

混合均值中心反向学习粒子群优化算法

孙 辉^{1,2,3}, 邓志诚¹, 赵 嘉^{1,2,3}, 王 晖^{1,2,3}, 谢海华¹

(1. 南昌工程学院信息工程学院, 江西南昌 330099; 2. 鄱阳湖流域水工程安全与资源高效利用国家地方联合工程实验室, 江西南昌 330099; 3. 江西省水信息协同感知与智能处理重点实验室, 江西南昌 330099)

摘 要: 为平衡粒子群算法勘探与开发能力, 本文提出混合均值中心反向学习粒子群优化算法. 算法将所有粒子和部分优质粒子分别构造的均值中心进行贪心选择, 得出的混合均值中心将对粒子所在区域进行精细搜索. 同时对混合均值中心进行反向学习, 使粒子能探索更多新区域. 将本文算法与最新改进的粒子群算法、人工蜂群算法和差分算法在多种测试函数集上进行比较, 实验结果验证了混合均值中心反向学习策略的有效性, 算法的综合优化性能更强.

关键词: 全局寻优; 混合均值中心; 反向学习; 粒子群优化算法

中图分类号: TP182; TP391

文献标识码: A

文章编号: 0372-2112 (2019)09-1809-10

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

DOI: 10.3969/j.issn.0372-2112.2019.09.001

Hybrid Mean Center Opposition-Based Learning Particle Swarm Optimization

SUN Hui^{1,2,3}, DENG Zhi-cheng¹, ZHAO Jia^{1,2,3}, WANG Hui^{1,2,3}, XIE Hai-hua¹

(1. School of Information Engineering, Nanchang Institute of Technology, Nanchang, Jiangxi 330099, China;

2. National-Local Joint Engineering Laboratory of Water Engineering Safety and Effective Utilization of Resources in Poyang Lake Area, Nanchang, Jiangxi 330099, China; 3. Jiangxi Province Key Laboratory of Water Information Cooperative Sensing and Intelligent Processing, Nanchang, Jiangxi 330099, China)

Abstract: In order to balance the exploration and exploitation of particle swarm optimization, this paper proposes a hybrid mean center opposition-based learning particle swarm optimization. The algorithm performs greedy selection on the mean center of all particles and some high-quality particles respectively, and the obtained hybrid mean center will search the region in detail where the particles are located. At the same time, the hybrid mean center is using opposition-based learning, so that the particles can explore more new regions. The proposed algorithm are compared with the latest improved particle swarm optimization, artificial bee colony algorithm and difference algorithm in various test function sets, and the results verify the effectiveness of the hybrid mean center opposition-based learning and the overall optimization performance of the algorithm is stronger.

Key words: global optimization; hybrid mean center; opposition-based learning; particle swarm optimization

1 引言

为解决传统优化算法无法解决的复杂优化问题, 研究者们受启发于生物和自然物理现象, 提出了粒子群优化算法(Particle Swarm Optimization, PSO)、差分进化算法(Differential Evolution, DE)、人工蜂群算法(Arti-

ficial Bee Colony, ABC)、蚁群算法(Ant Colony Optimization, ACO)等优化技术以解决各领域的优化问题.

粒子群优化算法是由 Kennedy 和 Eberhart 在 1995 年提出的群智能优化算法^[1]. 该算法是源于对鸟群捕食行为的模仿研究, 人们从鸟群捕食过程当中得到启示, 并将其用于解决优化问题. 该算法通过群体中粒子

收稿日期: 2019-01-06; 修回日期: 2019-03-15; 责任编辑: 蓝红杰

基金项目: 国家自然科学基金(No. 61663029, No. 51669014, No. 61663028); 江西省杰出青年基金(No. 2018ACB21029); 江西省杰出青年人才资助计划(No. 20171BCB23075); 江西省自然科学基金(No. 20171BAB202035); 江西省教育厅落地计划基金(No. KJLD13096); 江西省 2018 年度研究生创新专项资金项目(No. YC2018 - S422)

间合作与竞争产生的群体智能指导优化搜索. 因具备计算快速和易实现等优点, 粒子群优化算法受到众多关注与研究, 且应用于解决图像处理^[2], 工程设计^[3]等众多领域的优化问题.

但是, 粒子群优化算法存在的多样性差、易早熟收敛等问题制约了粒子群优化算法的发展, 为克服上述问题, 众多学者做了大量研究. Zhao 等人提出将种群分为正常和早熟两种状态, 当种群在正常状态, 按标准 PSO 公式进化; 在早熟状态则引入矢量高斯学习策略^[4]. Hakll 等人采用列维飞行作为随机扰动步长, 当种群更新停滞次数达到预设值, 则使粒子绕最优值做列维飞行; 否则, 按标准更新公式更新种群^[5]. Yan 等人结合随机学习和列维飞行策略, 在标准速度更新公式中增加向随机粒子学习项, 当随机参数达到预设值, 则按改进速度公式更新, 否则按标准速度公式^[6]. Chen 等人提出先随机初始化种群, 再对粒子进行反向学习, 取最优粒子作为初始种群, 并使惯性权重, 加速因子服从正余弦函数变化, 以此提高收敛速度^[7]. 上述研究采用随机扰动策略或反向学习策略来提高种群逃离局部最优能力, 并取得一定效果, 但没能更好的平衡种群的勘探与开发能力.

本文提出混合均值中心反向学习粒子群优化算法 (Hybrid Mean Center Opposition-Based Learning Particle Swarm Optimization, HCOPSO). 首先, 构造混合均值中心, 其可引导种群在解空间进行更精细的搜索, 寻得更高精度的解. 针对种群易陷入局部最优, 引入反向学习策略, 生成混合均值中心反向解, 其可为种群逃离局部最优提供更多步长增量, 引导种群粒子探索更多新区域. 混合均值中心与反向学习结合, 可更好平衡种群的勘探与开发能力.

2 标准粒子群优化算法

设粒子数为 N 的种群, 在维度为 D 的空间中搜索, 粒子在解空间的位置为 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, $i = 1, 2, \dots, N$ 代表第 i 个粒子的位置, 而每个 x_i 都可由目标函数求得函数值 $f(x_i)$. $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, $i = 1, 2, \dots, N$ 代表第 i 个粒子的速度, 表示在每一次迭代中粒子移动的步长, $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, $i = 1, 2, \dots, N$ 代表第 i 个粒子所经历的个体最优位置, $g = (g_1, g_2, \dots, g_D)$, $g \in (1, 2, \dots, N)$ 代表全局最优位置. 速度和位置可按照以下公式更新:

$$v_{id}^{t+1} = wv_{id}^t + c_1r_1(p_{id}^t - x_{id}^t) + c_2r_2(g_d^t - x_{id}^t) \quad (1)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (2)$$

w 为惯性权重, c_1, c_2 为学习因子, r_1, r_2 为 $(0, 1)$ 随机数, p_{id}^t 为粒子 i 在第 t 次迭代时的历史最优位置, g_d^t 为第 t 次迭代时整个种群的历史最优位置.

3 混合均值中心反向学习粒子群优化算法

3.1 混合均值中心

粒子在搜索空间中都对对应适应值 $f(x)$. 在最优解所在小范围搜索空间内, 最优解位置对应的适应值为整个区域的峰值, 粒子离最优解位置越近, 其对应的适应值更优. 中心极限定理指出, 不论总体的分布情况如何, 大多统计量的分布近似正态分布; 正态分布在多数情况下可作为真实总体的近似值; 尽管实际数据与正态分布存在偏差, 但检验的统计量分布与正态分布的条件稳定, 偏差可忽略^[8]. 因此, 尽管实际数据不能严格服从多元正态分布, 但仍认为多维解空间服从多元正态分布.

根据正态分布 $N(\mu, \sigma^2)$ 随机变量的概率规律, 均值位置 μ 描述了正态分布的集中趋势, 其对应了正态分布的峰值. 根据粒子群优化算法特性, 种群粒子的目标是搜索并收敛于最优解位置, 即收敛于均值位置 μ . 当种群粒子在服从正态分布的空间中搜索时, 若粒子均匀的分布在均值 μ 两侧, 则种群各粒子的位置均值 \bar{x} 与解空间最优解对应的均值 μ 重合, 由于不可避免的模型误差, μ 和 \bar{x} 的位置或存在偏差. 但理论上, 均值 μ 的位置应与 \bar{x} 的位置重合. 因此, 本文构造粒子群体的均值中心 (Mean Center, MC), 其可用下式表示:

$$x_d^{MC} = \frac{1}{N} \left(\sum_{i=1}^N x_{id} \right) \quad (3)$$

$i = 1, 2, \dots, N$, N 为种群粒子总数, $d = 1, 2, \dots, D$ 为粒子维度, 式(3)表示均值中心的第 d 维度值, 为所有粒子在相同维度的算术平均值.

考虑到解空间并非总是具有严格的对称性, 而是有一定的偏斜, 呈现偏正态分布. 这时, 由种群所有粒子所得的位置均值 \bar{x} 受每一个数据的影响较大, 再用其描述整个种群的性质则不严谨. 因此, 本文以所有粒子适应值的均值 (Mean Value of Fitness, MVF) 为基准, 挑选优于 MVF 的粒子组成优质小群体, 构造偏均值中心 (Partial Mean Center, PMC). 由多元正态分布的性质可知, 多元正态随机向量的任何一个分量子集仍然服从正态分布. 本文把 D 维对应的所有较好粒子看成多元随机向量, 则每一维的较好粒子就是分量子集, 所以该优质小群体仍然服从正态分布. 偏均值中心所求过程如下.

首先求得群体平均适应值为:

$$\text{fitness}^{MVF} = \frac{1}{N} \sum_{i=1}^N \text{fitness}_i \quad (4)$$

fitness_i 表示第 i 个粒子的适应值, $i = 1, 2, \dots, N$. 再遴选种群中优于 MVF 的粒子记为 $n (1 \leq n \leq N)$, 偏均值中心的每一维度值可表示为:

$$x_d^{\text{PMC}} = \frac{1}{n} \left(\sum_{i=1}^n x_d \right) \quad (5)$$

其中 n 为优于 MVF 的粒子数,式(5)表示偏均值中心第 d 维度值,为所有优于 MVF 的粒子相同维度的算术平均值.

根据“*No Free Lunch*”理论,本文将 MC 与 PMC 优势结合,得出混合均值中心 (Hybrid Mean Center, HMC),可用下式表示:

$$\text{HMC} = \min(\text{MC}, \text{PMC}) \quad (6)$$

混合均值中心引导种群更新,可加快收敛速度.

3.2 算法基本原理

反向学习 (Opposition-Based Learning, OBL) 策略在 2005 年提出后,已成功应用于各类算法^[9],本文构造的混合均值中心位置,其位于种群的中心位置,由其引导种群进化时,对种群各粒子的步长增量更少,可加速粒子向中心靠拢,将大幅增强种群的局部搜索能力.当种群陷入局部最优位置时,所有粒子也将快速聚集在局部最优峰上.针对此问题,本文引入反向学习策略,对混合均值中心进行位置空间变换,生成混合均值中心反向解,使其跳出局部最优位置,同时也扩大了搜索区域,增强了种群的全局搜索能力.

在本文中,对混合均值中心反向, OBL_HMC 表示 HMC 的反向解,其可由式(7)而得:

$$\text{OBL_HMC}_d^t = r_3(a_d + b_d) - \text{HMC}_d^t \quad (7)$$

$$a_d = \min(\text{HMC}_d^t), b_d = \max(\text{HMC}_d^t) \quad (8)$$

式中, HMC_d^t 表示 HMC 第 t 次迭代时第 d 维的位置信息, r_3 为 $(0, 1)$ 间的随机数,其中 $\text{OBL_HMC}_d^t \in [a_d, b_d]$, a_d 为 HMC 在第 d 维的历史最小值, b_d 为第 d 维历史最大值. 根据混合均值中心的性质,对其进行反向学习有以下三点优势:

(1) 混合均值中心实质为各粒子的位置平均值,反应了各粒子的集中趋势,其到种群各粒子的距离比普通单个粒子到各粒子的距离更小. 因此,对混合均值中心反向学习,生成的反向解到种群各粒子的距离,比普通粒子生成的反向解到种群各粒子的距离更大,将为各粒子提供更多步长,引导种群探索更多区域.

(2) 混合均值中心实质为所有粒子位置信息的算术平均值,由平均值的数理性质可知,其对单个粒子位置的极端变化敏感,当采用反向学习策略生成新的距离较大位置时,混合均值中心能快速做出响应,生成成为相对差异大的新位置,使种群的多样性增强,收敛速度加快.

(3) 当种群粒子进入最优解所在区域,粒子向最优解聚集,混合均值中心整合粒子的有利位置信息,位于群体的中心位置,有望搜索到更高精度解.

3.3 混合均值中心策略分析

粒子数为 N 的群体,其均值中心记为 \bar{x} , $\bar{x} \in [x_{\min},$

$x_{\max}]$,对于均值中心 \bar{x} ,存在 $x_i \in \{x_1, \dots, x_N\}$,有

$$\bar{x} = x_m - x_i \quad (9)$$

x_m 为 $[x_{\min} + x_i, x_{\max} + x_i]$ 内的值. 参照文献[10]中的推导,

当 $x_i \neq \bar{x}$ 时, \bar{x} 比普通粒子 x_i 更靠近最优解 x^o 需满足

$$|\bar{x} - x^o| < |x_i - x^o| \quad (10)$$

因此

$$\begin{aligned} (\bar{x} - x^o)^2 - (x_i - x^o)^2 < 0 &\Rightarrow (\bar{x} + x_i - 2x^o)(\bar{x} - x_i) < 0 \\ &\Rightarrow (x_m - 2x^o)(x_m - 2x_i) < 0 \end{aligned} \quad (11)$$

显然

$$x_i < \frac{x_m}{2}, \text{ if } x^o > \frac{x_m}{2} \quad (12)$$

$$x_i > \frac{x_m}{2}, \text{ if } x^o < \frac{x_m}{2} \quad (13)$$

由此可知,当最优解 x^o 和粒子 x_i 在 $\frac{x_m}{2}$ 的不同侧时, \bar{x} 比 x_i 更靠近最优解 x^o .

选取粒子数为 10 的群体,在测试函数 Rastrigin 和 Schwefel 2.26 上进行测试,比较 MC 和 PMC 与整个粒子群体在每次迭代时成为最优粒子 (gbest) 的次数. 设置维度为 30,运行次数为 30 次,每次运行 20000 代,最终结果取 30 次的平均值,所得实验结果见图 1、图 2.

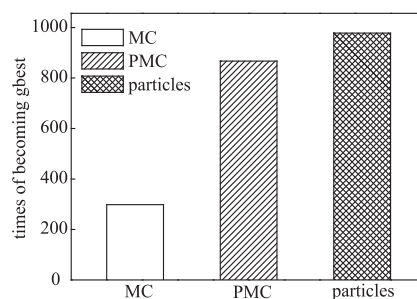


图1 Rastrigin函数测试结果

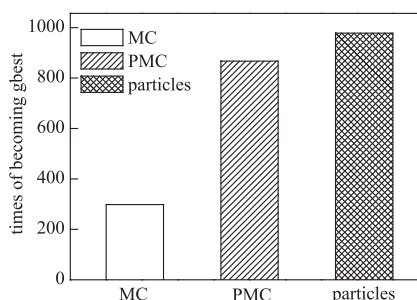


图2 Schwefel 2.26函数测试结果

在 Rastrigin 和 Schwefel 2.26 函数上, MC、PMC 成为最优粒子的次数占比较大. 由此可知, MC 和 PMC 为全局最优的更新做出重要贡献,且各有优势,将两者优势结合,可大幅提高全局最优的质量.

为进一步体现 HMC 的优势,在 12 个测试函数上进行测验,种群规模设置为 $N = 10$, 维度 $D = 30$, $c_1 = c_2 =$

2.0, 每个测试函数独立运行 30 次, 最终结果取 30 次的平均值, 每次运行评估次数为 20 万次, 所得实验结果见下表 1. HMC 在 7 个函数上有较大优势, 对表 1 中数据

做 Friedman 检验, HMC 的秩均值 (Ranks) 最小 (1.96), 验证了 HMC 更具优势.

表 1 三种均值中心的性能比较

| Fun | | PSO | MC | PMC | HMC |
|----------------|-----|------------------|-----------------|-----------------|-----------------|
| Sphere | MBF | 1.85E-29 | 2.24E-31 | 1.01E-42 | 6.64E-39 |
| Schewefel 2.22 | MBF | 3.93E-21 | 6.81E-23 | 4.98E-30 | 4.76E-28 |
| Schewefel 1.2 | MBF | 1.47E+01 | 3.23E+00 | 4.52E+00 | 4.56E+00 |
| Schewefel 2.21 | MBF | 5.23E-202 | 4.08E-197 | 8.97E-177 | 6.23E-188 |
| Rosenbrock | MBF | 3.34E+01 | 3.89E+01 | 3.29E+01 | 3.47E+01 |
| Step (f_6) | MBF | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Quartic | MBF | 1.54E-02 | 3.24E-03 | 2.39E-03 | 1.83E-03 |
| Schewefel 2.26 | MBF | 1.38E+03 | 1.41E+03 | 1.42E+03 | 1.35E+03 |
| Rastrigin | MBF | 2.40E+01 | 1.90E+01 | 1.94E+01 | 1.51E+01 |
| Ackley | MBF | 1.40E-14 | 1.04E-14 | 9.95E-15 | 9.95E-15 |
| Griewank | MBF | 1.77E-02 | 1.79E-02 | 1.03E-02 | 1.44E-02 |
| Penalized1 | MBF | 2.41E-02 | 3.46E-03 | 6.91E-03 | 3.46E-03 |
| Ranks | | 3.21(4) | 2.67(3) | 2.17(2) | 1.96(1) |

3.4 混合均值中心反向学习分析

粒子数为 N 的群体, 其均值中心记为 \bar{x} , 其反向学习所得反向解记为 \bar{x}^* , 另任选种群中某一粒子记为 x_n , $n \in (1, 2, \dots, N)$, 其反向解为 x_n^* . \bar{x} 反映了粒子的集中趋势, 种群中粒子位于或趋近该位置, 其对种群粒子的步长增量可表示为 $\sum_{i=1}^N |\bar{x} - x_i|$. x_n 为某单个粒子个体,

其对种群粒子的步长增量为 $\sum_{i=1}^N |x_n - x_i|$. 当种群进行局部搜索时, 种群粒子呈聚集态, \bar{x} 与种群多数粒子趋同, 一般情况下, 当 $\bar{x} \neq x_n$ 时, 两者对种群的步长增量关系满足下式:

$$\sum_{i=1}^N |\bar{x} - x_i| \leq \sum_{i=1}^N |x_n - x_i| \quad (14)$$

式(14)表明, \bar{x} 对种群粒子的步长增量更少, 其可引导种群对解空间进行更精细的搜索, 提高寻得最优解的精度, 但仅靠原种群粒子间的作用, 种群粒子易陷入局部最优. 本文引入反向学习策略, 变换搜索空间, 生成新的反向解, 以增强种群逃离局部最优的能力. 当对 \bar{x} 求反向解, \bar{x} 因与多数粒子位置趋同, 其反向解 \bar{x}^* 则与多数粒子位置存在较大距离. 同理, 当 $\bar{x} \neq x_n$, 其对应反向解 \bar{x}^* 和 x_n^* 对种群的步长增量则满足下式:

$$\sum_{i=1}^N |\bar{x}^* - x_i| \geq \sum_{i=1}^N |x_n^* - x_i| \quad (15)$$

因此, 混合均值中心生成的反向解, 可为种群粒子提供更多步长增量, 其比普通粒子反向学习对种群全局搜索能力的提升更大, 能进一步增强种群粒子逃离局部最优的能力.

全局搜索和局部搜索是种群进化的两个重要阶段, 全局搜索时, 种群为探索更广的搜索区域, 粒子的

位置较为分散; 局部搜索时, 种群对最优解区域进行精细搜索, 粒子位置相对集中. 因此本文引入粒子“聚集度 (aggregation)”来描述种群的进化过程:

$$\bar{x} = \frac{1}{ND} \sum_{i=1}^N \sum_{d=1}^D x_{id} \quad (16)$$

$$\text{aggregation} = \frac{1}{2bN\sqrt{D}} \sum_{i=1}^N \sqrt{\sum_{d=1}^D (x_{id} - \bar{x})^2} \quad (17)$$

其中, N 代表种群大小, D 代表粒子的维度, \bar{x} 为遍历种群所有粒子每一维元素求均值而得的位置基准值, 每个粒子以 \bar{x} 为基准的欧式距离为群体位置聚集度. $\text{aggregation} \in (0, 1)$, 当 aggregation 越大, 则表明种群粒子越分散, 反之, 则越集中. $2b$ 表示搜索空间范围.

全局最优粒子 (Global Best Position, GBP) 为种群的最好位置, 有时也位于群体的中心位置. 为验证本文采用 HMC 进行反向学习, 能更好的平衡种群全局勘探与局部开发能力, 将 HMC 进行反向的策略与 GBP 进行反向的策略进行对比, 两策略分别记为 OBL_HMC 和 OBL_GBP, 选取多峰函数 Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function^[11] 作为测试函数, 求两策略下的聚集度并进行收敛性分析. 种群规模 $N = 20$, 维度 $D = 30$, 评估次数设为 1500 次, 所得各策略下聚集度和收敛性如图 3 和 4 所示.

图 3 中, OBL_GBP 策略约 880 次评估前, aggregation 由大到小波动下降, 表明种群粒子位置由分散态转为聚拢态, 满足全局搜索过程搜索区域由大到小变化的特征, 可将此过程判定为全局搜索阶段. 在 880 次评估后, aggregation 将维持在 0.25 上下波动, 表明种群粒子集中在较小的区域内, 可将此过程判定为局部搜索阶段. 同理, OBL_HMC 策略的种群进化过程中, 约 700 次评估前, aggregation 由大到小波动下降, 此过程可判

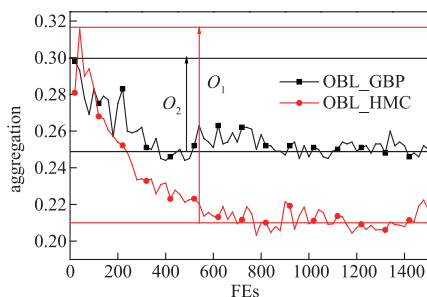


图3 函数聚集度测试

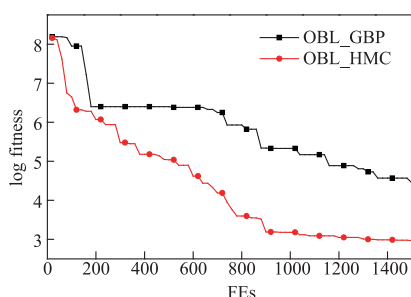


图4 函数收敛图

定为全局搜索阶段,700次评估后,aggregation将维持在0.21上下波动,可判定为局部搜索阶段。

将OBL_HMC与OBL_GBP比较,由图3可知,在全局搜索阶段,OBL_GBP的搜索范围可等价于 O_2 ,OBL_HMC的搜索范围等价于 O_1 ,显然有 $O_1 > O_2$,验证了OBL_HMC离种群各粒子位置的距离较大,能提供种群逃离局部最优的步长更多,对全局搜索能力的提升更大的结论。在局部搜索阶段,OBL_GBP的aggregation在0.25上下波动,而OBL_HMC的aggregation在0.21上下波动,表明OBL_HMC策略下,种群粒子位置更集中,验证了HMC策略下,种群可进一步对解空间精细搜索,局部搜索能力强的结论。此外,图4中对两策略进行了收敛性分析,实验结果表明,相比OBL_GBP策略,OBL_HMC策略下收敛速度更快,收敛精度更高,使用混合均值中心进行反向学习,比使用全局最优进行反向学习对算法优化性能提升更大。

3.5 算法流程

HCOPSO算法的运行流程如下所示:

HCOPSO 算法

Beginning

- Step1 设置学习因子、惯性权重、评估次数、解空间维度、种群规模等相关参数;
- Step2 对种群中所有粒子初始化;
- Step3 将种群粒子根据式(1)和式(2)更新飞行速度和位置;
- Step4 根据式(3)、(4)、(5)和(6)构造均值中心和偏均值中心,再将其两种优势结合,构造更具优势的混合均值中心参与种群

进化;

Step5 按照式(7)、(8)对混合均值中心进行反向学习,生成的反向解参与种群进化,得到更新后的全局最优解;

Step6 当达到最大评估次数时,输出最优解,否则返回Step3.

Ending

4 仿真实验

4.1 参数设置

HCOPSO算法的参数设置如下:种群规模 N 设置为20, c_1, c_2 设置为2.0,用 t 表示算法的当前迭代次数,Max_t表示算法的最大迭代次数,惯性权重为 $w = 0.55 \cdot \exp(-0.5 \cdot t/\text{Max}_t)$.

4.2 仿真实验及结果分析

将HCOPSO在维度 $D = 100$ 的条件下,与7种算法:LFPPO^[5], RPSOLF^[6], H-PSO-SCAC^[8], HPSO-TVAC, CLPSO, DMS-PSO, PSOLF^[12]比较.测试的函数参考文献[12]中的 $f_1 \sim f_{14}$,各算法的参数都按照原文献设置,评估次数设为50万次,所有算法独立运行30次,最终结果取30次的平均值,具体实验数据见表2所示。

本节表2中实验数据来源如下:CLPSO、DMS-PSO所测试函数 $f_1, f_2, f_3, f_4, f_6, f_7, f_8, f_9$ 和 f_{10} 的数据参考文献[13], $f_5, f_{11}, f_{12}, f_{13}$ 和 f_{14} 为实验所得. PSOLF在 f_1, f_2, f_3, f_4, f_6 和 f_8 的数据参考文献[12],其余函数数据为实验所得. HPSO-TVAC、LFPPO、RPSOLF、H-PSO-SCAC在100维数据为实验所得。

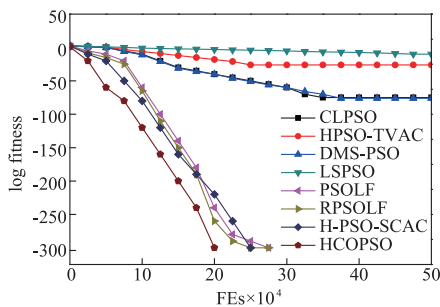
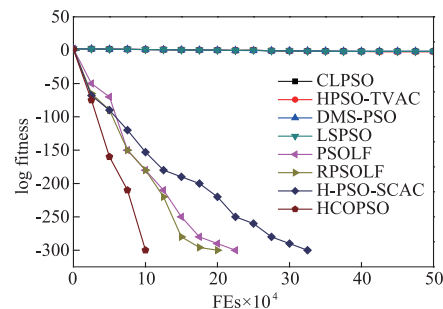
分析表2数据,得出以下结论:在相同评估次数下,HCOPSO在 $f_1, f_2, f_6, f_8, f_{12}, f_{13}$ 和 f_{14} 全部求得最优解,所测函数 f_9, f_{10} 同样求得比其它算法更高质量的解,测试 f_3, f_4, f_5 函数所得最优解相比其它算法优势明显. CLPSO在测试函数 f_5 时优势明显,其余函数时算法性能有待加强. HPSO-TVAC和DMS-PSO在 f_9, f_{10} 时优势明显,测试其余函数时,算法易陷入局部最优. PSOLF和RPSOLF在测试函数 $f_1, f_2, f_6, f_8, f_{12}, f_{13}$ 和 f_{14} 求得了函数最优解,但 f_9, f_{10} 时算法仍陷入局部最优. H-PSO-SCAC在函数 $f_1, f_2, f_6, f_8, f_{12}, f_{13}$ 和 f_{14} 同样能求得函数最优解,但在测试函数 f_9, f_{10} 时,算法仍陷入局部最优. 引入Friedman检验分析所得测试数据,表中HCOPSO算法相比其它7种改进的粒子群算法,所得秩均值最小,验证了本文将混合均值中心与反向学习结合后算法的综合性能更优。

4.3 收敛性分析

图5、6中,由于篇幅限制,在维度为100,评估次数50万次的条件下,选取单模函数 f_1 和旋转多模函数 f_{14} ,将HCOPSO算法与4.2节中7种算法进行比较。

表 2 100 维实验结果对比

| Fun | | CLPSO | HPSO-TVAC | DMS-PSO | LFPSO | PSOLF | RPSOLF | H-PSO-SCAC | HCOPSO |
|----------|-----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|------------------|-----------------|
| f_1 | MBF | 4.16E-75 | 5.48E-26 | 4.89E-77 | 2.65E-10 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 1.80E-74 | 2.59E-25 | 1.40E-76 | 3.89E-08 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_2 | MBF | 3.45E-45 | 3.77E-15 | 5.83E+02 | 1.02E+00 | 0.00E+00 | 1.22E-252 | 0.00E+00 | 0.00E+00 |
| | SD | 1.52E-44 | 1.20E-14 | 3.54E+01 | 3.21E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_3 | MBF | 1.46E+02 | 1.34E+02 | 1.20E+02 | 2.56E+01 | 9.04E+01 | 9.15E+01 | 9.27E+01 | 9.26E+01 |
| | SD | 4.78E+01 | 2.28E+02 | 3.62E+01 | 5.23E-01 | 2.57E+01 | 1.50E+00 | 4.31E+00 | 1.04E+00 |
| f_4 | MBF | 7.00E-03 | 1.17E-48 | 6.40E-03 | 3.21E-03 | 2.35E-05 | 8.68E-03 | 3.25E-125 | 2.22E-05 |
| | SD | 1.53E-03 | 1.27E-47 | 2.56E-03 | 3.24E-03 | 1.92E-05 | 5.47E-03 | 0.00E+00 | 1.50E-04 |
| f_5 | MBF | 5.36E+00 | 2.30E+03 | 6.58E+04 | 3.56E+04 | 2.46E+04 | 1.29E+03 | 2.66E+04 | 8.73E+03 |
| | SD | 2.35E+01 | 2.88E+03 | 5.21E+01 | 3.56E+03 | 5.75E+03 | 2.05E+03 | 4.11E+03 | 5.84E+03 |
| f_6 | MBF | 7.02E+00 | 4.90E+01 | 1.95E+01 | 3.25E+01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 1.00E+02 | 4.21E+01 | 2.59E+01 | 5.68E+01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_7 | MBF | 2.74E-14 | 3.21E-12 | 9.91E-01 | 3.87E-09 | 5.89E-16 | 5.89E-16 | 5.89E-16 | 5.88E-16 |
| | SD | 5.17E-15 | 4.29E-11 | 4.43E+00 | 5.89E-09 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_8 | MBF | 3.33E-17 | 2.71E-03 | 7.40E-04 | 3.89E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 7.29E-17 | 2.32E-02 | 2.28E-03 | 5.78E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_9 | MBF | 9.33E-03 | 2.39E-24 | 1.56E-02 | 9.56E-02 | 2.15E-02 | 1.27E-12 | 5.47E-01 | 8.58E-24 |
| | SD | 2.87E-02 | 7.19E-24 | 4.23E-02 | 3.46E-02 | 3.26E-02 | 1.24E-11 | 7.08E-01 | 9.27E-23 |
| f_{10} | MBF | 1.10E-03 | 1.74E-21 | 5.49E-03 | 8.97E-02 | 6.32E-03 | 3.25E-03 | 4.50E+00 | 3.90E-03 |
| | SD | 3.38E-03 | 7.05E-21 | 1.26E-02 | 2.65E-02 | 2.53E-02 | 9.57E-02 | 6.71E+00 | 9.54E-02 |
| f_{11} | MBF | 6.32E+03 | 2.85E+04 | 1.12E+04 | 3.56E+04 | 4.18E+04 | 3.43E+04 | 2.07E+04 | 1.26E+04 |
| | SD | 2.32E+02 | 5.14E+03 | 3.65E+03 | 3.25E+03 | 1.67E+02 | 3.57E+03 | 8.63E+02 | 7.92E+03 |
| f_{12} | MBF | 2.56E+03 | 5.47E+02 | 9.32E+02 | 1.25E+02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 5.32E+02 | 1.03E+02 | 1.35E+02 | 6.32E+01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_{13} | MBF | 6.87E-03 | 1.05E-06 | 3.56E-06 | 6.89E-06 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 3.98E-03 | 2.54E-06 | 3.89E-05 | 1.58E-05 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| f_{14} | MBF | 2.32E-02 | 3.89E-03 | 8.25E-01 | 9.58E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 1.25E-02 | 5.89E-03 | 9.23E-01 | 2.56E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Ranks | | 4.89(5) | 4.89(5) | 6.04(7) | 6.43(8) | 3.71(3) | 3.29(2) | 3.82(4) | 2.93(1) |

图5 Sphere(f_1)函数进化曲线图图6 Rotated Griewank(f_{14})函数进化曲线图

大部分算法在 f_1 函数上都能够求得较高精度的解, 但 HCOPSO 算法所用评估次数更少, 收敛速度最快. 在 f_{14} 函数上, HCOPSO、PSOLF、RPSOLF 和 H-PSO-SCAC 都能求得最优解, HCOPSO 收敛速度更快, 其余算法未求得最优解.

4.4 与相关新算法比较

为进一步验证 HCOPSO 算法的优势, 将其与基于反向学习策略的代表性改进粒子群优化算法 GOPSO、OPSO、NCOPSO^[14] 在 CEC 2013^[15] 函数集上比较. 评估次数设置为 10 万次, 维度 D 为 30, 各算

法独立运行 25 次, 最终结果取 25 次平均值, 其他算法的参数设置和实验数据参照文献[14], 所得实验结果见下表 3.

与 OPISO 算法相比, HCOPSO 算法在所测的 22 个函数上优势较大, 在 6 个测试函数上, HCOPSO 算法的优化性能低于 OPISO. 与 GOPSO 算法相比, HCOPSO 算法在 24 个测试函数占有优势, 在其余 4 个函数上, HCOPSO 算法的优化性能有待加强. 与 NCOPSO 算法相比, HCOPSO 算法在 14 函数占优, 在其余 14 个函数上, 两算法的优化性能接近.

表 3 CEC 2013 实验结果对比

| Fun | | OPSO | GOPSO | NCOPSO | HCOPSO |
|----------|-----|-----------------|-----------------|-----------------|-----------------|
| f_1 | MBF | 2.46E-04 | 9.36E-04 | 2.27E-13 | 2.73E-13 |
| f_2 | MBF | 3.26E+07 | 3.19E+07 | 3.65E+06 | 4.89E+06 |
| f_3 | MBF | 4.03E+09 | 6.20E+09 | 7.28E+07 | 1.89E+08 |
| f_4 | MBF | 3.47E+04 | 3.49E+04 | 2.05E+04 | 1.66E+04 |
| f_5 | MBF | 2.74E-02 | 2.22E-02 | 2.85E-03 | 1.18E-12 |
| f_6 | MBF | 1.16E+02 | 1.32E+02 | 1.08E+02 | 4.15E+01 |
| f_7 | MBF | 1.13E+02 | 1.16E+02 | 2.93E+01 | 5.90E+01 |
| f_8 | MBF | 2.10E+01 | 2.10E+01 | 2.10E+01 | 2.10E+01 |
| f_9 | MBF | 3.55E+01 | 3.41E+01 | 2.34E+01 | 2.75E+01 |
| f_{10} | MBF | 1.50E+01 | 3.73E+01 | 2.36E-01 | 7.30E-01 |
| f_{11} | MBF | 5.74E+01 | 5.38E+01 | 6.02E+01 | 4.23E+01 |
| f_{12} | MBF | 1.53E+02 | 1.44E+02 | 5.93E+01 | 8.39E+01 |
| f_{13} | MBF | 2.28E+02 | 2.31E+02 | 1.33E+02 | 1.66E+02 |
| f_{14} | MBF | 1.63E+03 | 1.52E+03 | 6.98E+03 | 1.19E+03 |
| f_{15} | MBF | 4.54E+03 | 4.68E+03 | 6.23E+03 | 6.12E+03 |
| f_{16} | MBF | 2.62E+00 | 2.46E+00 | 2.51E+00 | 2.37E+00 |
| f_{17} | MBF | 6.59E+01 | 6.92E+01 | 1.83E+02 | 8.26E+01 |
| f_{18} | MBF | 2.57E+02 | 2.86E+02 | 1.80E+02 | 2.43E+02 |
| f_{19} | MBF | 5.70E+00 | 5.61E+00 | 9.39E+00 | 3.84E+00 |
| f_{20} | MBF | 1.47E+01 | 1.45E+01 | 1.32E+01 | 1.22E+01 |
| f_{21} | MBF | 2.93E+02 | 3.16E+02 | 3.06E+02 | 3.11E+02 |
| f_{22} | MBF | 1.86E+03 | 2.04E+03 | 6.69E+03 | 1.18E+03 |
| f_{23} | MBF | 4.87E+03 | 5.26E+03 | 7.23E+03 | 5.62E+03 |
| f_{24} | MBF | 2.92E+02 | 2.96E+02 | 2.40E+02 | 2.74E+02 |
| f_{25} | MBF | 3.19E+02 | 3.20E+02 | 3.11E+02 | 2.91E+02 |
| f_{26} | MBF | 2.02E+02 | 2.02E+02 | 2.74E+02 | 2.58E+02 |
| f_{27} | MBF | 1.22E+03 | 1.22E+03 | 7.14E+02 | 9.82E+02 |
| f_{28} | MBF | 8.78E+02 | 1.09E+03 | 3.00E+02 | 5.99E+02 |
| Ranks | | 3.13(4) | 2.84(3) | 2.20(2) | 1.84(1) |

对表 3 数据所得实验数据进行 Friedman 检验, 所得结果中, HCOPSO 的所得秩均值评价指标最小 (1.84), 验证了 HCOPSO 算法的优化性能比其它三种使用反向学习的粒子群算法更好, 验证了采用混合均值中心进行反向学习对算法性能提升更大。

近年来, 在智能计算领域, 另提出了许多较有优势

表 4 与 8 种相关算法实验结果对比

| Fun | | BABC | CoDE | JADE | jDEscop | AABCLS | CMAES | ABCVSS | HCOPSO |
|-------|-----|----------|----------|----------|----------|----------|-----------------|----------|-----------------|
| f_1 | MBF | 1.01E-14 | 1.16E-37 | 2.50E-87 | 1.15E-36 | 4.60E-35 | 1.21E-28 | 6.68E-23 | 0.00E+00 |
| | SD | 5.07E-14 | 2.69E-37 | 8.62E-87 | 5.75E-36 | 1.67E-35 | 2.09E-29 | 3.16E-32 | 0.00E+00 |
| f_2 | MBF | 2.18E-06 | 1.21E-20 | 1.89E-41 | 1.71E-22 | 4.98E-18 | 3.69E-04 | 2.89E-18 | 0.00E+00 |
| | SD | 1.06E-05 | 8.94E-21 | 8.32E-41 | 8.46E-22 | 1.34E-18 | 1.84E-03 | 6.52E-18 | 0.00E+00 |
| f_3 | MBF | 6.29E-02 | 3.32E+01 | 3.19E-01 | 2.58E+01 | 3.09E+00 | 1.75E-25 | 1.09E-01 | 4.10E+01 |
| | SD | 1.24E-01 | 2.26E+01 | 1.10E+00 | 2.68E+01 | 1.36E+01 | 4.31E-26 | 2.52E-01 | 3.51E+00 |
| f_4 | MBF | 5.74E-02 | 8.17E-03 | 2.50E-03 | 3.91E-03 | 1.63E-02 | 2.80E-01 | 6.14E-02 | 3.20E-05 |
| | SD | 1.11E-02 | 2.79E-03 | 1.54E-03 | 1.47E-03 | 4.68E-03 | 6.70E-02 | 1.38E-02 | 1.74E-04 |
| f_5 | MBF | 3.78E-12 | 4.74E+00 | 4.75E+00 | 6.18E+01 | 1.41E-11 | 9.01E+03 | 4.66E-12 | 7.19E+02 |
| | SD | 7.28E-13 | 2.37E+01 | 2.32E+01 | 1.69E+02 | 3.53E-12 | 1.13E+03 | 2.47E-12 | 2.08E+03 |

的算法^[16], 且以成功应用于工程实践. 为进一步体现 HCOPSO 算法的性能, 将其与改进人工蜂群算法 BABC、AABCLS 和 ABCVSS, 改进差分算法 CoDE、JADE 和 jDEscop, 及高效的进化策略 CMAES 比较. 实验设置维度 $D=50$, 评估次数为 25 万次, 各函数参数设置参照原文献, 所得实验结果见下表 4, 表中各算法的数据和所测函数的 22 个函数参照文献[17].

从表 4 中数据可知, 在测试的 22 个函数中, BABC 算法在对函数的优化性能上, 在 $f_6, f_8, f_{15}, f_{17}, f_{20}, f_{21}$ 和 f_{22} 上, 7 个函数达到了最优解值, 在算法的稳定性上, 有 7 个函数达到最好. AABCLS 在对函数的优化性能上, 在 f_6, f_{15}, f_{21} 和 f_{22} 上有 4 个函数达到了最好值, 在算法的稳定性上, f_6, f_9, f_{15} 和 f_{29} 共 5 个函数达到最好. ABCVSS 在对函数的优化性能上, 在 $f_6, f_{15}, f_{17}, f_{20}, f_{17}$ 和 f_{20} 上有 6 个函数达到了最优值, 在算法的稳定性上, $f_6, f_{15}, f_{17}, f_{20}$ 和 f_{21} 共 5 个函数达到最好.

CoDE 在对函数的优化性能上, 在 $f_9, f_{15}, f_{16}, f_{19}$ 和 f_{21} 上有 5 个函数达到了最好值, 在算法的稳定性上, f_9, f_{15}, f_{19} 和 f_{21} 共 4 个函数达到最好. JADE 在对函数的优化性能上, 在 f_{15}, f_{16}, f_{19} 和 f_{21} 上, 有 4 个函数达到了最好值, 在算法的稳定性上, 在 f_{15}, f_{16} 和 f_{19} 上, 有 3 个函数达到最好.

jDEscop 在对函数的优化性能上, 有 5 个函数达到了最好值, 在算法的稳定性上, 有 2 个函数达到最好. CMAES 有 1 个函数达到了最好值, 在算法的稳定性上, 有 1 个函数达到最好.

HCOPSO 算法在 $f_1 \sim f_7, f_{11} \sim f_{13}, f_{18}, f_{20}$ 共 12 个函数上都寻得最优解, 且稳定性强. 在 f_9, f_{15} 函数上, HCOPSO 算法所得最优解质量, 算法的稳定性都优于其他算法. 在剩余的函数上, HCOPSO 算法总体优势明显. 将 8 种新相关算法与 HCOPSO 算法进行 Friedman 检验, 所得各算法的秩均值中, HCOPSO 算法秩均值最小 (3.61), 验证算法的综合优化性能更强.

续表

| Fun | | BABC | CoDE | JADE | jDEscop | AABCLS | CMAES | ABCVSS | HCOPSO |
|----------|-----|------------------|------------------|------------------|------------------|------------------|-----------|------------------|-----------------|
| f_6 | MBF | 0.00E+00 | 7.34E-01 | 1.78E-11 | 1.03E-14 | 0.00E+00 | 3.89E+02 | 0.00E+00 | 0.00E+00 |
| | SD | 0.00E+00 | 8.82E-01 | 1.74E-11 | 1.31E-14 | 0.00E+00 | 7.11E+01 | 0.00E+00 | 0.00E+00 |
| f_7 | MBF | 7.50E-15 | 2.81E-15 | 6.22E-15 | 1.43E+01 | 2.65E-14 | 1.99E+01 | 1.70E-14 | 5.89E-16 |
| | SD | 2.49E-15 | 7.11E-16 | 0.00E+00 | 6.50E+00 | 3.48E-15 | 2.63E-02 | 5.75E-33 | 0.00E+00 |
| f_8 | MBF | 0.00E+00 | 2.96E-04 | 7.88E-04 | 1.47E-16 | 4.44E-18 | 1.38E-03 | 3.67E-14 | 0.00E+00 |
| | SD | 0.00E+00 | 1.48E-03 | 2.82E-03 | 2.60E-16 | 2.22E-17 | 3.34E-03 | 1.84E-13 | 0.00E+00 |
| f_9 | MBF | 1.22E-13 | 9.42E-33 | 2.49E-03 | 1.91E-31 | 9.42E-33 | 4.98E-03 | 1.07E-32 | 3.25E-31 |
| | SD | 6.09E-13 | 1.40E-48 | 1.24E-02 | 5.28E-31 | 1.40E-48 | 1.72E-02 | 5.74E-33 | 4.92E-30 |
| f_{10} | MBF | 2.50E-15 | 1.55E-33 | 1.01E-04 | 5.02E-32 | 1.50E-33 | 8.01E+03 | 1.80E-33 | 8.89E-31 |
| | SD | 1.25E-14 | 2.47E-34 | 8.05E-05 | 5.95E-32 | 0.00E+00 | 6.25E+03 | 1.08E-33 | 2.14E-29 |
| f_{11} | MBF | 3.14E-29 | 1.25E-34 | 4.43E-78 | 4.77E-36 | 4.48E-33 | 4.88E-23 | 1.11E-23 | 0.00E+00 |
| | SD | 4.95E-29 | 1.50E-34 | 2.21E-77 | 1.85E-35 | 2.16E-33 | 9.20E-24 | 4.17E-23 | 0.00E+00 |
| f_{12} | MBF | 1.23E-13 | 1.94E-38 | 2.27E-85 | 1.80E-41 | 2.54E-35 | 4.22E-27 | 1.77E-33 | 0.00E+00 |
| | SD | 6.16E-13 | 4.08E-38 | 1.14E-82 | 1.47E-16 | 1.24E-35 | 9.12E-28 | 7.88E-33 | 0.00E+00 |
| f_{13} | MBF | 2.96E-92 | 6.20E-143 | 3.01E-100 | 3.58E-124 | 1.52E-50 | 6.82E-13 | 7.21E-41 | 0.00E+00 |
| | SD | 0.00E+00 | 3.15E-142 | 1.00E-99 | 1.79E-123 | 4.88E-50 | 5.39E-13 | 3.56E-40 | 0.00E+00 |
| f_{14} | MBF | 7.20E+00 | 6.00E-05 | 8.20E-10 | 2.52E+00 | 1.16E-01 | 6.00E-15 | 2.06E+00 | 0.00E+00 |
| | SD | 2.78E+00 | 9.51E-05 | 9.51E-10 | 5.61E-01 | 1.24E-01 | 7.45E-16 | 3.77E-01 | 0.00E+00 |
| f_{15} | MBF | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 1.60E-01 | 0.00E+00 | 0.00E+00 |
| | SD | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 4.73E-01 | 0.00E+00 | 0.00E+00 |
| f_{16} | MBF | 2.67E-109 | 2.67E-109 | 2.67E-109 | 2.67E-109 | 2.67E-109 | 5.40E-66 | 2.67E-109 | 6.00E-106 |
| | SD | 2.62E-116 | 9.67E-125 | 9.60E-125 | 9.65E-125 | 2.77E-119 | 1.99E-65 | 3.08E-120 | 1.02E-104 |
| f_{17} | MBF | 0.00E+00 | 2.28E+01 | 2.74E-08 | 8.02E-02 | 8.00E-02 | 3.78E+02 | 0.00E+00 | 0.00E+00 |
| | SD | 0.00E+00 | 4.68E+00 | 2.07E-08 | 2.77E-01 | 2.77E-01 | 4.90E+01 | 0.00E+00 | 0.00E+00 |
| f_{18} | MBF | 2.07E-16 | 4.39E-03 | 1.01E-04 | 2.95E-05 | 5.32E-11 | 8.59E-01 | 2.14E-16 | 0.00E+00 |
| | SD | 7.79E-16 | 6.44E-03 | 8.05E-05 | 3.09E-05 | 9.68E-11 | 8.49E-01 | 7.43E-16 | 0.00E+00 |
| f_{19} | MBF | 1.35E-31 | 1.35E-31 | 1.35E-31 | 1.30E-30 | 1.35E-31 | 3.77E-01 | 1.35E-31 | 1.76E-02 |
| | SD | 2.23E-47 | 2.47E-33 | 2.23E-47 | 3.69E-30 | 2.23E-47 | 1.22E+00 | 2.23E-47 | 2.01E-01 |
| f_{20} | MBF | 0.00E+00 | 3.45E+00 | 3.33E-01 | 0.00E+00 | 2.56E-06 | 9.41E+00 | 0.00E+00 | 0.00E+00 |
| | SD | 0.00E+00 | 2.94E-01 | 4.45E-02 | 0.00E+00 | 7.39E-06 | 4.19E+00 | 0.00E+00 | 0.00E+00 |
| f_{21} | MBF | -7.83E+01 | -7.83E+01 | -7.83E+01 | -7.83E+01 | -7.83E+01 | -6.43E+01 | -7.83E+01 | -7.07E+01 |
| | SD | 1.16E-14 | 4.10E-15 | 2.26E-01 | 1.12E-13 | 4.10E-15 | 2.63E+00 | 1.00E-14 | 8.04E+00 |
| f_{22} | MBF | -5.00E+01 | -4.86E+01 | -4.98E+01 | -5.00E+01 | -5.00E+01 | -4.10E+01 | -5.00E+01 | -4.12E+01 |
| | SD | 6.79E-06 | 4.45E-01 | 3.86E-02 | 9.23E-03 | 9.13E-04 | 2.65E+00 | 3.82E-07 | 1.03E+01 |
| Ranks | | 4.23(4) | 4.43(7) | 4.07(3) | 4.36(6) | 3.98(2) | 7.09(8) | 4.23(4) | 3.61(1) |

群智能算法经过的大量研究,绝大多数改进算法对较早版本的标准测试函数有较好优化效果.因此,本文采用新近提出的测试函数 CEC 2015^[11]进一步验证算法的优化性能. HCOPSO 算法分别与新近提出的萤火虫与粒子群混合优化算法 FFPSO, HPSOFF, HFPSO^[18]

比较,维度 $D=30$, 评估次数为 1500, 实验数据和其他参数设置参见文献[18].

表5为各算法测函数 CEC 2015 的实验结果, HCOPSO 算法在所测 15 个函数中,有 14 个函数占较大优势,对各算法进行 Friedman 检验, HCOPSO 算法所得

秩均值最小,再次验证了本文算法更具优势.

表 5 CEC 2015 实验结果对比

| Fun | | PSO | FA | FFPSO | HPSOFF | HFPSO | HCOPSO |
|----------|-----|------------|------------|------------|-------------------|------------|-------------------|
| f_1 | MBF | 3.9049E+09 | 2.8899E+10 | 9.2383E+10 | 4.7539E+09 | 1.1795E+09 | 7.8725E+07 |
| f_2 | MBF | 9.9760E+04 | 1.3418E+05 | 6.9430E+06 | 9.7376E+04 | 8.5653E+04 | 5.0939E+03 |
| f_3 | MBF | 3.3113E+02 | 3.3850E+02 | 3.4771E+02 | 3.3059E+02 | 3.2638E+02 | 3.1695E+02 |
| f_4 | MBF | 7.7928E+03 | 8.0330E+03 | 9.6696E+03 | 6.8199E+02 | 5.1202E+03 | 7.2189E+02 |
| f_5 | MBF | 5.0418E+02 | 5.0425E+02 | 5.0586E+02 | 5.0422E+02 | 5.0410E+02 | 5.0251E+02 |
| f_6 | MBF | 6.0096E+02 | 6.0410E+02 | 6.0755E+02 | 6.0090E+02 | 6.0076E+02 | 6.0038E+02 |
| f_7 | MBF | 7.0405E+02 | 7.6997E+02 | 8.9401E+02 | 7.0750E+02 | 7.0074E+02 | 7.0035E+02 |
| f_8 | MBF | 4.0013E+03 | 2.3491E+06 | 1.6032E+08 | 1.7191E+04 | 2.6354E+03 | 8.2272E+02 |
| f_9 | MBF | 9.1358E+02 | 9.1381E+02 | 9.1427E+02 | 9.1365E+02 | 9.1337E+02 | 9.1248E+02 |
| f_{10} | MBF | 7.5602E+06 | 2.9938E+07 | 3.9391E+08 | 1.1337E+07 | 5.4690E+06 | 8.6004E+05 |
| f_{11} | MBF | 1.1614E+03 | 1.2889E+03 | 2.1094E+03 | 1.1551E+03 | 1.1336E+03 | 1.1212E+03 |
| f_{12} | MBF | 2.2417E+03 | 2.9655E+03 | 4.9876E+05 | 2.0617E+03 | 1.7752E+03 | 1.3981E+03 |
| f_{13} | MBF | 1.7719E+03 | 1.9596E+03 | 3.6329E+03 | 1.7390E+03 | 1.6866E+03 | 1.6425E+03 |
| f_{14} | MBF | 1.6644E+03 | 1.7522E+03 | 2.1683E+03 | 1.6711E+03 | 1.6469E+03 | 1.6231E+03 |
| f_{15} | MBF | 2.5522E+03 | 2.8256E+03 | 3.9013E+03 | 2.6529E+03 | 2.4467E+03 | 2.1061E+03 |
| Ranks | | 6.00(6) | 5.00(5) | 3.47(4) | 3.40(3) | 2.07(2) | 1.07(1) |

5 结束语

为解决粒子群优化算法多样性差,早熟收敛等现象,本文主要做了两点改进:一是构造混合均值中心;二是采用反向学习策略,生成混合均值中心反向解,引导种群粒子探索更多搜索区域.在基准测试函数上与新改进粒子群优化算法进行 100 维的测试,且与其它新种类智能算法比较,并引入新测试函数集 CEC 2013 和 CEC 2015 测试算法优化性能,结果都表明 HCOPSO 算法优化性能更具优势.但是,HCOPSO 在高维情况下测试复杂函数,仍易陷入局部最优.因此,在下一步的研究工作中,将在各参数设置方式,和自适应处理等方面进行改进,进一步提升算法性能.

参考文献

- [1] KENNEDY J, EBERHART R. Particle swarm optimization [A]. Proceedings of International Conference on Neural Networks [C]. Piscataway: IEEE, 1995. 1942 - 1948.
- [2] 李奕, 吴小俊. 粒子群进化学习自适应双通道脉冲耦合神经网络图像融合方法研究 [J]. 电子学报, 2014, 42(2): 217 - 222.
LI Yi, WU Xiao-Jun. A novel image fusion method using self-adaptive dual-channel pulse coupled neural networks based on PSO evolutionary learning [J]. Acta Electronica Sinica, 2014, 42(2): 217 - 222. (in Chinese)
- [3] 俞海珍, 汪鹏君, 张会红, 万凯. 基于三值多样性粒子群算法的 MPRM 电路综合优化 [J]. 电子学报, 2017, 45(7): 1601 - 1607.
YU Hai-zhen, WANG Peng-jun, ZHAO Hui-hong, et al. Optimization of MPRM circuits based on ternary diversity particle swarm optimization [J]. Acta Electronica Sinica, 2017, 45(7): 1601 - 1607. (in Chinese)
- [4] ZHAO J, LV L, WANG H, et al. Particle swarm optimization based on vector Gaussian learning [J]. Ksii Transactions on Internet and Information Systems, 2017, 11(4): 2038 - 2057.
- [5] HAKLI H, UGUZ H. A novel particle swarm optimization algorithm with Levy flight [J]. Applied Soft Computing, 2014, 23: 333 - 345.
- [6] YAN B, ZHAO Z, ZHOU Y, et al. A particle swarm optimization algorithm with random learning mechanism and Levy flight for optimization of atomic clusters [J]. Computer Physics Communications, 2017, 219: 79 - 86.
- [7] CHEN K, ZHOU F, YIN L, et al. A hybrid particle swarm optimizer with sine cosine acceleration coefficients [J]. Information Sciences, 2018, 422: 218 - 241.
- [8] 朱蓉, 靳雁霞, 范卫华. 融合优质粒子分布的粒子群优化算法 [J]. 小型微型计算机系统, 2015, 36(3): 576 - 580.
ZHU Rong, JIN Yan-xia, FAN Wei-hua. Particle swarm optimization algorithm combination with the distribution of superior quality particles [J]. Journal of Chinese Computer Systems, 2015, 36(3): 576 - 580. (in Chinese)
- [9] WANG H, WU Z, RAHNAMEYAN S, et al. Enhancing particle swarm optimization using generalized opposition-based learning [J]. Information Sciences, 2011, 181(20): 4699 - 4714.
- [10] WANG H, WU Z, LIU Y, et al. Space transformation search: a new evolutionary technique [A]. Proceedings of Word Summit Genetic Evolutionary Computation [C]. New York: ACM, 2009. 537 - 544.
- [11] CHEN Q, LIU B, ZHANG Q, et al. Problem Definitions

- and Evaluation Criteria for CEC 2015 Special Session and Competition on Bound Constrained Single-Objective Computationally Expensive Numerical Optimization [R]. Zhengzhou: Zhengzhou University, 2014.
- [12] JENSI R, JIJI G W. An enhanced particle swarm optimization with levy flight for global optimization [J]. Applied Soft Computing, 2016, 43: 248 – 261.
- [13] OUYANG H, GAO L, LI S, et al. Improved global-best-guided particle swarm optimization with learning operation for global optimization problems [J]. Applied Soft Computing, 2017, 52: 987 – 1008.
- [14] 周凌云, 丁立新, 彭虎, 等. 一种邻域重心反向学习的粒子群优化算法 [J]. 电子学报, 2017, 45 (11): 2815 – 2824.
ZHOU Ling-yun, DING Li-xin, PENG Hu, et al. Neighborhood centroid opposition-based particle swarm optimization [J]. Acta Electronica Sinica, 2017, 45 (11): 2815 – 2824. (in Chinese)
- [15] LIANG J J, QU B Y, SUGANTHAN P N, et al. Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session on Real-Parameter Optimization [R]. Zhengzhou: Zhengzhou University, 2013.
- [16] 孙辉, 谢海华, 赵嘉. 正弦选择概率模型的全局最优引导人工蜂群算法 [J]. 南昌工程学院学报, 2018, 37 (6): 84 – 90.
SUN Hui, XIE Hia-hua, ZHAO Jia. Global optical guided artificial bee colony algorithm based on sinusoidal selection probability model [J]. Journal of Nanchang Institute of Technology, 2018, 37 (6): 84 – 90. (in Chinese)
- [17] CUI L, LI G, LIN Q, et al. A novel artificial bee colony algorithm with depth-first search framework and elite-guided search equation [J]. Information Sciences, 2016, 367: 1012 – 1044.

- [18] AYDILEK İ B. A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems [J]. Applied Soft Computing, 2018, 66: 232 – 249

作者简介



孙辉 男, 1959 年 3 月生, 江西九江人. 博士、二级教授. 主要研究方向为智能计算、Rough 集和粒计算、变分不等原理与变分不等式.

E-mail: sun_hui2006@163.com



邓志诚 男, 1995 年 2 月生, 江西丰城人. 硕士研究生. 主要研究方向为智能计算.



赵嘉 男, 1981 年 9 月生, 安徽桐城人. 教授、硕士生导师、中国电子学会高级会员. 主要研究方向为智能计算与大数据挖掘.

王晖 男, 1982 年 8 月生, 湖北红安人. 博士、教授、硕士生导师. 主要研究方向为智能计算与并行计算.

谢海华 男, 1994 年 2 月生, 江西萍乡人. 硕士研究生. 主要研究方向为智能计算.