



混沌反馈共享和群体协同效应的蝴蝶优化算法

李守玉¹, 何庆¹⁺, 杜逆索²

1. 贵州大学大数据与信息工程学院, 贵阳 550025

2. 贵州大学贵州省大数据产业发展应用研究院, 贵阳 550025

+ 通信作者 E-mail: qhe@gzu.edu.cn

摘要:针对蝴蝶优化算法(BOA)寻优精度低和易陷入局部最优等缺点,提出了混沌反馈共享和群体协同效应的蝴蝶优化算法(CFSBOA)。首先,利用Hénon混沌初始化种群,能够使种群尽可能地覆盖搜索盲区,增加种群多样性,提高算法寻优性能;其次,利用反馈控制电路中正负反馈作用机制的思想,构建蝴蝶之间的反馈共享交流网络,使得蝴蝶个体能够接收来自多个方向的信息,帮助种群定位最优解的位置并执行精细搜索,增强算法逃离局部最优的能力和加快算法收敛的速度;最后,利用群体协同效应机制,提高和平衡全局与局部搜索的能力,增强算法的全局和局部的寻优能力。使用不同维度的基准测试函数、统计检验、Wilcoxon检验及多类型的CEC2014部分函数验证改进蝴蝶优化算法的性能,并与新改进的蝴蝶算法及其他群智能算法进行对比,实验结果表明该算法具有明显优势。

关键词:蝴蝶优化算法(BOA);混沌映射;反馈共享;群体协同效应

文献标志码:A 中图分类号:TP301.6

Butterfly Optimization Algorithm for Chaotic Feedback Sharing and Group Synergy

LI Shouyu¹, HE Qing¹⁺, DU Nisuo²

1. College of Big Data & Information Engineering, Guizhou University, Guiyang 550025, China

2. Guizhou Big Data Academy, Guizhou University, Guiyang 550025, China

Abstract: A butterfly optimization algorithm (BOA) based on chaotic feedback sharing and group synergy (CFSBOA) is proposed to solve the shortcomings of low precision and easy to fall into local optimum. Firstly, using Hénon chaos to initialize the population can make the population cover the search blind area as much as possible, increase the diversity of the population, and improve the ability of optimizing the algorithm. Secondly, using the ideas of positive and negative feedback mechanism in feedback control circuit, it builds butterfly feedback shared communication network, allowing individuals to receive information from multiple directions to help populations of positioning the location of the optimal solution and perform careful search, enhance the ability to escape from local optimum and accelerate the algorithm convergence speed. Finally, the collective synergistic effect mechanism is used to improve and balance the global and local search ability and enhance the global and local optimization ability

基金项目:贵州省科技计划项目重大专项项目(黔科合重大专项字[2018]3002,黔科合重大专项字[2016]3022);贵州省公共大数据重点实验室开放课题(2017BDKFJJ004);贵州省教育厅青年科技人才成长项目(黔科合KY字[2016]124);贵州大学培育项目(黔科合平台人才[2017]5788)。

This work was supported by the Science and Technology Planning Project Major Special Project of Guizhou Province ([2018]3002, [2016]3022), the Open Project of Guizhou Key Laboratory of Public Big Data (2017BDKFJJ004), the Youth Science and Technology Talent Growth Project of Guizhou Provincial Department of Education ([2016]124) and the Guizhou University Cultivation Project ([2017]5788).

收稿日期:2020-11-30 **修回日期:**2021-01-25

of the algorithm. The performance of the improved butterfly optimization algorithm is verified by using different dimension benchmark test functions, statistical test, Wilcoxon test and multiple types of CEC2014 partial functions. Compared with the new improved butterfly algorithm and other swarm intelligence algorithms, the experimental results show that the proposed algorithm has obvious advantages.

Key words: butterfly optimization algorithm (BOA); chaotic mapping; feedback sharing; group synergy

为了更好地解决现实中的复杂优化问题,研究者们通过观察、研究自然界中生物的生理或物理现象,提出了粒子群优化算法、灰狼优化算法、樽海鞘优化算法、飞蛾扑火算法及蝴蝶优化算法等元启发式算法,并将它们用于解决现实中的工程优化任务。

传统的粒子群算法^[1]虽收敛速度快、易于实现,但存在早熟现象,蚁群算法^[2]虽有较强的记忆性,但它容易出现停滞现象且效率低。蝴蝶优化算法(butterfly optimization algorithm, BOA)^[3]是由 Arora 和 Singh 于 2019 年提出的基于全局优化的群智能算法。BOA 通过蝴蝶之间散发的香味形成一个具有一定信息损失的一般社会交互网络,其具有原理简单、参数少、计算耗时少及易于实现等优点,因此被应用于阴影光伏阵列的重构技能的工程设计^[4]、优化无线传感器网络的能量有效簇^[5]、小波神经网络的训练^[6]、家庭 CO₂减排策略混合智能预测模型^[7]及认知智能电网中基于能效优化的频谱分配策略^[8]等领域。

然而蝴蝶优化算法也存在收敛速度慢及寻优精度不高的缺陷。为此,研究者提出了不同的改进策略。文献[9]通过将柯西变异和自适应权重结合增强算法跳出局部最优的能力和全局搜索的能力,使用动态切换概率调节全局与局部搜索的比重。文献[10]利用混沌理论与正余弦算法结合的方式增强算法的寻优能力,使用线性递减控制香味的幂指数提高收敛精度。文献[11]通过交叉熵和协同进化技术找寻全局与局部搜索之间平衡,提高算法的全局搜索能力。文献[12]利用 limit 阈值克服陷入局部的难题,再将单纯形法和正弦余弦指引策略结合对位置进行更新,提高算法的寻优性能。文献[13]利用扰动策略和疯狂因子增加算法的多样性,通过线性递减惯性权重平衡全局与局部搜索的能力。尽管改进算法寻优精度有所提升,但收敛速度和稳定性仍有很大改进空间。

针对蝴蝶优化算法存在的问题,本文提出了混沌反馈共享和群体协同效应的蝴蝶优化算法(butterfly optimization algorithm based on chaotic feedback sharing and group synergy, CFSBOA)。算法利

用二维的 Hénon 映射,使得种群个体的搜索分布更加均匀,增加算法多样性;通过正反馈或负反馈网络加快算法的信息流动,进而增强逃离局部最优的能力;最后融合群体协同,进一步矫正和协调种群的进化方向及更好地平衡全局与局部搜索能力。通过求解 12 个基准测试函数和部分 CEC2014 测试函数验证算法的有效性和鲁棒性。

1 蝴蝶优化算法

在标准 BOA 算法中,蝴蝶种群采用随机初始化蝴蝶种群:

$$X_{N \times d} = lb + rand(N, d) \times (ub - lb) \quad (1)$$

式中, N 表示蝴蝶种群规模, d 表示搜索空间维度。

蝴蝶算法假定搜索空间中的每个个体能感知彼此散发的香味;蝴蝶能根据香味浓度的强弱进行随机移动或向最好蝴蝶移动;蝴蝶的感觉模态受目标函数的范围影响或决定。由于蝴蝶进行食物源搜索时,香味大小受风、雨、温度等因素影响和限制,采用转换概率 P 控制蝴蝶的搜索模式:全局搜索或局部搜索。

蝴蝶香味计算公式:

$$f = cI^a \quad (2)$$

$$c = 0.01 + \frac{0.025}{0.1T_{\max}} \quad (3)$$

其中, f 代表香味大小; c 是感官模态强度; I 是刺激强度; a 是依赖感官模态的幂指数; T_{\max} 表示最大迭代次数。

蝴蝶的位置更新由局部搜索和全局搜索构成。若蝴蝶因距离和自然因素无法感知到其他蝴蝶散发的香味,则蝴蝶在局部执行随机搜索;若蝴蝶能够感知到其他蝴蝶散发的香味,它能根据香味强弱,飞向最好蝴蝶实现蝴蝶位置更新,从而使得种群向食物源位置进行搜索。

当 $r_1 \geq P$ 时,进行局部搜索:

$$dis_1 = r_1^2 x_j^t - x_i^t \quad (4)$$

$$x_i^{t+1} = x_i^t + dis_1 f_i \quad (5)$$

其中, dis_1 为局部搜索中两只蝴蝶之间的距离公式;

x_j^t, x_i^t 表示第 j, i 只蝴蝶在第 t 次迭代的位置; f_i 表示第 i 只蝴蝶的香味大小。

当 $r_1 < P$ 时, 进行全局搜索:

$$dis_2 = r_2^2 g^* - x_i^t \quad (6)$$

$$x_i^{t+1} = x_i^t + dis_2 f_i \quad (7)$$

其中, dis_2 为全局搜索中当前蝴蝶和最好蝴蝶之间的距离, g^* 为最好蝴蝶的位置, x_i^t 是第 i 只蝴蝶在第 t 次迭代的位置。由式(5)和式(7)能够模拟蝴蝶种群的觅食行为。

2 混沌共享反馈和群体协同效应的蝴蝶优化算法

2.1 Hénon 初始化机制

标准 BOA 利用式(1)随机初始化蝴蝶位置, 可能会出现蝴蝶位置在搜索空间中分布不均匀, 导致算法过早陷入局部最优, 出现早熟现象。文献[14]利用 Tent 映射初始化樽海鞘种群, 使得樽海鞘个体均匀分布在搜索空间上, 从而提高算法寻优精度。

混沌映射具有遍历性、随机性和规律性等优点, 其基本思想利用映射关系产生 [0,1] 的混沌序列, 再将混沌序列转化到个体的搜索空间。常用来初始化种群的混沌映射有 Tent 映射、Logistic 映射, 但文献[15]指出 Tent 映射存在小周期和不稳定周期容易使 Tent 映射易陷入不动点; Logistic 映射在 [0,1] 范围内呈切比雪夫分布, 搜索盲区大。然而 Hénon 不仅能克服 Tent 映射和 Logistic 映射的缺点, 其产生的混沌序列也更加均匀, 如图 1 所示。利用 Hénon 的混沌序列分布更加均匀的特点, 种群可以最大程度覆盖搜索盲区, 增加蝴蝶个体的多样性, 从一定程度上帮助算法跳出

局部极值, 进而增强算法寻优的能力。因此, 本文采用 Hénon 混沌映射初始化蝴蝶种群。其数学描述:

$$\begin{cases} x(k) = 1 - \gamma(x(k-1))^2 + \varphi y(k-1) \\ y(k) = x(k-1) \end{cases} \quad (8)$$

$$z_i^k = \frac{x_i^k - \varphi_i}{\gamma_i - \varphi_i}, i = 1, 2, \dots, N \quad (9)$$

式(8)中, $\gamma = 1.4, \varphi = 0.3, x(0) = 0, y(0) = 0, k$ 代表混沌迭代次数, $k \leq N$ 。为了避免 Hénon 映射的不动点, 以当前最优解 $x_i^* = x_i^k, lb_i \leq x_i^* \leq ub_i$ 为初始值, 选取 n 个微小差异的初始值, 通过式(8)产生 [0,1] 区间上 n 个不同的混沌序列式(9), 最后通过逆映射得到个体搜索空间变量 x_i^t :

$$x_i^t = lb_i + z_i^k(ub_i - lb_i) \quad (10)$$

其中, z_i^k 是最优混沌变量, $[lb_i, ub_i]$ 是 x_i^t 搜索空间的上、下界。

2.2 反馈共享机制

由于标准 BOA 算法只通过全局最优解的位置进行信息交流, 很大程度上限制蝴蝶的搜索范围, 增大算法陷入局部最优的风险, 降低算法的寻优精度。为了增强算法跳出局部最优的能力, 参考电路分析中反馈控制电路的概念, 让蝴蝶受到其他蝴蝶的正反馈或负反馈的作用。具体表现受生物互惠互利的行为机制影响, 让蝴蝶受到比自身适应度好的个体的正反馈作用以及比自身适应度差的负反馈作用, 从而使得蝴蝶能够感知到更多方向的位置信息, 增强蝴蝶的全局搜索能力。

从搜索空间中随机选择一只蝴蝶, 与当前蝴蝶、最优蝴蝶构建香味反馈机制实现信息共享。在飞行过程中, 蝴蝶散发的香味会向不同方向传播, 为搜索

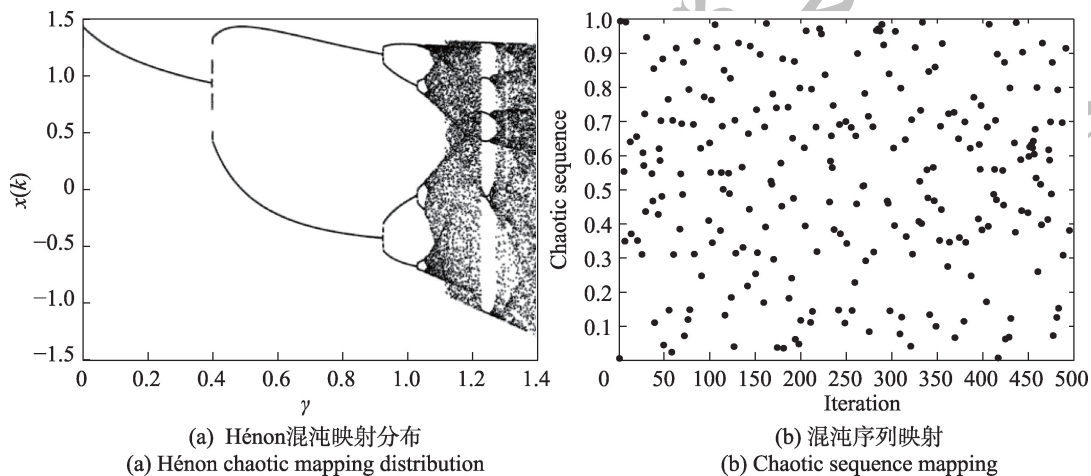


图1 Hénon 混沌映射
Fig.1 Hénon chaotic mapping

空间中其他蝴蝶寻优提供有利信息。因此,假设每只蝴蝶能向两个不同方向传播香味,然后决定它将飞向哪个方向,两个传播方向分别是随机蝴蝶所在方向和最优蝴蝶所在方向。通过香味反馈机制,当前蝴蝶根据传递的香味信息能够知道食物是否在这两只蝴蝶周围。若随机蝴蝶的适应度比当前的蝴蝶的适应度更好,则认为该食物存在,否则附近没有食物来源。

如果食物源被证实存在于两只蝴蝶的周围,通过正反馈作用,当前蝴蝶会移动到两只蝴蝶周围。如果不是,通过负反馈作用,它向最好的蝴蝶移动。数学模型描述如下:

$$\alpha = (F_{\max} - F_{\min})r_3 + F_{\min} \quad (11)$$

$$\theta = \frac{1}{1 + \exp(-\lambda)} \quad (12)$$

$$x_i' = \begin{cases} x_i' + (g^* - x_i')\alpha + (x_h' - x_i')\alpha + \theta, & F(x_h') < F(x_i') \\ x_i' + (g^* - x_i')\alpha + \theta, & F(x_h') \geq F(x_i') \end{cases} \quad (13)$$

式中, α 为共享系数; F_{\max} 、 F_{\min} 分别为种群最大、最小适应度值; r_3 是[0,1]之间的随机数; θ 是反馈因子,能够加速蝴蝶之间的信息反馈速率实现信息快速共享,其中 λ 表示当前迭代个体数; x_h' 表示随机蝴蝶的位置; F 表示计算适应度。

香味反馈机制使得蝴蝶个体可以更多地围绕最佳位置进行挖掘,同时式(13)提出的飞行方向使蝴蝶个体运动方向具有多样化的能力,可以增强探索能力,特别是在迭代的初始阶段,可以避免过早收敛。

2.3 群体协同效应位置更新机制

标准 BOA 中计算蝴蝶之间距离计算方式存在如下缺陷:(1)更新后的蝴蝶个体对前一次迭代的蝴蝶会产生很强的依赖性,具有一定盲目性致使算法出现停滞,从而导致收敛速度下降。(2)算法全局搜索和局部开发的能力受到很大限制,这也是标准 BOA 算法寻优精度不高的主要原因。为了进一步提升和平衡标准 BOA 算法的全局搜索和局部搜索能力,受文献[16]和文献[17]共同启发,本文将黄金正弦算法作为媒介,提出群体效应位置更新机制,针对式(5)、式(6),进行如下改进。

其一,利用黄金分割数不需要梯度信息的优点,对标准 BOA 算法的距离公式进行改进;其二,群体协同效应主要由个体矫正因子和群体协调因子构成,其利用正弦曲线值域分布特点,具体如下:

$$\tau = \frac{\sqrt{5} - 1}{2} \quad (14)$$

$$l_1 = -\pi + 2\pi(1 - \tau) \quad (15)$$

$$l_2 = -\pi + 2\pi\tau \quad (16)$$

新局部搜索公式:

$$dis_1' = |l_1x_j' - l_2x_i'| \quad (17)$$

$$x_i'^{t+1} = x_i^t |\sin \psi_1| + dis_1' f_i \psi_2 \sin \psi_1 \quad (18)$$

新全局搜索公式:

$$dis_2' = |l_1g^* - l_2x_i'| \quad (19)$$

$$x_i'^{t+1} = x_i^t |\sin \psi_1| + dis_2' f_i \psi_2 \sin \psi_1 \quad (20)$$

其中, τ 是黄金分割数, l_1 、 l_2 是具备黄金分割数优点的距离更新系数,在两只蝴蝶相互作用时获得巨大利益,提高算法收敛速度。 dis_1' 、 dis_2' 是新的距离公式, $\psi_1 \in [0, \pi]$, $\psi_2 \in [0, 2\pi]$ 的随机数。

式(18)、式(20)增加了个体矫正因子 $|\sin \psi_1|$ 的波动范围是 $[0, \frac{1}{2}]$, 从个体层面指引个体飞行调整; 群体协调因子 $\psi_2 \sin \psi_1$ 的浮动范围是 $[-2\pi, 2\pi]$, 从群体层面指引种群搜索方向,两者协同配合使得蝴蝶种群向着食物源位置搜索。与原来的位置更新公式(5)、公式(6)相比,加入了个体矫正因子、群体协调因子能够更好地指引蝴蝶从原位置到食物源位置附近,更利于保持种群多样性,进而增强全局搜索能力。

综上三节,利用混沌初始化种群,使得种群分布更加均匀,为后续算法寻优打下基础。通过反馈共享,引入随机蝴蝶加快蝴蝶种群之间的信息交流实现信息流动达到共享,增强算法跳出局部最优能力。引入群体协同,降低蝴蝶的盲目依赖性,增强及平衡全局和局部搜索能力。寻优过程中,蝴蝶种群能够进行针对性的搜索,调整适合个体进化方向,进而增强算法寻优能力。

2.4 改进算法的步骤

CFSBOA 算法:

begin

设置初始参数: 种群规模 N , 转换概率 P , 感觉模态 c , 幂指数 a , 黄金分割数 τ

Hénon 初始化种群, 计算个体适应度

while $t < T_{\max}$ do

for $i=1$ to N do

根据式(2)计算香味

比较随机蝴蝶和当前蝴蝶的适应度

if $F(x_h') < F(x_i')$

蝴蝶受式(13)中正反馈作用

else

蝴蝶受式(13)中负反馈作用

```

if rand > P
    根据式(18)进行位置更新
else
    根据式(20)进行位置更新
    检查蝴蝶个体是否超出搜索范围
    计算新位置与原位置的适应度, 并进行比较
    if F(new) < F(old)
        新位置, 香味分别作为下一次迭代的位置和香味
    end
    更新当前全局最优个体位置和感官模式 c
end for
end while
end
    
```

2.5 时间复杂度分析

CFSBOA算法主要由混沌初始化、反馈共享及群体协同效应组成,假设种群规模为 N ,搜索空间维度为 D ,则BOA的随机初始化的时间复杂度为 $O(N \times D)$,计算个体适应度为 $O(N)$,迭代过程中的时间复杂度为 $O(N \times D)$,更新最优解的时间复杂度为 $O(1)$,BOA总的时间复杂度为 $O(N \times D)$ 。

同理,CFSBOA在BOA的基础上将随机初始化替换为混沌初始化时间复杂度为 $O(N \times D)$,增加的反馈共享机制的时间复杂度为 $O(N \times D)$,增加的群体协同机制的时间复杂度为 $O(N \times D)$,因此CFSBOA总的时间复杂度为 $O(N \times D)$ 。因为CFSBOA和BOA的时间复杂度一样,所以CFSBOA并未对算法产生负面影响。

3 实验仿真与结果分析

实验环境为 Windows 7,64位操作系统,CPU为 Intel Core i5-6500H,主频3.2 GHz,内存8 GB,算法基于MATLAB2014b编写。

为了充分验证本文所提出的CFSBOA在求解复杂优化问题时的鲁棒性和有效性,将CFSBOA算法与加入混沌初始化的蝴蝶算法(chaotic butterfly optimization algorithm,CBOA)、加入反馈共享机制的蝴蝶算法(feedback butterfly optimization algorithm,FBOA)、加入群体协同效应的蝴蝶算法(sharing butterfly optimization algorithm,SBOA)、BOA^[3]、MFO(moth-flame optimization)^[18]、SSA(salp swarm algorithm)^[19]和GWO(grey wolf optimize)^[20]在12个经典测试函数上进行最优值求解,然后进行实验对比。

实验1采用如表1所示的12个基准测试函数作为目标函数,测试函数包含单峰可分(unimodal sepa-

表1 基准测试函数

Table 1 Benchmark functions

函数	维度	特征	范围	最佳值
f_1 Sphere	10	US	[-100,100]	0
f_2 Schwefel 2.22	50	UN	[-10,10]	0
f_3 Schwefel 1.2	50	UN	[-100,100]	0
f_4 Schwefel 2.21	50	US	[-100,100]	0
f_5 Matyas	2	UN	[-10,10]	0
f_6 Schaffer	2	MN	[-100,100]	0
f_7 Rastrigin	10	MS	[-5.12,5.12]	0
f_8 Ackley	50	MN	[-32,32]	0
f_9 Griewank	100	MN	[-600,600]	0
f_{10} Foxholes	2	MN	[-65.56,65.56]	≈1
f_{11} Apline	100	MN	[-10,10]	0
f_{12} Salomon	200	MN	[-100,100]	0

table,US)、单峰不可分(unimodal non-separable,UN)、多峰可分(multimodal separable,MS)、多峰不可分(multimodal non-separable,MN)等不同特征的函数。同时,测试函数的维度跨度大,从2维到200维,因此可以更加全面地检验算法的综合性能。

为保证每种算法对比的公平性,实验中最大迭代次数设为500,种群规模为30,独立运行50次,其余参数设置如表2所示。

表2 主要参数

Table 2 Main parameters

算法	主要参数
CFSBOA	$P=0.8, c=0.01, a=0.1, \tau=0.618\ 033, \gamma=1.4, \varphi=0.3$
SBOA	$P=0.8, c=0.01, a=0.1, \tau=0.618\ 033 \dots$
FBOA	$P=0.8, c=0.01, a=0.1$
CBOA	$P=0.8, c=0.01, a=0.1, \gamma=1.4, \varphi=0.3$
BOA	$P=0.8, c=0.01, a=0.1$
MFO	—
SSA	$m=2$
GWO	$a_{\max}=2, a_{\min}=0$

表3通过最优值、平均值、标准差、成功率(success rate,SR)以及平均耗时 T (单位:s)五个性能指标来评估各算法的性能,实验对比数据如表3所示。其中,成功率是计算成功次数与实验的总次数之比,一次求解成功率公式如下:

$$\begin{cases} \frac{|S_A - S_T|}{S_T} < 10^{-5}, S_T \neq 0 \\ |S_A - S_T| < 10^{-5}, S_T = 0 \end{cases} \quad (21)$$

式中, S_A 为每次求解的最优值, S_T 为基准测试函数的最佳值。

表3 基准测试函数结果对比

Table 3 Results comparison of benchmark test functions

函数	算法	最优值	平均值	标准差	SR/%	T/s	函数	算法	最优值	平均值	标准差	SR/%	T/s
f_1	CFSBOA	0.00E+00	0.00E+00	0.00E+00	100	0.40	f_7	CFSBOA	0.00E+00	0.00E+00	0.00E+00	100	0.63
	SBOA	0.00E+00	0.00E+00	0.00E+00	100	0.25		SBOA	0.00E+00	0.00E+00	0.00E+00	100	0.46
	FBOA	6.51E-80	9.13E-79	9.00E-79	100	0.37		FBOA	0.00E+00	0.00E+00	0.00E+00	100	0.56
	CBOA	7.30E-13	1.09E-12	3.10E-13	100	0.23		CBOA	0.00E+00	5.68E-16	2.81E-15	100	0.40
	BOA	7.13E-12	1.01E-11	1.50E-12	100	0.22		BOA	2.84E-14	2.49E+01	2.11E+01	28	0.38
	MFO	3.91E-16	3.44E-13	5.72E-13	100	0.11		MFO	5.97E+00	2.47E+01	1.47E+01	0	0.16
	SSA	3.12E-10	9.43E-10	3.07E-10	100	0.14		SSA	2.98E+00	1.68E+01	8.34E+00	0	0.18
	GWO	7.01E-61	1.48E-56	7.76E-56	100	0.11		GWO	0.00E+00	4.25E-01	1.06E+00	88	0.16
f_2	CFSBOA	2.02E-225	2.53E-169	0.00E+00	100	0.47	f_8	CFSBOA	8.88E-16	8.88E-16	0.00E+00	100	0.82
	SBOA	4.51E-149	2.81E-113	1.99E-112	100	0.28		SBOA	8.88E-16	8.88E-16	0.00E+00	100	0.54
	FBOA	2.68E-180	6.61E+23	3.85E+24	70	0.43		FBOA	8.88E-16	8.88E-16	0.00E+00	100	0.78
	CBOA	3.00E-12	1.73E-01	6.91E-01	92	0.27		CBOA	7.55E-14	5.62E-12	6.87E-12	100	0.49
	BOA	4.38E-13	5.40E+24	3.64E+25	68	0.26		BOA	5.15E-09	5.94E-09	3.51E-10	100	0.47
	MFO	7.42E+00	6.99E+01	3.45E+01	0	0.23		MFO	1.29E+01	1.93E+01	1.20E+00	0	0.31
	SSA	4.02E+00	1.08E+01	7.95E+00	0	0.21		SSA	2.87E+00	4.64E+00	1.13E+00	0	0.29
	GWO	6.80E-13	2.80E-12	2.12E-12	100	0.25		GWO	1.14E-11	3.57E-11	2.08E-11	100	0.34
f_3	CFSBOA	0.00E+00	0.00E+00	0.00E+00	100	4.55	f_9	CFSBOA	0.00E+00	0.00E+00	0.00E+00	100	0.96
	SBOA	0.00E+00	0.00E+00	0.00E+00	100	2.31		SBOA	0.00E+00	0.00E+00	0.00E+00	100	0.62
	FBOA	2.94E-217	5.73E-198	0.00E+00	100	4.50		FBOA	0.00E+00	0.00E+00	0.00E+00	100	0.92
	CBOA	1.61E-16	1.38E-12	1.24E-12	100	2.29		CBOA	6.35E-13	1.36E-12	3.75E-13	100	0.57
	BOA	1.08E-11	1.32E-11	9.79E-13	100	2.31		BOA	1.98E-12	1.28E-11	4.12E-12	100	0.56
	MFO	3.27E+04	6.62E+04	2.13E+04	0	1.26		MFO	3.04E+02	5.42E+02	1.43E+02	0	0.49
	SSA	3.39E+03	9.13E+03	3.85E+03	0	1.23		SSA	7.20E+00	1.34E+01	3.20E+00	0	0.44
	GWO	3.05E-03	8.96E-01	5.13E+00	0	1.25		GWO	1.09E-13	1.94E-03	6.75E-03	88	0.52
f_4	CFSBOA	1.35E-193	2.08E-156	1.42E-155	100	0.56	f_{10}	CFSBOA	9.98E-01	9.98E-01	1.84E-10	100	3.53
	SBOA	1.71E-164	4.69E-153	1.66E-152	100	0.33		SBOA	9.98E-01	5.02E+00	4.21E+00	8	1.91
	FBOA	1.11E-28	2.75E-28	9.52E-29	100	0.54		FBOA	9.98E-01	1.41E+00	6.41E-01	30	3.49
	CBOA	6.68E-14	2.89E-11	8.86E-11	100	0.32		CBOA	9.98E-01	9.98E-01	2.11E-10	100	1.86
	BOA	5.51E-09	6.20E-09	3.67E-10	100	0.31		BOA	9.98E-01	1.31E+00	5.06E-01	16	1.80
	MFO	6.74E+01	8.32E+01	5.17E+00	0	0.25		MFO	9.98E-01	1.99E+00	1.49E+00	32	0.85
	SSA	1.22E+01	2.00E+01	3.49E+00	0	0.23		SSA	9.98E-01	1.22E+00	6.43E-01	88	0.88
	GWO	2.92E-05	3.51E-04	3.50E-04	10	0.26		GWO	9.98E-01	4.42E+00	4.25E+00	42	0.86
f_5	CFSBOA	0.00E+00	2.55E-297	0.00E+00	100	0.61	f_{11}	CFSBOA	2.17E-215	6.29E-165	0.00E+00	100	0.61
	SBOA	0.00E+00	0.00E+00	0.00E+00	100	0.41		SBOA	5.03E-165	1.76E-151	1.19E-150	100	0.45
	FBOA	3.41E-51	3.63E-49	8.70E-49	100	0.57		FBOA	1.41E-45	3.18E-44	3.64E-44	100	0.59
	CBOA	1.07E-18	6.74E-13	2.56E-13	100	0.37		CBOA	9.26E-12	3.79E-03	1.71E-02	76	0.44
	BOA	6.30E-17	9.28E-13	3.70E-13	100	0.37		BOA	3.24E-10	8.30E-10	3.61E-10	100	0.43
	MFO	1.34E-86	2.53E-22	1.79E-21	100	0.13		MFO	4.29E+01	6.64E+01	1.60E+01	0	0.39
	SSA	4.86E-18	1.29E-15	1.85E-15	100	0.16		SSA	1.93E+01	2.94E+01	5.58E+00	0	0.34
	GWO	5.97E-129	1.82E-102	9.01E-102	100	0.14		GWO	2.80E-08	3.39E-03	2.86E-03	6	0.43
f_6	CFSBOA	0.00E+00	0.00E+00	0.00E+00	100	0.66	f_{12}	CFSBOA	0.00E+00	0.00E+00	0.00E+00	100	38.72
	SBOA	0.00E+00	0.00E+00	0.00E+00	100	0.44		SBOA	1.97E-305	3.16E-258	0.00E+00	100	19.41
	FBOA	0.00E+00	1.53E-02	1.10E-02	4	0.58		FBOA	0.00E+00	6.38E-106	4.51E-105	100	38.54
	CBOA	1.44E-15	1.94E-04	1.37E-03	96	0.37		CBOA	3.99E-14	7.45E-02	4.45E-02	18	19.47
	BOA	1.65E-05	1.28E-02	7.97E-03	0	0.36		BOA	1.52E-16	3.98E-03	1.97E-02	94	19.44
	MFO	0.00E+00	9.49E-03	4.81E-03	10	0.14		MFO	0.00E+00	5.56E-30	9.42E-30	100	10.08
	SSA	2.60E-14	5.05E-03	4.90E-03	46	0.17		SSA	1.12E-16	2.35E-12	3.52E-12	100	10.00
	GWO	0.00E+00	5.64E-03	4.84E-03	42	0.14		GWO	7.21E-08	3.75E-04	7.36E-04	30	10.19

首先,表3通过最优值和平均值来反映算法的寻优精度和收敛速度的能力。对于5个单峰函数 $f_1 \sim f_5$, CFSBOA对 f_1 、 f_3 和 f_5 函数求解时,找到了理论最佳值。随着函数维度增加,算法求解最优值的难度也将呈指数增长,因此在 f_2 和 f_4 函数上寻优精度有所降低,并未寻到理论最佳值。伴随着函数维度的增加,对比算法MFO、SSA和GWO的精度有所下降,尤其是在求解50维的 f_3 时,MFO与理论最佳值相差 1.0×10^4 的量级。对于CBOA、FBOA和SBOA的寻优精度和收敛速度,三种改进策略得到的结果均比BOA好,这充分证明了不同改进策略对算法的寻优能力均有不同程度的增强。在5个单峰函数上,除 f_5 上CFSBOA的平均值略微高于SBOA外,CFSBOA的综合性能均比其他几种算法好。对于7个多峰函数 $f_6 \sim f_{12}$,与单峰函数相比,多峰函数的求解会变得更加困难,因此寻优精度低于单峰函数。随着多峰函数维度增加,CFSBOA在对7个多峰函数求解时,5个多峰函数上能寻到理论最佳值。对7个多峰函数寻优时,CFSBOA、SBOA和FBOA算法的寻优能力均优于标准的BOA,CBOA仅在 f_{12} 的寻优能力不如BOA,但在其他函数的寻优能力均强于标准BOA。在 f_8 上,CFSBOA、SBOA和FBOA的寻优精度和收敛速度都比其他算法要好。在 f_{10} 上,各算法都找到理论最佳值,但CFSBOA的平均值和标准差均比其他算法好。其余函数的求解上,CFSBOA都能寻到理论最佳值,进一步验证了三种改进策略的寻优有效性。由此可见,CFSBOA对单峰可分、单峰不可分、多峰可分和多峰不可分及高维度的基准测试函数有着明显的优势。

其次,表3中成功率和标准差体现算法跳出局部最优的能力及稳定性。CFSBOA的结果很接近理论最佳值,标准差也比较小,对基准测试函数寻优有着较强的稳定性。在12个基准测试函数上,CFSBOA的寻优成功率都是100%,SBOA在11个函数寻优成功率为100%,FBOA在9个函数成功率为100%,虽然CBOA在 $f_{11} \sim f_{12}$ 的成功率不如标准BOA,但其在8个函数上寻优成功率为100%,而标准BOA只在7个函数上成功率为100%。另外,随着基准测试函数维度增加,标准BOA的寻优精度不高的特点愈发明显,精度大多停留在 10^{-14} 以下,而CFSBOA和FBOA引入的反馈共享机制,增加算法逃离局部最优能力,寻优精度得到提高;CFSBOA和SBOA引入群体协同效应使得算法的寻优精度和收敛速度大幅提升,证明了

反馈共享机制和群体协同效应机制对算法的局部和全局搜索能力有很大的增强作用。

最后,从平均耗时角度分析。由表3可知,MFO和SSA的平均耗时最短,改进算法相对标准BOA平均耗时都要长,因为改进策略中引入更多的机制使得算法不仅能在搜索空间中进行大范围搜索,还能找到更多的解,所以导致算法平均耗时变长,但是CFSBOA的平均耗时与其他算法相比,所增加部分在可接受范围内。

为了形象地观测改进策略的性能,对CFSBOA与SBOA、FBOA、CBOA、BOA、MFO、SSA与GWO进行收敛性分析。图2给出部分测试函数的平均收敛曲线图。因为寻优精度较高,为了便于观察平均曲线的收敛情况,本文的纵坐标取以10为底的对数。

在单峰函数 f_2 上,CFSBOA的寻优精度最高,SBOA其次,其余算法的寻优精度都不高,同时CFSBOA的收敛速度最快。在 f_3 上,寻优精度依次为CFSBOA、SBOA和FBOA,其余算法的精度不高,CFSBOA相比其他算法求解最佳值的迭代次数更少并且能够快速收敛。

在多峰函数 f_8 上,所有算法未能寻到最佳值,但CFSBOA、SBOA和FBOA的寻优性能突出,寻优精度比其他算法高,CFSBOA和SBOA能够快速收敛到精度较高的解上且收敛速度较快。在 f_{10} 上,所有算法都寻到最佳值,但CFSBOA和CBOA收敛速度最快,CBOA对于该函数的寻优能力相比其他函数优势突出。在 f_{11} 上,CFSBOA虽未能寻到最佳值,但寻优精度最高且相比其他算法收敛速度快。在 f_{12} 上,CFSBOA和FBOA都能寻到最优值;对于收敛速度,CFSBOA收敛速度最快,FBOA其次,SBOA慢于两者,其余算法收敛速度慢且收敛精度低。

另外,由图2(a)~(f)可以看出,在种群迭代前期,CFSBOA、SBOA和FBOA曲线速度下降很快且寻优精度很高,说明引入群体协同机制和反馈共享机制能够增强算法跳出局部最优的能力,使得算法在迭代开始时收敛速度变快。随着迭代的进行,CFSBOA能够持续寻优,而其他算法在迭代后期陷入局部最优出现停滞状态。同时,在 f_2 上,CFSBOA比只增加反馈共享机制的FBOA收敛速度更快且寻优精度更高,证明了群体协同效应机制,通过对个体和群体的协同配合使得种群逃离局部最优和全局搜索的能力得到增强,有利于算法的持续寻优。在 f_3 上,CFSBOA比只增加群体协同效应的SBOA收敛速度更快,这证明

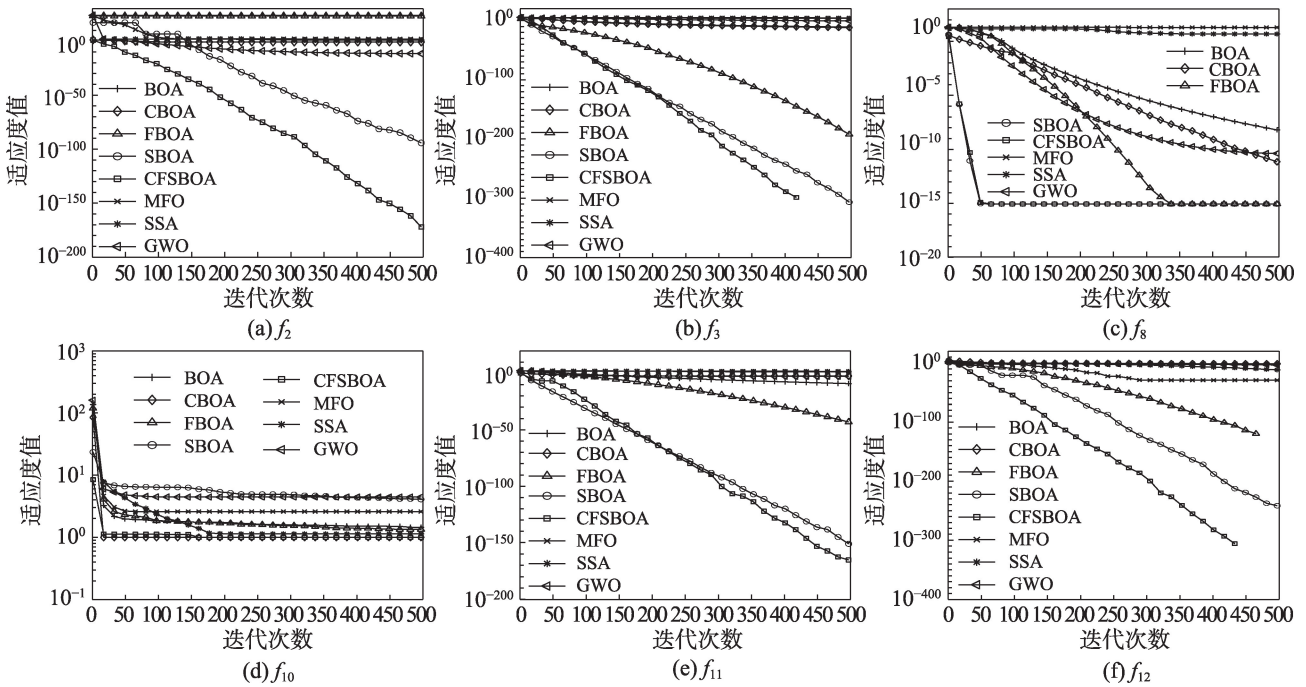


图2 基准测试函数平均收敛曲线

Fig.2 Average convergence curve of benchmark test functions

反馈共享机制能帮助算法快速找到最优解位置附近。另外,CFSBOA在 f_3 和 f_{12} 上寻到了基准测试函数的理论最佳值。除了在 f_2 上,FBOA的平均收敛曲线和标准BOA平均收敛曲线在一个数量级上,其余改进算法的平均曲线都在标准BOA算法平均收敛曲线的下方,验证了改进算法的有效性。结合表3和图2的实验结果和平均收敛曲线图证明,不管基准测试函数是单峰、多峰,还是在低维、高维,CFSBOA的

综合性能均比其他算法要好。图2(b)和图2(f),因纵坐标取对数的原因加上CFSBOA在这两个函数上找到了0,所以曲线后面没有显示。

根据Derrac等人在文献[21]中提出的,对于改进进化算法性能的评估,需要进行统计检验。本文选择用Wilcoxon秩和检验并在5%的显著性水平下进行。表4给出了CFSBOA和其他算法在所有基准测试函数上的 p 值。假设CFSBOA为最佳算法,则在

表4 基准函数Wilcoxon秩和检验的 p 值

Table 4 p value for Wilcoxon's rank-sum test on benchmark functions

F	p -value win						
	BOA	CBOA	FBOA	SBOA	MFO	SSA	GWO
f_1	3.31E-20+	3.31E-20+	3.31E-20+	N/A=	3.31E-20+	3.31E-20+	3.31E-20+
f_2	7.07E-18+	7.07E-18+	2.78E-17+	7.07E-18+	7.07E-18+	7.07E-18+	7.07E-18+
f_3	3.31E-20+	3.31E-20+	3.31E-20+	N/A=	3.31E-20+	3.31E-20+	3.31E-20+
f_4	7.07E-18+	7.07E-18+	7.07E-18+	2.39E-13+	7.07E-18+	7.07E-18+	7.07E-18+
f_5	1.12E-18+	1.12E-18+	1.12E-18+	6.50E-05+	1.12E-18+	1.12E-18+	1.12E-18+
f_6	3.31E-20+	3.31E-20+	3.31E-20+	N/A=	7.55E-21+	3.28E-20+	5.25E-13+
f_7	3.31E-20+	1.59E-01-	N/A=	N/A=	3.31E-20+	3.31E-20+	1.82E-03+
f_8	3.31E-20+	3.31E-20+	N/A=	N/A=	3.31E-20+	3.31E-20+	3.31E-20+
f_9	3.31E-20+	3.31E-20+	N/A=	N/A=	3.31E-20+	3.31E-20+	3.31E-20+
f_{10}	7.07E-18+	5.74E-01-	7.07E-18+	7.07E-18+	2.38E-12+	3.59E-12+	1.27E-12+
f_{11}	7.07E-18+	7.07E-18+	7.07E-18+	7.07E-18+	7.07E-18+	7.07E-18+	7.07E-18+
f_{12}	3.31E-20+	3.31E-20+	N/A=	3.31E-20+	1.75E-06+	3.31E-20+	3.31E-20+
+/-/-	12/0/0	10/0/2	8/4/0	6/6/0	12/0/0	12/0/0	12/0/0

CFSBOA vs SBOA、CFSBOA vs FBOA 等之间进行成对比较。另外,由于最佳算法与自身无法进行比较,因此针对最佳算法采用 N/A 标记,代表“不适用”,符号“=”“+”和“-”分别代表相当于、优于、劣于对比算法。Derrac 等人的文献中指出 p 值小于 0.05 认为是拒绝零假设的有力验证。表 4 结果表明,CFSBOA 的 p 值基本小于 0.05。只有在 f_7 、 f_{10} 的 CFSBOA vs CBOA 时, p 值大于 0.05。这表明 CFSBOA 算法的优越性在统计检验上是显著的。因此可以认为 CFSBOA 算法比其他算法具有更高的收敛精度, p 值大于 0.05 已用粗体显示。

文献[22]提出使用平均绝对误差(mean absolute error, MAE)对优化算法进行排序,这是一种有效且可行的性能指标。表 5 中所有算法的分析是基于 12 个基准测试函数的 MAE。MAE 公式如下:

$$MAE = \sum_{i=1}^{N_f} \frac{|B_i - V_i|}{N_f} \quad (22)$$

式(22)中, B_i 为算法最优值的平均值; V_i 为相应基准函数的理论最优值; N_f 为基准函数个数。

表 5 各算法 MAE 排名

Table 5 MAE ranking of each algorithm

算法	MAE	Rank
CFSBOA	1.66E-04	1
CBOA	1.04E-02	2
SBOA	2.98E-01	3
GWO	3.52E-01	4
SSA	8.46E+02	5
MFO	5.38E+03	6
FBOA	2.93E+22	7
BOA	5.38E+22	8

由表 5 可知,CFSBOA 排名第一,CBOA 排名第二,SBOA 排名第三。同时,与另外七种算法相比,CFSBOA 的 MAE 值最小。表 5 的实验数据进一步表明了 CFSBOA 算法的有效性和鲁棒性。

实验 2 为了更进一步评估 CFSBOA 的有效性和鲁棒性。实验 2 分为两个环节:第一环节主要采用 CEC2014 函数对 CFSBOA 的综合性能进行验证;第二环节验证 CFSBOA 的竞争性,在保证种群规模、迭代次数、独立运行次数及基准测试函数维度相同的条件下,将其与目前改进蝴蝶优化算法及新改进的群智能算法进行公平对比。

本文在 CEC2014 测试函数选取部分单峰、多峰、混合(hybrid function, HF)及复合(composition function, CF)函数类型进行验证,所选 CEC2014 部分函数如表 6 所示。为了保证算法对比公平性,将种群规模设为 30,CEC2014 函数的搜索维度设为 30,迭代次数 500,独立运行 30 次,实验结果如表 7 所示。

表 6 CEC2014 函数(部分)

Table 6 CEC2014 function (part)

函数	特征	定义域	最佳值
CEC03	UF	[-100,100]	300
CEC05	MF	[-100,100]	500
CEC16	HF	[-100,100]	1 600
CEC25	CF	[-100,100]	2 500
CEC27	CF	[-100,100]	2 700
CEC30	CF	[-100,100]	3 000

表 7 分别记录各种算法的平均值和标准差,因 CEC2014 函数较为复杂,很难找到最佳值。由表 7 可知,CFSBOA 在 CEC03 的平均收敛不如 GWO 和

表 7 CEC2014 测试函数的结果对比

Table 7 Comparison of CEC2014 test function results

函数	指标	CFSBOA	BOA	MFO	SSA	GWO	PSO
CEC03	Mean	8.951 0E+04	1.077 5E+10	9.546 7E+04	1.026 6E+05	5.368 9E+04	2.799 2E+04
	Std	4.765 1E+02	0.000 0E+00	5.072 7E+04	2.751 8E+04	1.387 3E+04	2.221 6E+04
CEC05	Mean	5.211 5E+02	5.217 6E+02	5.203 2E+02	5.200 6E+02	5.210 8E+02	5.210 0E+02
	Std	8.644 2E-03	3.468 9E-13	1.530 4E-01	9.554 9E-02	4.418 3E-02	7.922 0E-02
CEC16	Mean	1.613 4E+03	1.614 9E+03	1.612 8E+03	1.612 6E+03	1.612 2E+03	1.612 5E+03
	Std	2.494 5E-04	2.312 6E-13	5.504 8E-01	3.771 0E-01	5.385 8E-01	4.627 2E-01
CEC25	Mean	2.700 0E+03	5.125 0E+03	2.719 0E+03	2.718 6E+03	2.713 9E+03	2.714 9E+03
	Std	0.000 0E+00	0.000 0E+00	1.021 0E+01	4.834 9E+00	6.214 1E+00	3.990 2E+00
CEC27	Mean	2.900 0E+03	1.933 9E+04	3.627 7E+03	3.558 1E+03	3.464 5E+03	3.412 5E+03
	Std	0.000 0E+00	1.110 1E-11	1.995 7E+02	2.412 9E+02	1.317 1E+02	1.711 3E+02
CEC30	Mean	3.200 0E+03	3.236 5E+07	4.084 2E+04	6.971 4E+04	1.222 8E+05	1.806 6E+04
	Std	0.000 0E+00	1.515 6E-08	3.337 1E+04	3.733 2E+04	7.403 6E+04	8.312 1E+03

PSO,但标准差远低于两者;在CEC05和CEC16函数的平均值略低于GWO、MFO、SSA和PSO,但CFSBOA的标准差低于四种算法。另外在最后三个函数上,CFSBOA算法的平均值和标准差均低于其他算法,验证CFSBOA算法的鲁棒性和有效性。

与参考文献中数据进行对比,每个算法的种群规模设为30,迭代次数为500,基准测试函数搜索维度为30,对独立运行30次后的结果进行对比,其中“—”代表无数据,如表8所示。

由表8可知,CFSBOA算法在 f_1 、 f_3 、 f_6 、 f_7 、 f_9

表8 与参考文献中算法平均值和标准差的结果对比

Table 8 Results comparison of mean and standard deviation of algorithms in references

算法	指标	f_1	f_2	f_3	f_4	f_5	f_6
CFSBOA	Mean	0.00E+00	3.40E-170	0.00E+00	1.13E-155	2.83E-296	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	6.18E-155	0.00E+00	0.00E+00
HPSOBOA ^[23]	Mean	3.74E-104	2.63E-22	3.04E-71	3.61E-46	—	—
	Std	2.05E-103	1.44E-21	1.67E-70	1.97E-45	—	—
LBOA ^[23]	Mean	3.92E-12	1.39E-09	2.74E-12	2.30E-09	—	—
	Std	4.46E-12	2.08E-09	2.44E-12	2.36E-09	—	—
IBOA ^[23]	Mean	1.61E-30	5.11E-19	6.15E-31	1.36E-19	—	—
	Std	3.90E-30	1.73E-18	1.16E-30	1.97E-19	—	—
CWBOA ^[9]	Mean	0.00E+00	3.86E-134	—	3.29E-134	0.00E+00	0.00E+00
	Std	0.00E+00	1.52E-133	—	1.80E-133	0.00E+00	0.00E+00
BOA-CE ^[11]	Mean	1.26E-95	3.60E-47	2.44E-09	8.82E-07	—	—
	Std	9.78E-96	2.50E-47	1.83E-10	1.51E-07	—	—
CSO ^[24]	Mean	3.50E-14	2.68E-08	7.17E-09	1.04E-02	—	—
	Std	6.34E-14	2.61E-08	1.16E-08	7.96E-03	—	—
MPA ^[25]	Mean	4.51E-23	3.01E-13	1.27E-04	3.26E-09	2.97E-51	3.84E-13
	Std	3.56E-23	3.07E-13	2.19E-04	2.22E-09	1.57E-50	2.10E-12
MSIWOA ^[26]	Mean	0.00E+00	7.17E-203	0.00E+00	4.59E-139	—	—
	Std	0.00E+00	1.64E-203	0.00E+00	2.51E-138	—	—
QOWOA-CS ^[27]	Mean	1.01E-232	2.23E-125	2.61E-201	3.58E-103	—	—
	Std	0.00E+00	5.85E-125	0.00E+00	6.47E-103	—	—
算法	指标	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}
CFSBOA	Mean	0.00E+00	8.88E-16	0.00E+00	9.98E-01	1.27E-166	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	1.20E-10	0.00E+00	0.00E+00
HPSOBOA ^[23]	Mean	0.00E+00	8.96E-11	0.00E+00	—	2.54E-45	2.53E-08
	Std	0.00E+00	4.73E-10	0.00E+00	—	1.39E-44	1.38E-07
LBOA ^[23]	Mean	0.00E+00	2.34E-12	3.48E-13	—	6.32E-14	3.65E-02
	Std	0.00E+00	7.87E-12	8.78E-13	—	1.73E-13	4.88E-02
IBOA ^[23]	Mean	0.00E+00	8.88E-16	0.00E+00	—	8.93E-20	2.25E-32
	Std	0.00E+00	0.00E+00	0.00E+00	—	1.19E-19	5.88E-32
CWBOA ^[9]	Mean	0.00E+00	8.88E-16	0.00E+00	—	2.13E-137	—
	Std	0.00E+00	0.00E+00	0.00E+00	—	8.15E-137	—
BOA-CE ^[11]	Mean	1.01E+01	4.44E-15	0.00E+00	—	—	—
	Std	3.46E+01	0.00E+00	0.00E+00	—	—	—
CSO ^[24]	Mean	2.40E+01	3.75E+00	3.56E-01	9.98E-01	—	—
	Std	6.48E+00	1.68E+00	1.91E-01	3.39E-07	—	—
MPA ^[25]	Mean	0.00E+00	1.59E-12	0.00E+00	9.98E-01	9.12E-14	5.58E-17
	Std	0.00E+00	9.87E-13	0.00E+00	1.62E-16	8.01E-14	1.40E-16
MSIWOA ^[26]	Mean	0.00E+00	0.00E+00	0.00E+00	—	—	—
	Std	0.00E+00	0.00E+00	0.00E+00	—	—	—
QOWOA-CS ^[27]	Mean	0.00E+00	8.88E-16	0.00E+00	—	—	—
	Std	0.00E+00	0.00E+00	0.00E+00	—	—	—

和 f_{12} 函数上平均值均为最佳值;在 f_2 和 f_5 函数平均值高于 MSIWOA(multi-strategy improved whale optimization algorithm)算法和 CWBOA(Cauchy variation and adaptive weight butterfly optimization algorithm)算法;在 f_8 上,CFSBOA 算法寻优性能不如 MSIWOA 算法;在 f_{10} 上,CFSBOA 的标准差高于 MPA(marine predator algorithm),而在其余函数上 CFSBOA 算法的平均值和标准差低于其他算法。因此,本文提出的 CFSBOA 算法在求解复杂优化问题上具有一定的竞争性。

综上所述,实验 1 和实验 2 的实验结果充分证明了混沌反馈共享和群体协同效应的蝴蝶优化算法对于求解多种类型的基准测试函数,尤其是多峰、高维的函数具有较好的鲁棒性和较高的寻优精度。

4 结束语

为了增强标准 BOA 算法的跳出局部最优的能力及提高算法的寻优精度和加快收敛速度,本文提出了混沌反馈共享和群体协同效应的蝴蝶优化算法,并将改进的蝴蝶优化算法应用于基准测试函数和 CEC2014 函数寻优。混沌机制对种群进行初始化,增加种群的多样性,该机制对特定函数优势明显;迭代过程中,通过利用正负反馈的原理,构建反馈共享网络使得蝴蝶信息来源更加多元,增强算法逃离局部最优的能力;利用群体协同效应平衡和增强算法的全局和局部搜索能力,提高寻优精度和加快算法收敛。三个改进机制能增强算法跳出局部最优的能力,同时提高寻优精度和加快算法收敛能力。另外,本文不仅使用最优值、标准差等指标对算法评估,还使用统计检验 Wilcoxon 对算法进行显著性分析和 MAE 排序各算法,实验结果表明 CFSBOA 算法具备出色优化性能,全局和局部搜索能力得到增强,能够收敛到精度更高的最优解,同时算法的有效性和鲁棒性也得到验证。

在未来,准备改进混沌机制增强其对测试函数的一般性以及将改进的蝴蝶优化算法应用于实际的工程优化问题,解决工程问题的同时,进一步验证算法的性能。

参考文献:

- [1] 周蓉,李俊,王浩.基于灰狼优化的反向学习粒子群算法[J].计算机工程与应用,2020,56(7):48-56.
ZHOU R, LI J, WANG H. Reverse learning particle swarm optimization based on grey wolf optimization[J]. Computer Engineering and Applications, 2020, 56(7): 48-56.
- [2] 张明伟,李波,屈晓龙,等.基于混合蚁群算法的异质车队低碳 VRP 研究[J].计算机工程与应用,2020,56(14):240-249.
ZHANG M W, LI B, QU X L, et al. Research on low carbon VRP of heterogeneous fleet based on hybrid ant colony algorithm[J]. Computer Engineering and Applications, 2020, 56(14): 240-249.
- [3] ARORA S, SINGH S. Butterfly optimization algorithm: a novel approach for global optimization[J]. Soft Computing, 2019, 23(3): 715-734.
- [4] FATHY A. Butterfly optimization algorithm based methodology for enhancing the shaded photovoltaic array extracted power via reconfiguration process[J]. Energy Conversion and Management, 2020, 220: 113115.
- [5] MAHESHWARI P, SHARMA A K, VERMA K. Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization[J]. Ad Hoc Networks, 2021, 110: 102317.
- [6] TAN L S, ZAINUDDIN Z, ONG P. Wavelet neural networks based solutions for elliptic partial differential equations with improved butterfly optimization algorithm training[J]. Applied Soft Computing, 2020, 95: 106518.
- [7] WEN L, CAO Y. A hybrid intelligent predicting model for exploring household CO₂ emissions mitigation strategies derived from butterfly optimization algorithm[J]. Science of the Total Environment, 2020, 727: 138572.
- [8] 张达敏,王依柔,徐航,等.认知智能电网中基于能效优化的频谱分配策略[J].控制与决策,2021,36(8):1901-1910.
ZHANG D M, WANG Y R, XU H, et al. Spectrum allocation strategy based on energy efficiency optimization in cognitive smart grid[J]. Control and Decision, 2021, 36(8): 1901-1910.
- [9] 高文欣,刘升,肖子雅,等.柯西变异和自适应权重优化的蝴蝶算法[J].计算机工程与应用,2020,56(15):43-50.
GAO W X, LIU S, XIAO Z Y, et al. Butterfly optimization algorithm based on Cauchy variation and adaptive weight[J]. Computer Engineering and Applications, 2020, 56(15): 43-50.
- [10] 王依柔,张达敏.融合正弦余弦和无限折叠迭代混沌映射的蝴蝶优化算法[J].模式识别与人工智能,2020,33(7):660-669.
WANG Y R, ZHANG D M. Butterfly optimization algorithm combining sine cosine and iterative chaotic map with infinite collapses[J]. Pattern Recognition and Artificial Intelligence, 2020, 33(7): 660-669.
- [11] LI G C, SHUANG F, ZHAO P, et al. An improved butterfly optimization algorithm for engineering design problems using the cross-entropy method[J]. Symmetry, 2019, 11(8): 1049.
- [12] 高文欣,刘升,肖子雅,等.全局优化的蝴蝶优化算法[J].

- 计算机应用研究, 2020, 37(10): 2966-2970.
- GAO W X, LIU S, XIAO Z Y, et al. Butterfly optimization algorithm for global optimization[J]. Application Research of Computers, 2020, 37(10): 2966-2970.
- [13] 王依柔, 张达敏, 徐航, 等. 基于自适应扰动的疯狂蝴蝶算法[J]. 计算机应用研究, 2020, 37(11): 3276-3280.
- WANG Y R, ZHANG D M, XU H, et al. Crazy butterfly algorithm based on adaptive perturbation[J]. Application Research of Computers, 2020, 37(11): 3276-3280.
- [14] 张达敏, 陈忠云, 辛梓芸, 等. 基于疯狂自适应的樽海鞘群算法[J]. 控制与决策, 2020, 35(9): 2112-2120.
- ZHANG D M, CHEN Z Y, XIN Z Y, et al. Salp swarm algorithm based on craziness and adaptive[J]. Control and Decision, 2020, 35(9): 2112-2120.
- [15] 谭光兴, 朱燕飞, 毛宗源. 基于 Hénon 映射的自适应克隆选择优化算法[J]. 计算机工程与应用, 2006, 42(9): 73-76.
- TAN G X, ZHU Y F, MAO Z Y. Adaptive clone selection optimization algorithm based on Hénon map[J]. Computer Engineering and Applications, 2006, 42(9): 73-76.
- [16] TANYILDIZI E, DEMIR G. Golden sine algorithm: a novel math-inspired algorithm[J]. Advances in Electrical and Computer Engineering, 2017, 17(2): 71-78.
- [17] 周新, 邹海. 融合黄金正弦混合变异的自适应樽海鞘群算法[J]. 计算机工程与应用, 2021, 57(12): 75-85.
- ZHOU X, ZOU H. Adaptive salp swarm algorithm with golden sine algorithm and hybrid mutation[J]. Computer Engineering and Applications, 2021, 57(12): 75-85.
- [18] MIRJALILI S. Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm[J]. Knowledge-Based Systems, 2015, 89: 228-249.
- [19] MIRJALILI S, GANDOMI A H, MIRJALILI S Z, et al. Salp swarm algorithm: a bio-inspired optimizer for engineering design problems[J]. Advances in Engineering Software, 2017, 114: 163-191.
- [20] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.
- [21] DERRAC J, GARCÍA S, MOLINA D, et al. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms[J]. Swarm and Evolutionary Computation, 2011, 1(1): 3-18.
- [22] NABIL E. A modified flower pollination algorithm for global optimization[J]. Expert Systems with Applications, 2016, 57: 192-203.
- [23] ZHANG M J, LONG D Y, 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.
- [24] AHMED A M, RASHID T A, SAEED S A M, et al. Cat swarm optimization algorithm: a survey and performance evaluation[J]. Computational Intelligence and Neuroscience, 2020(1): 20.
- [25] FARAMARZI A, HEIDARINEJAD M, MIRJALILI S, et al. Marine predators algorithm: a nature-inspired metaheuristic [J]. Expert Systems with Applications, 2020, 152: 113377.
- [26] 郝晓弘, 宋吉祥, 周强, 等. 混合策略改进的鲸鱼优化算法[J]. 计算机应用研究, 2020, 37(12): 3622-3626.
- HAO X H, SONG J X, ZHOU Q, et al. Improved whale optimization algorithm based on hybrid strategy[J]. Application Research of Computers, 2020, 37(12): 3622-3626.
- [27] 冯文涛, 邓兵. 一种基于交叉选择的柯西反向鲸鱼优化算法[J]. 兵器装备工程学报, 2020, 41(8): 131-137.
- FENG W T, DENG B. Quasi-oppositional whale optimization algorithm based on crossover and selection strategy[J]. Journal of Ordnance Equipment Engineering, 2020, 41(8): 131-137.



李守玉 (1996—), 男, 贵州安顺人, 硕士研究生, 主要研究方向为进化计算、深度学习。

LI Shouyu, born in 1996, M.S. candidate. His research interests include evolutionary computation and deep learning.



何庆 (1982—), 男, 贵州黔南州人, 博士, 副教授, 主要研究方向为大数据应用、进化计算。

HE Qing, born in 1982, Ph.D., associate professor. His research interests include big data application and evolutionary computation.



杜逆索 (1986—), 男, 贵州六盘水人, 博士研究生, 讲师, 主要研究方向为深度学习。

DU Nisuo, born in 1986, Ph.D. candidate, lecturer. His research interest is deep learning.