

基于天牛须搜索自适应的樽海鞘算法

杨 兴, 郭明昊, 方 霞, 祝忠明, 蒋美琪

(成都理工大学 信息科学与技术学院, 四川 成都 610059)

摘 要:针对樽海鞘算法在优化求解问题时收敛速度慢和局部优化能力差等缺点,引入不同的优化策略对其进行改进,提出一种基于天牛须搜索自适应的樽海鞘算法。改进的樽海鞘算法在领导者位置更新中引入天牛须搜索机制和洛伦兹函数替代基本樽海鞘算法领导者位置更新公式中的随机值,提高了算法的局部优化能力;在追随者位置更新中引入自适应惯性权重,调节自身位置和上一代位置对追随者位置更新的影响,在全局搜索和局部搜索之间提供更好的平衡。将改进的樽海鞘算法通过对8个不同类型的基准测试函数进行优化实验分析,结果表明改进的樽海鞘算法相较于基本的樽海鞘算法、其他已经改进的樽海鞘算法在求解精度以及收敛速度上均有明显的提高,且具有更佳的鲁棒性。

关键词:樽海鞘算法;天牛须搜索;洛伦兹函数;自适应;全局最优

中图分类号:TP301

文献标识码:A

文章编号:1673-629X(2021)06-0001-06

doi:10.3969/j.issn.1673-629X.2021.06.001

Salp Swarm Algorithm Based on Beetle Antennae Search and Adaptive

YANG Xing, GUO Ming-hao, FANG Xia, ZHU Zhong-ming, JIANG Mei-qi

(School of Information Science and Technology, Chengdu University of Technology, Chengdu 610059, China)

Abstract: In order to improve the shortcomings such as slow convergence speed and poor local optimization ability of the salp swarm algorithm in solving problems, different optimization strategies are introduced and an improved salp swarm algorithm based on beetle antennae search and adaptive is proposed. The improved salp swarm algorithm introduces the beetle antennae search and Lorentz function instead of the random value in the leader position updating formula of the basic salp algorithm, which improves its local optimization ability. Introducing adaptive inertial weights in the follower's position update to adjust the influence of its own position and the previous generation's position on the follower's position update, providing a better balance between global search and local search. The improved salp swarm algorithm has been optimized and analyzed by 8 different test functions. It is showed that compared with the basic salp swarm algorithm and other improved salp swarm algorithm, the proposed algorithm has obvious improvement in solving accuracy and convergence speed with better robustness.

Key words: salp swarm algorithm; beetle antennae search; Lorentz function; adaptive; global optimal

0 引 言

樽海鞘算法(salp swarm algorithm, SSA)^[1]是Mirjalili等人基于樽海鞘链在海洋中觅食的种群机制开发的一种群体仿生优化算法。与其他著名的优化算法相比,SSA在许多问题上都证明了它的优越性。但SSA作为一种随机群体优化算法,依旧存在着与其他相似算法^[2-3]同样的易陷入局部最优解以及收敛速度慢等缺点。文献[4]提出一种基于反向学习和新的局部搜索算法改进樽海鞘算法用于特征选择和特征的最优子集选择,并验证了算法在大多数使用的数据集上的优越性。文献[5]在樽海鞘算法中引入反向学习与混沌局部搜索,对测试函数进行全局寻优验证算法的

性能。文献[6]在樽海鞘算法中食物源位置引入动态权重因子,构成自适应樽海鞘算法,并与BP神经网络结合形成一种新的预测模型用于对土壤墒情预测。文献[7]将樽海鞘算法运用于TDOA问题,取得了更高的定位精度。文献[8]提出一种具有交叉变换的二进制樽海鞘算法,引入交叉算子增强算法的探索能力,改进的算法在特征选择过程中约90%的数据集优于其他算法。文献[9]提出了混沌樽海鞘算法,使用十个不同的混沌图来提高收敛速度和结果精度,并在特征选择中验证了通过逻辑混沌映射可以显著提高原始SSA的性能。文献[10]提出基于二次积分和火神经模型的混沌樽海鞘算法,将二次积分和火模型产生的

收稿日期:2020-08-10

修回日期:2020-12-15

基金项目:国家重点研发计划深地资源勘查开采重点专项(2018YFC0603301)

作者简介:杨 兴(1996-),男,硕士研究生,研究方向为智能算法、无线传感网络;祝忠明,博士,副教授,研究方向为信号处理、智能计算。

混沌振荡引入到 SSA 算法中。这样可以改善勘探与开发之间的平衡,从而在不同种群之间产生多样性,跳出局部最优。并在不同的工程应用中验证了改进的樽海鞘算法的可行性。文献[11]提出一种增强型樽海鞘算法,对其进行函数优化验证性能并运用于变速风力发电机中。

为提高樽海鞘算法在求解问题时的优化性能,引入天牛须搜索策略和自适应权重值,提出一种基于天牛须搜索自适应的樽海鞘算法 (beetle antennae search and adaptive salp swarm algorithm, BASSA)。主要的改进为:(1)在樽海鞘算法的领导者位置更新中引入洛伦兹函数替代樽海鞘算法中 c_2 随机值,控制领导者的全局搜索范围;(2)引入天牛须搜索机制替代樽海鞘算法的 c_3 的随机值,决定领导者位置更新方向;(3)在追随者位置更新中,为了更好地权衡全局搜索和局部搜索,引入自适应惯性权重,调节自身位置和上一个个体位置对追随者位置更新的权重。通过对多个不同类型的基准函数进行求解优化对比,验证了改进的樽海鞘算法的可行性与鲁棒性。

1 樽海鞘算法

樽海鞘算法起源于樽海鞘链的群聚行为,樽海鞘链群体分为两部分:领导者和追随者。在樽海鞘链中最前端的个体被认为是领导者,其他个体认为是追随者。领导者根据食物源位置引导着追随者移动。将樽海鞘群体对最佳食物源搜索的过程类比为对函数寻优。在樽海鞘算法中,为了更加平衡算法的全局性和鲁棒性,选取一半的种群为领导者。在一个 $N \times d$ 维的空间内搜寻食物源, N 为樽海鞘种群数量, d 为空间维度,则樽海鞘个体的位置表示为 $X_n = [X_{n1}, X_{n2}, \dots, X_{nd}]^T, n = 1, 2, \dots, N$, 对应的食物源表示为 $F = [F_1, F_2, \dots, F_d]^T$, 在一个搜索上下界分别为 ub_d, lb_d 的搜索空间范围中进行寻优。故在 SSA 中领导者根据下式进行位置更新:

$$X_d^i = \begin{cases} F_d + c_1((ub_d - lb_d)c_2 + lb_d), & c_3 > 0.5 \\ F_d - c_1((ub_d - lb_d)c_2 + lb_d), & c_3 < 0.5 \end{cases} \quad (1)$$

式中,第 i 个领导者在 d 维空间中的位置 X_d^i 进行位置更新的因素由 d 维空间中的食物源位置 F_d 、收敛因子 c_1 、控制搜索空间大小 c_2 、搜索方向 c_3 决定。 c_2, c_3 是 $[0, 1]$ 上的随机数, c_1 是 SSA 中最重要的参数,它平衡了算法全局搜索和局部搜索,定义如下:

$$c_1 = 2e^{-(4t/T)^2} \quad (2)$$

式中, t 为当前迭代次数, T 为最大迭代次数。

追随者位置更新如下所示:

$$X_d^i = \frac{1}{2}(X_d^i + X_d^{(i-1)}) \quad (3)$$

追随者的位置更新取决于前一个个体的位置,其中 X_d^{i-1} 表示 d 维空间第 $i-1$ 个追随者的位置。

2 改进的樽海鞘算法

2.1 领导者的改进

在樽海鞘算法的领导者位置更新公式中 c_2, c_3 的随机值分别控制领导者搜索范围和方向。首先引入洛伦兹函数替代 c_2 , 提高领导者位置搜索空间的有效性;再引入天牛须搜索算法中的左右须搜索机制替代 c_3 的随机值,决定樽海鞘算法中领导者位置更新方向,使算法中领导者位置前进方向更加有利于对全局最优的搜索。

樽海鞘算法领导者位置更新中, c_2 是属于均匀分布的随机值,在算法中对其取值具有一定的盲目性,在算法迭代过程中,算法前期为寻找出全局最优解需要算法的个体尽可能地遍历整个搜索空间区域,算法后期,较小的搜索空间能一定程度跳出局部最优解,故该文引入具有缓慢下降速度的洛伦兹函数替代 c_2 值。具体函数如下:

$$H(x; x_0, r, A) = A * \left[\frac{r}{(x - x_0)^2 + r^2} \right] \quad (4)$$

$$K = H(t/T; 0, 1, 1) \quad (5)$$

式中, A 为峰值, x_0 为位置参数, r 为尺度参数, K 为通过洛伦兹函数计算出来的参数, t 为当前迭代次数, T 为最大迭代次数。

天牛须搜索算法 (beetle antennae search, BAS) 是 Jiang 等人^[12-13]通过对天牛的觅食行为模拟出来的一种仿生算法。天牛觅食的过程通过自身的左右须对食物的浓度感知情况,天牛朝着食物源浓度强的方向移动。天牛须搜索算法中天牛每次迭代后头朝向都是随机的,左右须的位置与两须之间距离和自身的位置有关,具体表示如下:

$$X(l) = X + d_0 * \text{dir}/2 \quad (6)$$

$$X(f) = X - d_0 * \text{dir}/2 \quad (7)$$

式中, $X(l)$ 、 $X(f)$ 分别表示天牛的左右须, X 表示天牛个体自身的位置, dir 表示天牛的朝向是随机的。 d_0 为左右须之间的距离,两须之间距离的大小,影响天牛在搜索空间中对最优食物源位置方向的判断,故对两须之间位置根据迭代次数进行衰减。

$$d_0 = C * d_0 \quad (8)$$

C 为衰减因子,表示为:

$$C = e^{-30(t/T)} \quad (9)$$

在樽海鞘算法中,引入天牛须搜索算法中的左右须搜索寻优方向的策略,得到一种改进的樽海鞘算法

(beetle antennae search salp swarm algorithm, BSSA)。将领导者类比为天牛,首先由式(6)、式(7)构建领导者左右须,再通过目标函数计算出领导者左右须位置的适应度值,比较左右须适应度值大小,使领导者在进行位置更新时向具有更佳的适应度值的方向移动。改进的樽海鞘算法领导者位置更新如下:

$$X_d^i = F_d - c_1 * \text{sign}(f(X(l)) - f(X(f))) * ((ub_d - lb_d) * K + lb_d) \quad (10)$$

式中, $f(X(l))$ 、 $f(X(f))$ 分别表示通过目标函数所计算出来的左须和右须的适应度值, sign 为符号函数, K 为通过洛伦兹函数计算出来的参数。

2.2 追随者的改进

该文将自适应惯性权重引入到追随者位置更新中改进樽海鞘算法 (adaptive weighted salp swarm algorithm, ASSA)。SSA 中追随者的位置更新取决于自身的位置和上一个个体的位置,在上一个个体位置较差或是陷入局部最优解时,追随者盲目的跟随,将会使算法陷入局部最优。文献[14]提出权重值的大小对全局搜索空间寻找最优解和跳出局部最优解均有不同程度的影响。较大的权重值有助于算法在全局搜索空间中找到最优解,而较低的权重值则有助于其跳出局部最优解。故该文在追随者位置更新中引入自适应惯性权重,决定了自身个体和上一个个体对位置更新的影响程度。

追随者位置更新如下所示:

$$X_d^i = \frac{1}{2} (X_d^i + \omega * (X_d^{(i-1)} - X_d^i) * \text{rand}) \quad (11)$$

$$\omega = \omega_{\max} * (\omega_{\max} - \omega_{\min}) * (t - T) / T \quad (12)$$

式中, rand 为 $[0, 1]$ 上的随机值, ω 为惯性权重, $\omega_{\max} = 0.9$, $\omega_{\min} = 0.4$, 自适应惯性权重值主要根据当前迭代次数 t 与最大迭代次数 T 决定。追随者位置的改进,进一步提升了算法的局部搜索能力。

2.3 BASSA 伪代码

(1) 设置 BASSA 参数: 最大迭代次数 T , 两须距离 d_0 , 洛伦兹函数相关参数, 种群数量大小 N 等。

(2) 随机初始化 BASSA 种群, 计算每个樽海鞘位置的适应度值, 将最佳的适应度值对应的位置作为食物源位置 X_{best} 。

(3) While($t < T$)

(4) For($i = 1 : N$)

(5) if($i < N/2$)

(6) 根据式(4)、(5)得到洛伦兹函数值。

(7) 根据式(6)、(7)计算得到领导者位置的左右须。

(8) 根据式(10)更新领导者位置。

(9) else

(10) 根据式(11)更新追随者位置。

(11) end if

(12) end for

(13) 通过目标函数计算樽海鞘算法位置更新后的种群个体的适应度值。

(14) 选取最佳适应度值对应的樽海鞘个体位置与食物源位置进行比较, 并更新食物源位置。

(15) $t = t + 1$

(16) end while

(17) 返回最佳食物源位置 X_{best} 及其适应度值。

2.4 BASSA 算法的时间复杂度分析

SSA 在一个 $N \times D$ 维的空间内优化问题的时间复杂度分析如下: 在种群初始化设置所产生的时间为 θ_1 , 设均匀分布随机值在 D 维空间中每一维度产生时间为 η_1 , SSA 中通过目标函数计算每个个体位置的适应度值的时间为 λ_1 , 则求解初始化后所有位置的适应度值所需要的时间为 $N * \lambda_1$ 。排序得到食物源位置的时间复杂度为 η_2 , 故在未进入迭代前所产生的时间为 $o(ND)$ 。

在迭代过程中, 领导者部分种群数量为 $N/2$, 由公式(2)计算 c_1 所产生的时间为 θ_2 , c_1 每次迭代运行 1 次, c_2 、 c_3 是属于均匀分布的随机值则运行时间为 η_1 , 故领导者位置的时间复杂度表示为 $o(\frac{1}{2}ND)$, 在追随者部分种群的数量为 $N/2$, 则追随者部分时间复杂度表示为 $o(\frac{1}{2}ND)$ 。因此标准的 SSA 算法表示为 $o(ND)$ 。

文中改进的 BASSA, 在初始化种群阶段与 SSA 一致, 时间复杂度为 $o(ND)$, 在领导者部分增加了洛伦兹函数和天牛须搜索策略分别替代 c_2 、 c_3 。洛伦兹函数由公式(4)知, 每次迭代只运行一次, 所产生的时间为 θ_3 , 天牛须搜索策略由公式(6)、(7)知, 每一个领导者在每一维所产生左右须的时间分别为 d_1 、 d_2 , 所有领导者产生左右须的时间为 $\frac{1}{2}ND * (d_1 + d_2)$, c_1 与标准 SSA 相同产生的时间为 θ_2 , 故 BASSA 在领导者位置更新阶段时间复杂度表示为:

$$T_1 = o(\theta_2 + \theta_3 + \frac{1}{2}ND * (d_1 + d_2) + \frac{1}{2}ND) = o(ND) \quad (13)$$

在追随者位置更新, 引入的自适应惯性权重在每次迭代中只运行一次, 所需要时间为 η_2 。故 BASSA 追随者部分时间复杂度为:

$$T_2 = o(\eta_2 + \frac{1}{2}ND) = o(\frac{1}{2}ND) \quad (14)$$

综上, 改进的 BASSA 的整体时间复杂度为:

$$T = o(ND) + T_1 + T_2 = o(ND) \quad (15)$$

故改进的 BASSA 与 SSA 相比,没有增加时间复杂度。

3 算法仿真与分析

3.1 实验设计

为验证提出的 BASSA 的有效性和鲁棒性,在 8 个包含单峰、多峰、可分、不可分等不同特征的测试函数上对 BASSA、BSSA、ASSA、文献[10]的 RDSSA、文献[11]中的 ESSA 以及基本的 SSA 进行测试。实验设置为在相同的迭代次数下,对比各算法在同一测试函数中的收敛速度和收敛精度。对 BASSA 算法设置的

参数如表 1 所示, RDSSA、ESSA 所需参数根据相关文献中设置相同。测试函数如表 2 所示。

表 1 参数设置

最大迭代 次数 T	种群 数量 N	洛伦兹函数 相关参数	两须之间的 初始距离 d_0
500	30	$x_0 = 0; r = 1;$ $A = 1$	$d_0 = ((ub_d - lb_d) +$ $lb_d)$

本实验在 Matlab16b 环境下采用 M 语言进行,同一算法对相同的测试函数独立运行 50 次,对每次运行结果的最优值进行统计,并计算其平均值和标准差。具体实验结果如表 3 所示。

表 2 测试函数

函数表达式	取值范围	最值
$f_1(x) = \sum_{i=1}^x x_i^2$	$[-100, 100]$	0
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	$[-10, 10]$	0
$f_3(x) = \max(x_i , 1 \leq i \leq n)$	$[-100, 100]$	0
$f_4(x) = \sum_{i=1}^n x_i \sin(x_i) + 0.1 x_i $	$[-10, 10]$	0
$f_5(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.12, 5.12]$	0
$f_6(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	$[-32, 32]$	0
$f_7(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]$	0
$f_8(x) = 1 + 0.1 \sqrt{\sum_{i=1}^n x_i^2} - \cos\left(2\pi \sqrt{\sum_{i=1}^n x_i^2}\right)$	$[-100, 100]$	0

3.2 结果分析

3.2.1 与原算法 SSA 的比较分析

通过图 1 中各算法对不同的函数优化的收敛曲线可以看出,经过改进的 BSSA、ASSA、BASSA 与 SSA 相比,除了在 f_5 函数上, BSSA 与 SSA 基本持平,在其余优化函数上改进的樽海鞘算法在收敛速度和收敛精度上均优于 SSA。改进的 BSSA 与 SSA 相比,后期的寻优精度上具有明显的优势。改进的 ASSA 与 SSA 相比, ASSA 在迭代前期能够快速收敛,并在迭代后期依旧具有较高的收敛速度,且收敛精度明显高于 SSA。BASSA 综合了 BSSA 和 ASSA 的优势后,在收敛速度和收敛精度上,表现出了明显的优势。

经过表 3 中的 BASSA、BSSA、ASSA、SSA 在相同的迭代次数下,50 次最佳适应度值 Best, 平均最佳适应度值 Mean, 最佳适应度值标准差 Std 的具体实验值表明,在单峰测试函数中改进的 BASSA 最高达到了

E-190 级别,其次 ASSA、BSSA,相比 SSA,寻优效果均具有更加明显的优势。在多峰测试函数中, BASSA 和 ASSA 均能达到理论最优值,而 BSSA 相比于 SSA 的寻优精度更高。通过表 3 中的平均值、标准差值, BASSA、BSSA、ASSA 在对相同的函数进行优化时均为最小。表明 BASSA 具有一定的鲁棒性。

3.2.2 与已改进 SSA 的比较分析

将该文改进的 BASSA 与文献[10]的 RDSSA 和文献[11]中的 ESSA 进行对比。通过观察图 1 的收敛曲线,在单峰测试函数中改进的 BASSA 与 ESSA 以及 RDSSA 相比,在迭代前期,具有更快的收敛速度,而在迭代后期, BASSA 的收敛速度依旧较快,收敛至一个最优的结果。在多峰测试函数中, BASSA 与 ESSA、RDSSA 均能达到理论最优值,但 BASSA 达到理论最优值所需迭代次数最少。

通过表 3 相关算法的具体测试结果中, f_5 、 f_7 、 f_8

三者算法在收敛精度和标准差上相当,在 f_6 上 BASSA、RDSSA 表现相当且在平均最佳值和标准差上

优于 ESSA。在其他测试函数上,BASSA 均表现为更佳收敛能力和稳定性。

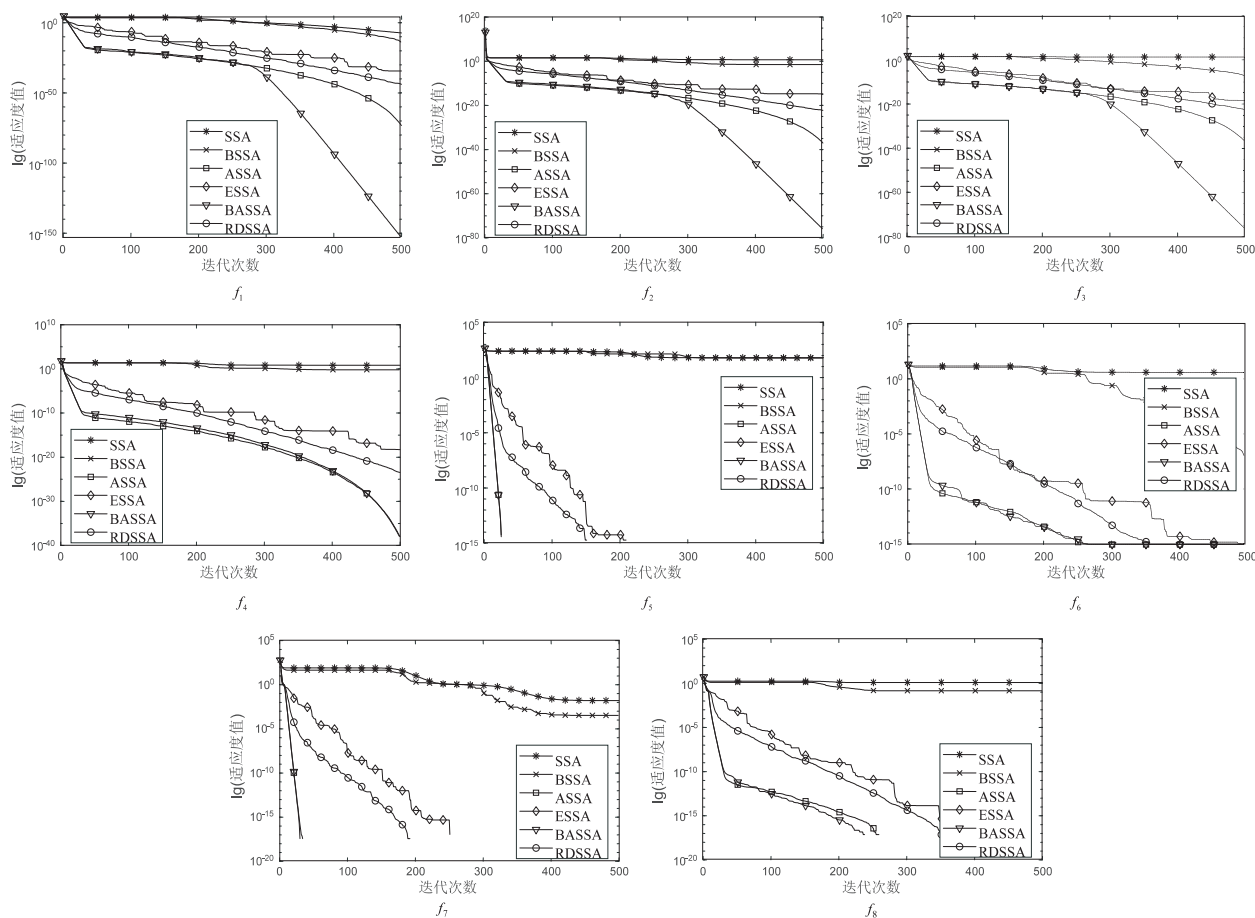


图1 各算法平均收敛曲线比较

表3 各算法实验测试结果

函数序号	指标	SSA	BSSA	ASSA	BASSA	ESSA	RDSSA
f_1	Best	1.83e-08	4.75e-15	1.45e-74	1.66e-190	2.55e-69	6.93e-47
	Mean	8.98e-08	1.71e-14	1.69e-73	7.49e-151	1.46e-35	4.47e-45
	Std	7.81e-08	8.77e-15	1.71e-73	3.88e-150	2.30e-34	2.44e-44
f_2	Best	4.99e-01	2.95e-08	3.84e-38	1.94e-100	3.32e-45	6.05e-24
	Mean	3.17e+00	1.15e-02	1.87e-37	2.72e-78	1.72e-15	2.03e-23
	Std	2.26e+00	5.10e-02	7.55e-38	9.58e-78	9.17e-15	5.47e-23
f_3	Best	8.37e+00	2.27e-08	8.46e-38	2.58e-111	1.21e-34	2.80e-24
	Mean	1.67e+01	4.43e-08	1.61e-37	3.39e-76	1.07e-19	2.67e-23
	Std	4.40e+00	8.46e-09	6.49e-38	1.78e-75	4.42e-18	2.64e-23
f_4	Best	2.52e+00	2.95e-02	3.21e-38	4.71e-39	7.42e-37	7.60e-26
	Mean	6.12e+00	7.92e-01	1.72e-38	1.68e-38	1.18e-19	9.22e-25
	Std	2.64e+00	2.03e+00	1.02e-38	9.43e-39	3.26e-19	7.12e-25
f_5	Best	2.58e+01	1.39e+01	0	0	0	0
	Mean	6.26e+01	6.09e+01	0	0	0	0
	Std	1.84e+01	2.67e+01	0	0	0	0
f_6	Best	1.77e+00	1.51e-08	8.88e-16	8.88e-16	8.88e-16	8.88e-16
	Mean	4.23e+00	2.68e-08	8.88e-16	8.88e-16	1.59e-15	8.88e-16
	Std	1.06e+00	8.42e-09	0	0	6.48e-16	0

续表 3

函数序号	指标	SSA	BSSA	ASSA	BASSA	ESSA	RDSSA
f_7	Best	2.53e-04	1.48e-14	0	0	0	0
	Mean	1.21e-02	1.26e-04	0	0	0	0
	Std	1.61e-02	1.80e-03	0	0	0	0
f_8	Best	8.99e-01	9.98e-02	0	0	0	0
	Mean	1.20e+00	1.36e-01	0	0	0	0
	Std	1.97e-01	4.90e-02	0	0	0	0

4 结束语

该文提出的 BASSA 在标准的樽海鞘算法的领导者位置引入天牛须搜索机制使樽海鞘领导者向着更优的方向搜索最优解,引入洛伦兹函数控制算法的搜索空间,加快了收敛速度,在追随者位置引入自适应惯性权重改善了算法的局部搜索能力。通过对不同的基准测试函数优化,实验结果表明:改进的 BASSA 相较于 SSA、已经改进的 ESSA、RDSSA 具有更快的收敛速度,在相同的测试条件下,能够达到更高的收敛精度,并在可行性以及鲁棒性方面都更具有优势。在下一步研究中,尝试将改进的算法应用到具体的工程中。

参考文献:

- [1] 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(6): 163-191.
- [2] KENNEDY J, EBERHART R C. Particle swarm optimization [C]//Proc of ICNN'95-IEEE international conference on neural networks. Perth: IEEE, 1995: 1942-1948.
- [3] DORIGO M, GAMBARDELLA L M. Ant colony system: a cooperative learning approach to the traveling salesman problem [J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 53-66.
- [4] TUBISHAT M, IDRIS N, SHUIB L, et al. Improved salp swarm algorithm based on opposition based learning and novel local search algorithm for feature selection [J]. Expert Systems with Applications, 2020, 145: 113122.
- [5] ZHAO X Q, YANG F, HAN Y Z, et al. An opposition-based chaotic salp swarm algorithm for global optimization [J]. IEEE Access, 2020, 8: 36485-36501.
- [6] 安小宇, 鲁奎豪, 崔光照. 基于改进樽海鞘优化 BP 神经网络的土壤墒情预测 [J]. 中国农机化学报, 2019, 40(11): 124-130.
- [7] 陈涛, 王梦馨, 黄湘松. 基于樽海鞘群算法的无源时差定位 [J]. 电子与信息学报, 2018, 40(7): 1591-1597.
- [8] HOSSAM F, MAJDI M M, ALI A H, et al. An efficient binary salp swarm algorithm with crossover scheme for feature selection problems [J]. Knowledge-Based Systems, 2018, 154: 43-67.
- [9] SAYED G I, KHORIBA G, HAGGAG M H. A novel chaotic salp swarm algorithm for global optimization and feature selection [J]. Applied Intelligence, 2018, 48(10): 3462-3481.
- [10] SANTOSH K M, ABHILASH M, ROSY P. A chaotic salp swarm algorithm based on quadratic integrate and fire neural model for function optimization [J]. Progress in Artificial Intelligence, 2019, 8(3): 343-358.
- [11] MOHAMMED H Q, HANY M H, SAAD A. Enhanced salp swarm algorithm: application to variable speed wind generators [J]. Engineering Applications of Artificial Intelligence, 2019, 80: 82-96.
- [12] JIANG X Y, LI S. BAS: beetle antennae search algorithm for optimization problems [EB/OL]. (2017-10-30) [2020-06-19]. <https://arxiv.org/abs/1710.10724>.
- [13] JIANG X Y, LI S. Beetle antennae search without parameter tuning (BAS-WPT) for multi-objective optimization [EB/OL]. (2017-11-07) [2020-06-19]. <https://arxiv.org/abs/1711.02395>.
- [14] ZHOU Z, SHI Y. Inertia weight adaption in particle swarm optimization algorithm [C]//Advances in swarm intelligence. Chongqing: Springer, 2011: 71-79.