.57×10^{-32}	1.57×10^{-32}	1.57×10^{-32}	1.57×10^{-32}	1.24×10^{-5}	1.57×10^{-32}
	Std. Dev.	5.69×10^{-1}	0.00	0.00	0.00	0.00	7.60×10^{-4}	9.43×10^{-6}	0.00
f_9	Mean	2.68	1.35×10^{-32}	1.17×10^{-2}	1.29×10^{-2}	1.57×10^{-32}	1.58×10^{-1}	3.34×10^{-4}	1.35×10^{-32}
	Best value	4.43×10^{-1}	1.35×10^{-32}	1.35×10^{-32}	1.35×10^{-32}	1.57×10^{-32}	6.40×10^{-2}	1.76×10^{-4}	1.35×10^{-32}
	Std. Dev.	3.40	0.00	1.17×10^{-2}	1.81×10^{-2}	0.00	1.42×10^{-1}	1.16×10^{-4}	0.00
f_{10}	Mean	4.00	7.998×10^{-15}	3.30×10^{-1}	2.49×10^{-1}	6.81×10^{-15}	2.16	3.73×10^{-15}	8.88×10^{-16}
	Best value	3.10	7.99×10^{-15}	8.88×10^{-16}	8.88×10^{-16}	4.44×10^{-15}	1.36	8.88×10^{-16}	8.88×10^{-16}
	Std. Dev.	6.38×10^{-1}	1.66×10^{-30}	7.38×10^{-1}	5.56×10^{-1}	1.83×10^{-15}	6.02×10^{-1}	1.59×10^{-15}	0.00
f_{11}	Mean	2.96×10^{-2}	2.77×10^{-3}	0.00	6.12×10^{-2}	3.20×10^{-3}	1.72×10^{-1}	0.00	0.00
	Best value	9.49×10^{-3}	0.00	0.00	0.00	0.00	1.44×10^{-1}	0.00	0.00
	Std. Dev.	2.39×10^{-2}	7.83×10^{-3}	0.00	8.49×10^{-2}	4.20×10^{-3}	1.71×10^{-2}	0.00	0.00
f_{12}	Mean	2.28×10^1	0.00	0.00	0.00	0.00	5.50	0.00	0.00
	Best value	1.40×10^1	0.00	0.00	0.00	0.00	2.00	0.00	0.00
	Std. Dev.	8.70	0.00	0.00	0.00	0.00	3.39	0.00	0.00

从表 3 可以看出,本文算法针对除 f_3 外所有测试函数的搜索精度都优于其他算法,表现出了最好的性能。无论是针对单峰函数还是多峰函数,ESLPSO 函数搜索结果的均值与标准差显示 ESLPSO

具有更加优秀的搜索精度和稳定性。对于单峰函数而言,ESLPSO 优于以往优化算法数十倍。面对多峰函数时,ESLPSO 大幅提升搜索结果精度,甚至得到精确最优解。同时,所提算法在解决多峰旋转问

题时也展现出显著优势。总而言之,本文所提出的改进方法可以有效提高算法收敛精度与搜索结果的稳健程度。

3.3 收敛性能比较分析

虽然 DNSPSO 在 f_3 与 f_{11} 上表现出了出色的性能,但 ESLPSO 也仅次于他收敛到最优值,并且其优势并不显著。观察图 3d),面对搜索空间极大、峰值极多且分布极不规则的 Schwefel 2.26 测试函数,并且这个函数的各个局部最优解的差值极小、寻优难度

较大,所有算法均陷入了局部最优,但 ESLPSO 的在收敛速度和求解精度方面的优势突出。

为了更加直观地对比不同算法在精度上的差别以及迭代速度,图 3 种绘制了 30 维下算法对各基准函数的收敛曲线。其中纵坐标为指数形式的平均适应度值,横坐标为迭代次数。从图中可以看到,ESLPSO 算法在大多数基准函数上表现出具有竞争力的性能。尤其是面对简单的单峰函数,ESLPSO 算法的收敛更快、具有更令人满意的平均适应度值。

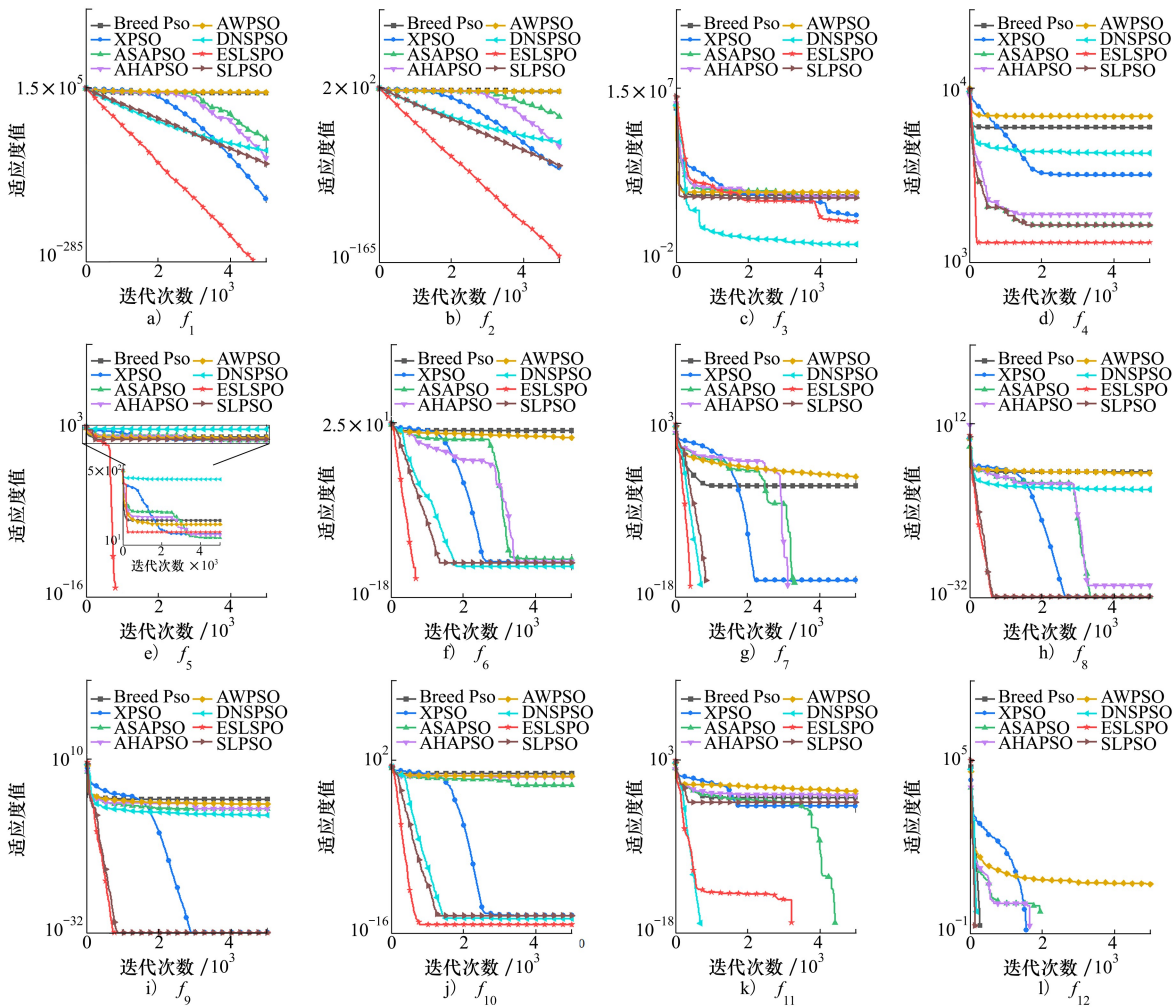


图 3 算法收敛性表现

如图 3e)~3j) 所示,大部分算法在迭代中期都陷入局部最优,跳出局部最优解的能力随搜索进程衰减。但本文所提出的 ESLPSO 算法融合了扰动算子,增加了状态迭加的不确定性,驱动种群逃脱了局部极小值,从而更接近最优结果,获得了更高效和成功的结果。综上所述,所提 ESLPSO 算法在收敛速

度上也具有一定的优势。

3.4 种群多样性比较分析

探索 and 开发能力的协调决定了算法的全局搜索性能,对粒子在进化进程中种群多样性变化的分析将更深层次地反映算法中粒子搜索时的收敛状态。种群多样性度量可计算为

$$\Omega(t)=\frac{1}{N}\sum_{i=1}^N\sqrt{\sum_{j=1}^D(X_{i,j}^t-\bar{X}_j^t)^2}\tag{14}$$

式中, X_j^t 在第 t 次迭代第 j 维对所有粒子的平均值, 也即 $\bar{X}_j^t=(1/N)\sum_{i=1}^NX_{i,j}^t$ 。多样性曲线走势实质上反映了种群的搜索状态的变化,反映出种群演化进程中开发、探测、收敛和跳出 4 个基本种群状态。本文提出的 ELSPSO 和其他 6 种 PSO 变体的种群多样性如图 4 所示,其中纵坐标以指数形式表示蜂群的多样性度量,横坐标表示迭代次数。

由图 4 可以看出,ELSPSO 算法和其他 PSO 算法在优化过程前期都具有较大的种群多样性值。如图 4a)~4b) 所示,随着算法搜索进程的不断推进,所有算法的种群多样性都随着迭代有不同程度收敛。

虽然 BreedPSO,AWSPSO 以及 DNSPSO 算法的多样性曲线也有类似的起伏现象,但是其本质上都期望通过种群多样性调整提升算法的全局搜索能力。对于所提出的 ESLPSO 算法,正是由于扰动机制和学习机制的持续作用使得 ELSPSO 在整个收敛进程中多样性曲线呈现出显著的波动特性,在探索和开发之间达成较好的折中。

从图 4e)~4g)与图 4k),图 4l)可以看出,当种群搜索到最优值时,多样性下降到并保持在一定数值水平时,扰动机制使得种群在收敛到最优值时仍然具备一定水平的多样性程度。尤其是在图 4h)及图 4i)中,算法陷入到局部最优而搜索停滞的情况下,ELSPSO 实现了对种群多样性的调整,种群多样性跳跃攀升到一定水平以维持种群的探索能力,赋予了种群跳出局部最优的能力和更好的求解精度。

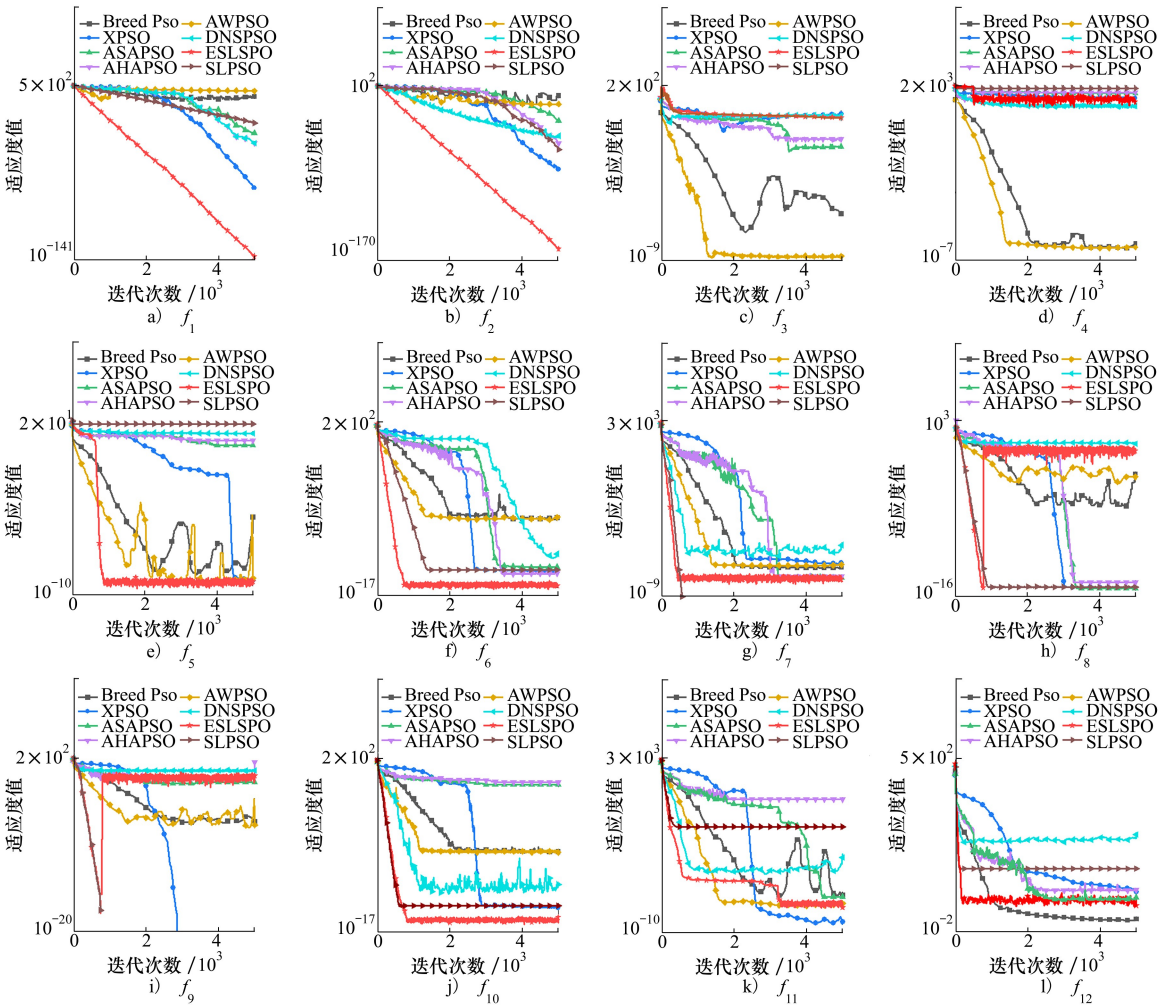


图 4 算法种群多样性表现

统筹分析图 3~4, 当基准算法由于早期收敛而陷入局部极小值时, ESLPSO 算法总能逃脱局部极小值, 并逼近多模态函数的最优解。特别是本文所提算法与 SLPSO 算法测试结果表明, 本文设计的耦合分层机制能够有效提升算法的优化求解性能。因此, 可以得出结论, 提出的 ESLPSO 算法在种群多样性表现上仍然具有一定有优势。

4 结 论

本文提出了 ESLPSO 算法, 以期提高经典粒子群优化算法的搜索能力。在 ESLPSO 中, 群体首先通过 Cubic 混沌初始化提高初始粒子在搜索空间的均匀覆盖程度。并将种群按照适应度值表现重新构造为具有分层拓扑结构的粒子组织, 并引入一种新颖的种群进化机制。

首先, 将适应度表现较优的粒子作为种群精英粒子, 其余粒子则作为平民粒子。平民粒子采用精

英引导的社会学习机制以加强群体引导, 改良了粒子最优状态和全局粒子最优状态引导的局限性。在此基础上引入极值扰动因子, 增加平民粒子态叠加的不确定性, 改善了最优粒子滞塞于局部最优的现象, 丰富了种群的多样性。为了验证 ESLPSO 算法的有效性, 本文利用了 12 个常用的优化基准函数 (包括单峰, 无旋转多峰和旋转多峰函数) 对 8 个 PSO 改进算法分别进行了求解精度、收敛分析以及种群多样性分析。实验结果表明: ESLPSO 显著提高了 PSO 的性能, 并在大多数优化问题上给出了最好的性能, 在提高问题求解精度和收敛速度方面具有一定的竞争力, 特别是在复杂的多模态函数上, 具有出色的综合性能。

未来的目标是进一步提高 ESLPSO 算法的收敛速度和求解精度。此外将着重把所提算法应用在自组织 RBF 神经网络^[40]、系统工程^[41-42]和信号处理^[43-44]等更复杂的实际优化问题中。

参考文献:

- [1] SHEN C, ZHANG K. Two-stage improved grey wolf optimization algorithm for feature selection on high-dimensional classification [J]. *Complex & Intelligent Systems*, 2021, 8(4): 2769-2789
- [2] OU X F, WU M, PU Y Y, et al. Cuckoo search algorithm with fuzzy logic and Gauss-Cauchy for minimizing localization error of WSN[J]. *Applied Soft Computing*, 2022, 125: 109211
- [3] MONTALVO I, IZQUIERDO J, PÉREZ R, et al. A diversity-enriched variant of discrete PSO applied to the design of water distribution networks[J]. *Engineering Optimization*, 2008, 40(7): 655-668
- [4] GONG Y J, ZHANG J, CHUNG S H, et al. An efficient resource allocation scheme using particle swarm optimization[J]. *IEEE Trans on Evolutionary Computation*, 2012, 16(6): 801-816
- [5] ZHANG H, XIE J, GE J, et al. An entropy-based PSO for DAR task scheduling problem[J]. *Applied Soft Computing*, 2018, 73: 862-873
- [6] GOLDBERG D E. *Genetic algorithms in search, optimization, and machine learning*[M]. Boston: Adotton-Wesley Professional, 1989
- [7] YANG J, YANG S, NI P. A vector tabu search algorithm with enhanced searching ability for pareto solutions and its application to multi-objective optimizations[J]. *IEEE Trans on Magnetics*, 2016, 52(3): 1-4
- [8] MARTINS D, MENDES M. A Hybrid evolutionary approach applied to the economic dispatch problem with prohibited operating zones and uncertainties[J]. *Latin America Transactions*, 2021, 19(7): 1225-1232
- [9] ZHAN Z H, ZHANG J, YUN L, et al. Adaptive particle swarm optimization[J]. *IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2009, 39(6): 1362-1381
- [10] LIU Q, DU S, WYK B J V, et al. Niching particle swarm optimization based on euclidean distance and hierarchical clustering for multimodal optimization[J]. *Nonlinear Dynamics*, 2020, 99(1): 1-19
- [11] KENNEDY J, EBERHART R. Particle swarm optimization[C]//*International Conference on Neural Networks*, 1995
- [12] EBERHART R, KENNEDY J. A new optimizer using particle swarm theory[C]//*Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995: 39-43
- [13] SHI Y. Parameter selection in particle swarm optimization[J]. *Evolutionary Programming*, 1998, 1447: 591-600
- [14] CAO Y, ZHANG H, LI W, et al. Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions[J]. *IEEE Trans on Evolutionary Computation*, 2019, 23(4): 718-731

- [15] YU W, LI B, WEISE T, et al. Self-adaptive learning based particle swarm optimization[J]. Information Sciences, 2011, 181(20): 4515-4538
- [16] YU F, TONG L, XIA X W. Adjustable driving force based particle swarm optimization algorithm[J]. Information Sciences, 2022, 609: 60-78
- [17] SHI Y, EBERHART R. A modified particle swarm optimizer[C]//1998 IEEE International Conference on Evolutionary Computation, 1998: 69-73
- [18] CLERC M, KENNEDY J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space[J]. IEEE Trans on Evolutionary Computation, 2002, 6(1): 58-73
- [19] RATNAWEERA A, HALGAMUGE S K, WATSON H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. IEEE Trans on Evolutionary Computation, 2004, 8(3): 240-255
- [20] LIU W, WANG Z, YUAN Y, et al. A novel sigmoid-function-based adaptive weighted particle swarm optimizer[J]. IEEE Trans on Cybernetics, 2021, 51(2): 1085-1093
- [21] RUI M, KENNEDY J, NEVES J. The fully informed particle swarm: simpler, maybe better[J]. IEEE Trans on Evolutionary Computation, 2004, 8(3): 204-210
- [22] LIANG J J, QIN A K, SUGANTHAN P N, et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J]. IEEE Trans on Evolutionary Computation, 2006, 10(3): 281-295
- [23] YE W, FENG W, FAN S. A novel multi-swarm particle swarm optimization with dynamic learning strategy[J]. Applied Soft Computing, 2017, 61: 832-843
- [24] ZENG N Y, WANG Z D, LIU W B, et al. A dynamic neighborhood-based switching particle swarm optimization algorithm[J]. IEEE Trans on Cybernetics, 2020, 50(1): 1-12
- [25] CHENG R, JIN Y. A social learning particle swarm optimization algorithm for scalable optimization[J]. Information Sciences, 2015, 291: 43-60
- [26] LI W, MENG X, HUANG Y, et al. Multi-population cooperative particle swarm optimization with a mixed mutation strategy[J]. Information Sciences, 2020, 529: 179-196
- [27] 黄懿, 梁放驰, 范成礼, 等. 带随机变异及感知因子的粒子群优化算法[J]. 西北工业大学学报, 2023, 41(2): 428-438
HUANG Yi, LIANG Fangchi, FAN Chengli, et al. Particle swarm optimization algorithm with random mutation and perception factor[J]. Journal of Northwestern Polytechnical University, 2023, 41(2): 428-438 (in Chinese)
- [28] KUMAR N, VIDYARTHI D P. A novel hybrid PSO-GA meta-heuristic for scheduling of DAG with communication on multiprocessor systems[J]. Engineering with Computers, 2016, 32(1): 35-47
- [29] JAVIDRAD F, NAZARI M. A new hybrid particle swarm and simulated annealing stochastic optimization method[J]. Applied Soft Computing, 2017, 60: 634-654
- [30] THARWAT A, ELHOSENY M, HASSANIEN A E, et al. Intelligent Bézier curve-based path planning model using chaotic particle swarm optimization algorithm[J]. Cluster Computing, 2019, 22(4): 1-22
- [31] ZHANG R, CHANG P C, SONG S, et al. Local search enhanced multi-objective PSO algorithm for scheduling textile production processes with environmental considerations[J]. Applied Soft Computing, 2017, 61(1): 447-467
- [32] WANG H M, QIAO Z W, XIA C L, et al. Self-regulating and self-evolving particle swarm optimizer[J]. Engineering Optimization, 2015, 47(1): 129-147
- [33] JAMES F. Chaos and randomness[J]. Chaos Solitons & Fractals, 1995, 6: 221-226
- [34] ZHANG M, LONG D, QIN T, et al. A chaotic hybrid butterfly optimization algorithm with particle swarm optimization for high-dimensional optimization problems[J]. Symmetry, 2020, 12(11): 1800
- [35] XIA X W, GUI L, HE G L, et al. An expanded particle swarm optimization based on multi-exemplar and forgetting ability[J]. Information Sciences: an International Journal, 2020, 508: 105-120
- [36] VESTERSTROM J, THOMSEN R. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems[C]//Proceedings of the 2004 Congress on Evolutionary Computation, 2004: 1980-1987
- [37] 闫群民, 马瑞卿, 马永翔, 等. 一种自适应模拟退火粒子群优化算法[J]. 西安电子科技大学学报, 2021, 48(4): 8
YAN Qunmin, MA Ruiqing, MA Yongxiang, et al. Adaptive simulated annealing particle swarm optimization algorithm[J]. Journal of Xidian University, 2021, 48(4): 8 (in Chinese)
- [38] 路复宇, 童宁宁, 冯为可, 等. 自适应杂交退火粒子群优化算法[J]. 系统工程与电子技术, 2022, 44(11): 3470-3476
LU Fuyu, TONG Ningning, FENG Weike, et al. Adaptive hybrid annealing particle swarm optimization algorithm[J]. Systems Engineering and Electronics, 2022, 44(11): 3470-3476 (in Chinese)
- [39] JIANG J, LI X, DENG Z H, et al. Control-oriented dynamic model optimization of steam reformer with an improved optimization

- algorithm[J]. International Journal of Hydrogen Energy, 2013, 38(26): 11288-11302
- [40] RAN C, JIN Y, et al. A social learning particle swarm optimization algorithm for scalable optimization[J]. Information Sciences an International Journal, 2015, 291: 43-60
- [41] HAN H, WU X, ZHANG L, et al. Self-organizing RBF neural network using an adaptive gradient multiobjective particle swarm optimization[J]. IEEE Trans on Cybernetics, 2019, 49(1): 69-82
- [42] LIU S, WANG Z, WEI G, et al. Distributed set-membership filtering for multi-rate systems under the round-robin scheduling over sensor networks[J]. IEEE Trans on Cybernetics, 2020, 50(5): 1910-1920
- [43] ZOU, LEI, WANG, et al. Recursive filtering for time-varying systems with random access protocol [J]. IEEE Trans on Automatic Control, 2019, 64(2): 720-727
- [44] ZOU L, WANG Z, HAN Q L, et al. Moving horizon estimation for networked time-delay systems under round-robin protocol[J]. IEEE Trans on Automatic Control, 2019, 64(12): 5191-5198

A novel elite guidance-based social learning particle swarm optimization algorithm

QI Cheng¹, XIE Junwei¹, WANG Xue², FENG Weike¹, ZHANG Haowei¹

(1. Air and Missile Defense College, Airforce Engineering University, Xi'an 710051, China;
2. School of Information Engineering, Binzhou Vocational College, Binzhou 256600, China)

Abstract: To improve the premature convergence and poor global search capability of the classical particle swarm algorithm (PSO), this paper proposes a novel elite guidance-based social learning particle swarm optimization (ESLPSO) algorithm. In the ESLPSO algorithm, a hierarchical topological search method is proposed. The method divides particles into optimal elite particles and other civilian particles according to their fitness performance, revolutionizing the update sample of the traditional population iterative search and enhancing the guidance of the whole population evolution information. In addition, an elite particle-guided social learning strategy is designed to better utilize the multidimensional information on population evolution by increasing the uncertainty of state superposition. On this basis, the extremum perturbation migration mechanism motivates the particles to experience new search paths and regions, increase population diversity and balance the population's exploration and exploitation in the search process. Moreover, the Cubic chaos initialization is employed to endow the initial particle population with a wide coverage in the search space. Finally, 12 benchmark test function sets covering unimodal, multimodal and rotated-multimodal functions are used to validate the performance of the proposed algorithm. The results on comparing the ESLPSO algorithm with other eight improved PSO algorithms show that the ESLPSO algorithm has excellent search performances in solving different types of functions, having efficient robustness and excellent solutions.

Keywords: particle swarm optimization; social learning; Cubic chaos; extremum perturbation

引用格式: 齐钺, 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 Cheng, XIE Junwei, WANG Xue, 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)