

文章编号:1672-3961(2025)03-0088-12

DOI:10.6040/j.issn.1672-3961.0.2024.075

增强型白鲸优化算法及其应用

文裕杰,张达敏*

(贵州大学大数据与信息工程学院,贵州 贵阳 550025)

摘要:针对白鲸优化算法搜索效率不足、易陷入局部极值的问题,提出增强型白鲸优化算法(enhance beluga whale optimization, EBWO)。加入基于权重的抢食型白鲸并应用于算法的开发阶段,丰富该阶段的位置更新方式,利用贪婪机制选择更优位置,提高解的质量;引入自适应高斯策略对鲸坠阶段的白鲸局部扰动,使其调整至最优位置附近,加快算法的收敛速度;使用凸透镜成像学习策略对信息共享后的位置做反向处理,提高算法跳出局部最优值的能力。通过对10个基准测试函数和CEC2020测试集的寻优对比分析,以及Wilcoxon秩和检验可知,EBWO的寻优速度和收敛精度都得到较大提升。为了验证EBWO算法的实用性和可行性,将其应用到减速器和压力容器工程设计的求解,通过试验分析可知,EBWO算法在解决实际优化问题上具有一定的优越性。

关键词:白鲸优化算法;抢食白鲸;高斯扰动;凸透镜成像;工程应用

中图分类号:TP301.6 **文献标志码:**A

引用格式:文裕杰,张达敏.增强型白鲸优化算法及其应用[J].山东大学学报(工学版),2025,55(3):88-99.

WEN Yujie, ZHANG Damin. Enhanced beluga whale optimization algorithm and its application[J]. Journal of Shandong University (Engineering Science), 2025, 55(3):88-99.

Enhanced beluga whale optimization algorithm and its application

WEN Yujie, ZHANG Damin*

(College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, Guizhou, China)

Abstract: Aiming at overcoming drawbacks of insufficient search efficiency and tendency to slip into local extremes of beluga optimization algorithm, an enhanced beluga whale optimization (EBWO) algorithm was proposed in this paper. First, a weight-based scramble beluga was included and applied to the algorithm's development phase to enrich the position updating technique, and a greedy mechanism was employed to select a better location and increase the quality of the understanding. Second, an adaptive Gaussian strategy was introduced to locally perturb the beluga in the whale falling phase, to make it adjusted to the vicinity of the optimal position to improve the convergence speed of the algorithm. Finally, a convex lens imaging learning strategy was used to carry out the information position after sharing. The comparative examination of the optimization of the ten benchmark test functions, the CEC2020 test set, and the Wilcoxon rank sum test revealed that EBWO's optimization speed and convergence accuracy had significantly improved. To test the EBWO algorithm's practicality and feasibility, it was applied to solve engineering design problems involving speed reducers and pressure vessels. It was discovered through experimental comparative analysis that the EBWO algorithm had a certain degree of superiority in solving actual optimization problems.

Keywords: beluga whale optimization; scramble beluga; Gaussian variation; convex lens imaging learning; engineering optimization

0 引言

最优化问题是人工智能、深度学习、无人驾驶、

工业控制等领域的重要问题之一,其参数值的选择影响最终性能。由于这类问题属于黑盒问题,目前该问题仍在深入研究中^[1]。传统的优化方法有梯度下降法^[2]、牛顿法^[3]等,随着最优化问题规模增

收稿日期:2024-04-12

基金项目:国家自然科学基金资助项目(62166006)

第一作者简介:文裕杰(1999—),男,贵州凯里人,硕士研究生,主要研究方向为智能优化与工业互联网安全。E-mail:3596682950@qq.com

* 通信作者简介:张达敏(1967—),男,贵州贵阳人,教授,硕士生导师,博士,主要研究方向为智能优化与计算机应用技术。

E-mail:1203813362@qq.com

大和复杂性增加,以上方法的求解效率变低且耗时严重^[4]。相比于传统方法,元启发式算法在最优化问题中显现出优异的性能,其算法模拟自然界动物行为或各类物理现象,具有适用性强、求解效率高和计算成本低的特点。在元启发式算法的研究热潮中,研究者提出一系列高效的优化算法,如变色龙优化算法^[5]、非洲秃鹫算法^[6]、蜣螂优化算法^[7]、沙猫优化算法^[8]、阿奎拉鹰算法^[9]等。

白鲸优化算法 (beluga whale optimization, BWO) 是 2022 年提出的一种新型启发式优化算法,模拟白鲸游泳、捕食和鲸鱼坠落的行为。文献[10]将 BWO 与其他元启发式算法对比,BWO 表现出寻优能力强、参数调节少和免推导机制等优势,在实际应用中这不仅能提高优化问题的精度,还能节省计算成本。基于白鲸优化算法的优异性能,文献[11]应用 BWO 优化噪声测量值的偏差,减少 Kalman 滤波的影响;文献[12]使用 BWO 优化新能源场站的功率。此外,白鲸优化算法还被应用于其他领域^[13-14]。

为提升 BWO 求解最优化问题的能力,研究者采用了多种有效的方法。文献[15]提出群体聚合策略和迁移策略,通过种群的聚合度来转换探索和开发阶段,并使用迁移策略为算法提供潜在解;文献[16]使用 Fuch 映射分别对初始化阶段和开发阶段进行扰动,以此丰富种群多样性和协调探索,并引入动态反向学习策略提升算法的抗停滞能力;文献[17]提出一种基于混沌反向学习和水波算法改进的白鲸优化算法,算法使用混沌映射和反向学习策略提高初始种群的多样性,且引入水波算法的折射操作来避免最优个体陷入局部最优;文献[18]将适应度距离平衡策略融入 BWO 的探索和鲸坠阶段以跳出局部最优,然后利用切线飞行的方式改进鲸坠阶段的位置更新方式,提升算法的收敛精度。

上述研究虽在一定程度上考虑了白鲸种群多样性和陷入局部最优情况,但增加了算法的复杂性且对算法的性能提升有限。为了更好的提高白鲸优化算法的寻优性能,提升算法的收敛速度和精度,本研究提出了一种增强型白鲸优化算法 (enhance beluga whale optimization, EBWO)。EBWO 采取了以下 3 项改进策略:(1)采用组合算法的思想在开发阶段引入“抢食型白鲸”,并基于最差位置和最优位置的重要性分别赋予权重,在丰富算法开发位置方式的同时,提高算法的精度;(2)使用自适应高斯扰动策略改进鲸鱼坠落方式,使其在最优值附近局部调整,提高算法的收敛速度;(3)引入一种可利用信息交流的透镜成像学习策略,利用当前个体与周围个体进行信息分享后的位置得到

透镜折射反向位置,增大算法跳出局部极值的概率。

本研究将 EBWO 与其他经典的群智能算法,以及其他改进的白鲸优化算法在基准测试函数、CEC2020 测试集上测试性能,并对使用的改进策略进行消融试验。试验结果表明,EBWO 的整体寻优性能突出。最后,将 EBWO 应用于 2 个现实工程设计问题,结果表明,所提出的增强型白鲸优化算法具有更好的设计结果。

1 白鲸优化算法简介

白鲸优化算法^[10]由白鲸游泳、捕食和坠落 3 种行为得到启发,对应算法的探索、开发和鲸坠阶段。白鲸种群在探索和开发过程之间的转换由自适应平衡因子 B_f 决定, B_f 的表达式为

$$B_f = B_0 \cdot [1 - T / (2 \cdot T_{\max})], \quad (1)$$

式中, B_0 为(0,1)范围内的随机数, T 和 T_{\max} 分别为当前迭代次数和最大迭代次数。当 $B_f > 0.5$,种群处于探索阶段;当 $B_f \leq 0.5$,种群处于开发阶段。

1.1 探索阶段

BWO 算法在探索阶段的位置更新由镜像距离游泳的成对白鲸共同决定,数学模型为

$$X_{i,j}^{T+1} = \begin{cases} X_{i,j}^T + (X_{r,P_1}^T - X_{i,P_1}^T) (1+r_1) \sin(2\pi r_2), & j = \text{even} \\ X_{i,P_1}^T + (X_{r,P_1}^T - X_{i,P_1}^T) (1+r_1) \cos(2\pi r_2), & j = \text{odd} \end{cases}, \quad (2)$$

式中: $X_{i,j}^{T+1}$ 为第 i 条白鲸在第 j 维更新后的位置; X_{i,P_j}^T 为该维度下第 i 条白鲸的位置,其中 $P_j (j = 1, 2, \dots, d)$ 表示从 d 维中选取的随机维度; X_{r,P_1}^T 为种群中随机选取的第 r 条白鲸; r_1 和 r_2 为(0,1)范围内的随机数;由偶数 (even) 和奇数 (odd) 选出镜像游泳的白鲸, $\sin(2\pi r_2)$ 和 $\cos(2\pi r_2)$ 用于模拟成对白鲸游泳时镜像的鳍。

1.2 开发阶段

开发阶段的位置更新模拟着白鲸的捕食行为。该行为下的最优个体为 X_{best}^T ,当前个体 X_i^T 和随机个体 X_r^T 共享彼此间的位置信息进行合作觅食。同时还引入了 Levy 飞行策略用于增强收敛性。该阶段的数学模型为

$$X_i^{T+1} = r_3 X_{\text{best}}^T - r_4 X_i^T + C_1 \cdot L_F (X_r^T - X_i^T), \quad (3)$$

式中, r_3, r_4 均为(0,1)范围内的随机数, $C_1 = 2 \cdot r_4 \cdot (1 - T / T_{\max})$ 是用于衡量 Levy 飞行随机跳跃程度的因子, L_F 为服从 Levy 分布的随机数。 L_F 表达式为

$$L_F = 0.05 \times \frac{u \sigma}{|v|^{1/\beta}}, \quad (4)$$

$$\text{式中: } \sigma = \left[\frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{(\beta-1)/2}} \right]^{1/\beta}, \text{ 其中 } \Gamma(\cdot) \text{ 为}$$

gamma 函数, $\beta = 1.5$; u 和 v 为服从正态分布的随机数。

1.3 鲸坠阶段

在自然环境中,白鲸可能受到虎鲸、北极熊等生命威胁而落入深海,为维持种群数量,使用当前白鲸 X_i^T 和随机白鲸 X_r^T 的位置,以及下落的步长来更新个体,数学模型为

$$X_i^{T+1} = r_5 X_i^T - r_6 X_r^T + r_7 X_{\text{step}}, \quad (5)$$

$$X_{\text{step}} = (u_b - l_b) \exp\left(-\frac{C_2 \cdot T}{T_{\max}}\right), \quad (6)$$

式中: r_5, r_6, r_7 均为 $(0, 1)$ 范围内的随机数; X_{step} 为白鲸下坠的步长; u_b 和 l_b 分别为搜索区域的上下界; $C_2 = 2 \cdot W_f \cdot n$ 是阶跃因子,其值与白鲸下坠概率 W_f 和种群规模 n 有关, $W_f = 0.1 - 0.05T/T_{\max}$, W_f 由迭代初始的 0.1 线性变化到最终的 0.05,表明白鲸越接近食物,死亡概率越小。

由上述白鲸优化算法的寻优过程可知,该算法 3 个阶段的个体种类较少,种群的丰富度不足,算法的收敛精度有限;同时,鲸坠阶段的调整变化范围大,不利于算法收敛。因此,本研究提出一种增强型白鲸优化算法针对以上问题进行改善。

2 增强型白鲸优化算法

2.1 基于权重的抢食型白鲸

蜣螂优化算法^[7]中的滚球蜣螂、繁育蜣螂、小蜣螂和偷窃蜣螂 4 种蜣螂采用不同的位置更新方式,共同寻找问题的最优解。其中,偷窃蜣螂通过偷取其他蜣螂的粪球更新自身位置,数学模型为

$$X_i^{T+1} = X_{\text{best}}^T + S \cdot \mathbf{g} \cdot (|X_i^T - X_{\text{worst}}^T| + |X_i^T - X_{\text{best}}^T|), \quad (7)$$

式中, S 为常量, \mathbf{g} 是服从正态分布的 $1 \times D$ 的随机向量, X_{worst}^T 是当前最差的个体。偷窃蜣螂通过计算当前位置与最差位置、最优位置的距离来不断靠近全局最优位置。更新位置时考虑多种类型的信息可以增加个体的位置丰富度, BWO 的开发过程忽略了当前最差位置信息,而偷窃蜣螂的更新方式恰恰弥补了这一不足。因此基于组合算法的思想,本研究将偷窃蜣螂行为引入 BWO 的开发阶段,丰富开发阶段位置更新方式的同时提高算法的收敛精度。

观察到式(7)中虽考虑到当前位置与最差位置、全局最优位置的距离,但这 2 个距离的权重相同。靠近全局最优位置可帮助算法收敛,相反,靠

近最差位置会增加算法收敛的时间。显然,当前位置与最差位置、全局最优位置的这 2 个距离的权重应当是不同的。通过大量试验得出,当前位置与最差位置的权重为 0.3、当前位置与全局最优位置的权重为 0.7 时, BWO 的表现较好。因此,式(7)更新为

$$X_i^{T+1} = X_{\text{best}}^T + S \cdot \mathbf{g} \cdot (0.3 |X_i^T - X_{\text{worst}}^T| + 0.7 |X_i^T - X_{\text{best}}^T|). \quad (8)$$

常量 S 影响算法的性能。 S 的不同取值在基准测试函数中 Step 函数的表现如图 1 所示。

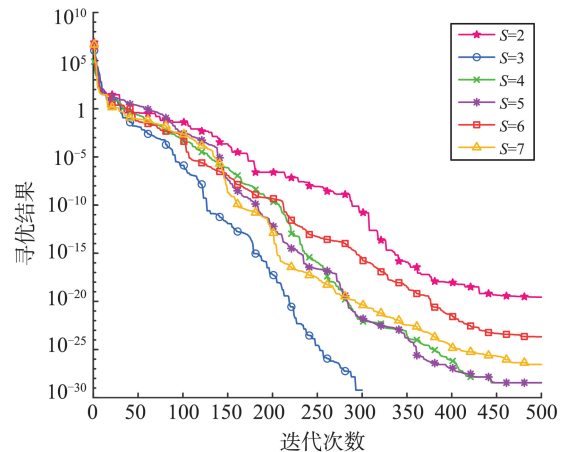


图 1 S 取不同值的性能表现

Fig.1 Performance for different values of S

由图 1 可知, $S = 3$ 时,算法的性能最优。故在开发阶段, BWO 同时进行式(3)和式(8)的位置更新,并由贪婪机制选择最优位置。

2.2 自适应的高斯扰动

鲸坠阶段的目的是维持白鲸种群的数量,对部分白鲸位置局部调整。假设在搜索上界 $u_b = 100$, 下界 $l_b = -100$ 的范围内,局部调整后的白鲸位置信息如图 2 所示。式(5)的调整结果如图 2(a)所示,由图 2(a)可知:有部分白鲸分布于搜索范围的边界区域(最优值为 0),随机白鲸和步长的设置虽增加位置信息,但调整后的位置均离最优值较远。

高斯变异扰动以较大的概率产生较小的变异值,在小范围内具有良好的搜索能力^[19]。随着寻优的进行,为防止个体间的同化程度增大,应使高斯扰动的能力逐渐提高。经自适应的高斯扰动改进后的鲸坠阶段数学模型为

$$X_i^{T+1} = X_i^T \cdot (T/T_{\max}) \cdot \text{Gauss}(0, 1), \quad (9)$$

式中, $\text{Gauss}(0, 1)$ 为满足标准高斯分布的随机变量。通过式(9)进行局部调整后的白鲸位置信息如图 2(b)所示,与式(5)的调整方式相比,所提出方法调整后的位置分布于最优值附近,提供变异调整的同时不降低算法的精度。

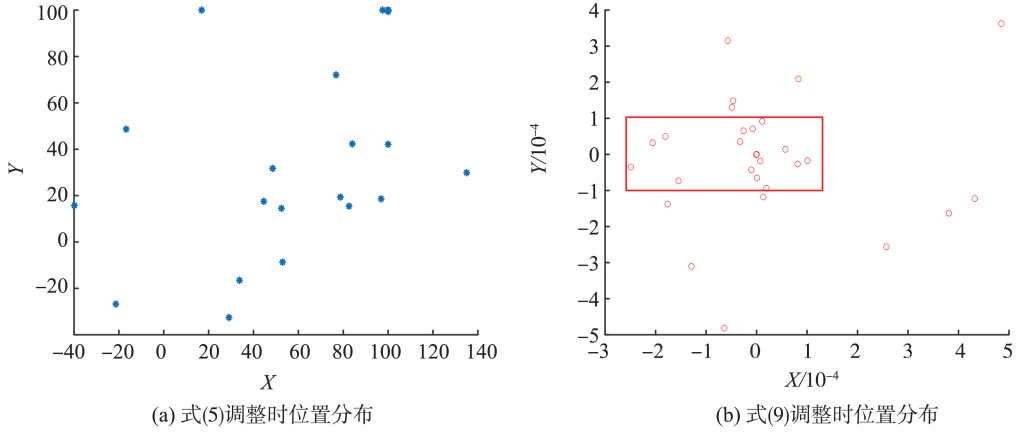


图2 采用不同调整方式后白鲸的位置分布
Fig.2 Position distribution of beluga whales with different adjustment methods

2.3 凸透镜成像反向学习

BWO算法寻优的后期,由于逐渐靠近当前最佳位置,白鲸群体间的多样性降低,若当前位置不是全局最优位置,算法将陷入局部最优。为解决这一问题,引入凸透镜成像反向学习策略^[20]。

根据光通过凸透镜发生折射的原理产生一个反向位置来扩展位置信息。透镜成像反向学习原理如图3所示,以二维搜索空间为例,设搜索范围为 (a, b) ,凸透镜放置在搜索范围的中点 $(a+b)/2$ 处,假设物体A在横轴上的投影为 x ,高度为 L ,通过凸透镜成像原理得到高度为 L^* 的像 A^* ,在横轴上的投影为 x^* 。

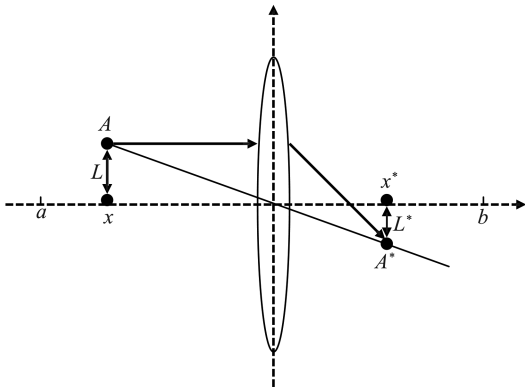


图3 透镜成像反向学习原理图

Fig.3 Lens imaging inverse learning schematic

透镜成像反向学习策略的数学模型为

$$\frac{(a+b)/2-x}{x^*-(a+b)/2} = \frac{L}{L^*} \quad (10)$$

设 k 为缩放因子,令 $k=L/L^*$,则凸透镜成像反向位置 x^* 的表达式为

$$x^* = \frac{a+b}{2} + \frac{a+b}{2k} - \frac{x}{k} \quad (11)$$

$k=1$ 即为一般反向学习方式。本研究的 k 为 1.2×10^{12} 。

为提高反向解的质量,使用当前迭代个体 X_i^T 与周围个体(X_{i-1}^T 和 X_{i-2}^T)进行信息交流后的位置 x 进行凸透镜反向学习, x 的数学表达式为

$$x = (X_i^T + X_{i-1}^T + X_{i-2}^T) / 3 \quad (12)$$

2.4 EBWO伪代码

算法1 增强型白鲸优化算法

输入 种群规模 N 、维度 d 、总迭代次数 T_{max}

输出 最优位置 X_{best} 、最优适应度值 F_{best}

- (a) BEGIN
- (b) 初始化种群,计算白鲸适应度值并记录最优个体
- (c) while ($T < T_{max}$) do
- (d) 计算平衡因子 B_f 和鲸坠概率 W_f
- (e) for $i = 1$ to N do
- (f) if $B_f > 0.5$ then
- (g) 根据式(2)更新白鲸个体的探索位置
- (h) else
- (i) 根据式(3)和式(8)计算白鲸和抢食白鲸位置,并由贪婪机制选择最佳白鲸
- (j) end if
- (k) if $B_f < W_f$ then
- (l) 根据式(9)进行自适应高斯扰动的鲸坠阶段
- (m) end if
- (n) end for
- (o) for $i = 1$ to N do
- (p) 根据式(11)得出凸透镜成像反向位置
- (q) end for
- (r) 更新当前最优位置 X_{best} 及适应度 F_{best}
- (s) $T = T + 1$

- (t) if $T > T_{\max}$ 则结束迭代寻优
 (u) end while
 (v) 输出最优解 X_{best} 和问题最优适应度值 F_{best}
 (w) END

2.5 时间复杂度分析

时间复杂度 O 是评价算法性能的一个重要指标。设白鲸种群数量为 N , 搜索空间维度为 d , 总迭代次数为 T_{\max} 。BWO 的时间复杂度取决于种群初始化 $O_1(N \times d)$, 迭代寻优部分 $O_2(N \times d \times T_{\max})$ 以及受平衡因子 B_f 和鲸坠概率 W_f 影响的鲸坠阶段 $O_3(0.1 \times N \times d \times T_{\max})$ 三个部分, BWO 的时间复杂度为 $O_4 = O_1 + O_2 + O_3 = O(N \times d \times T_{\max})$ 。

EBWO 的时间复杂度取决于种群初始化 $O_5(N \times d)$, 基于权重的抢食白鲸位于迭代寻优部分 $O_6(N \times d \times T_{\max})$, 自适应高斯扰动的鲸坠阶段 $O_7(0.1 \times N \times d \times T_{\max})$ 以及凸透镜成像反向学习 $O_8(N \times d \times T_{\max})$, 则 EBWO 的时间复杂度为 $O_9 = O_5 + O_6 + O_7 + O_8 = O(N \times d \times (1 + 2.1T_{\max})) = O(N \times d \times T_{\max})$ 。

可见, 对 BWO 采取的改进方法未增加算法的复杂度。

3 仿真试验与结果分析

3.1 试验环境与参数说明

测试环境: Intel(R) Core(TM) i5-8300H CPU @ 2.30 GHz ~ 2.30 GHz, 8 GB 内存, 64bit Windows 11 操作系统, MATLAB2021(b)。

为保证公平性, 试验中所使用算法的种群数量均设为 30, 总迭代次数设为 500。采用的蜣螂优化算法 (dung beetle optimizer, DBO)^[7]、基于 Fuch 映射的改进白鲸优化算法 (chaotic parameter optimization enhancements beluga whale optimization, CPOEBWO)^[16]、鲸鱼优化算法 (whale optimization algorithm, WOA)^[21]、粒子群优化算法 (particle swarm optimization, PSO)^[22] 对比算法的参数设置如表 1 所示。

表 1 参数设置
Table 1 Parameters setting

算法	参数
DBO	偏转系数 $K \in (0, 0.2]$, 控制系数 $b \in (0, 1)$
CPOEBWO	映射区域 $H_n \in (-1, 1)$
WOA	转换因子 $p = 0.5$, 攻击因子 $b = 1$
PSO	惯性因子 $\omega = 0.1$, 学习因子 $c_1 = 2, c_2 = 2$

3.2 基准测试函数

为验证 EBWO 算法性能, 选取 10 个基准测试函数进行寻优, 函数信息如表 2 所示, $f_1 \sim f_6$ 为单峰函数 (30 维), $f_7 \sim f_{10}$ 为多峰函数 (30 维)。

表 2 基准测试函数

Table 2 Benchmark function

序号	测试函数名	搜索范围	最优值
f_1	Sphere	$[-100, 100]$	0
f_2	Schwefel's 2.22	$[-10, 10]$	0
f_3	Powell Sum	$[-1, 1]$	0
f_4	Schwefel's 1.2	$[-100, 100]$	0
f_5	Rosenbrock	$[-30, 30]$	0
f_6	Step	$[-100, 100]$	0
f_7	Rastrigin	$[-5.2, 5.2]$	0
f_8	Ackley 1	$[-32, 32]$	0
f_9	Penalized	$[-50, 50]$	0
f_{10}	Penalized2	$[-50, 50]$	0

3.3 改进策略有效性分析

为了验证、分析所提出的改进策略对 BWO 性能的影响, 本节将三项策略分别与 BWO 进行比较。其中, 引入基于权重的抢食白鲸记为 Scramble-BWO; 自适应高斯扰动策略记为 Gauss-BWO; 凸透镜成像反向学习策略记为 Lens-BWO。

平均绝对误差 (mean absolute error, MAE) 可以有效的反映出算法的最优值与理论最优值之间的差距^[23], 在基准测试函数下的平均绝对误差 I_{MAE} 的计算公式为

$$I_{\text{MAE}} = \frac{1}{n} \sum_{i=1}^n |m_i - o_i|, \quad (13)$$

式中, n 为基准测试函数的个数, m_i 为算法求得最优结果的平均值, o_i 为对应基准测试函数的理论最优值。

EBWO 与三项策略分别单独运行 30 次取平均值的 I_{MAE} 结果如表 3 所示。由表 3 可知, 将本研究的改进策略去除后, Scramble-BWO、Gauss-BWO、Lens-BWO 的 I_{MAE} 较于 BWO 均有不同程度的增大, 说明所采用的改进策略对算法都起积极作用。其中, 引入基于权重的抢食白鲸对算法的影响最大, 其次是自适应高斯扰动、凸透镜成像反向学习策略。

表 3 改进策略的 I_{MAE} 结果统计

Table 3 Statistics of I_{MAE} results for improved strategies

算法	I_{MAE}	相较于 EBWO 的变化
EBWO	3.57×10^{-31}	0
Scramble-BWO	1.34×10^{-28}	1.33×10^{-28}
Gauss-BWO	6.63×10^{-14}	6.62×10^{-14}
Lens-BWO	1.28×10^{-13}	1.27×10^{-13}

本研究 3 项改进策略的部分基准测试函数收敛曲线如图 4 所示。由图 4 可知: 在引入基于权重的抢食白鲸后, 有效的探索方式和位置更新方式的多样性使得算法在开发阶段的寻优能力增强, 提高了寻优的精度; 自适应高斯扰动在鲸坠阶段将白鲸的位置调整到最优值附近, 极大地提高了算法的收敛

速度;对基于凸透镜成像反向学习策略的收敛曲线,可观察到寻优过程中,曲线持续地发生“拐折”。

这说明在该策略下的算法不断挣脱局部极值,有效避免了陷入局部最优的风险。

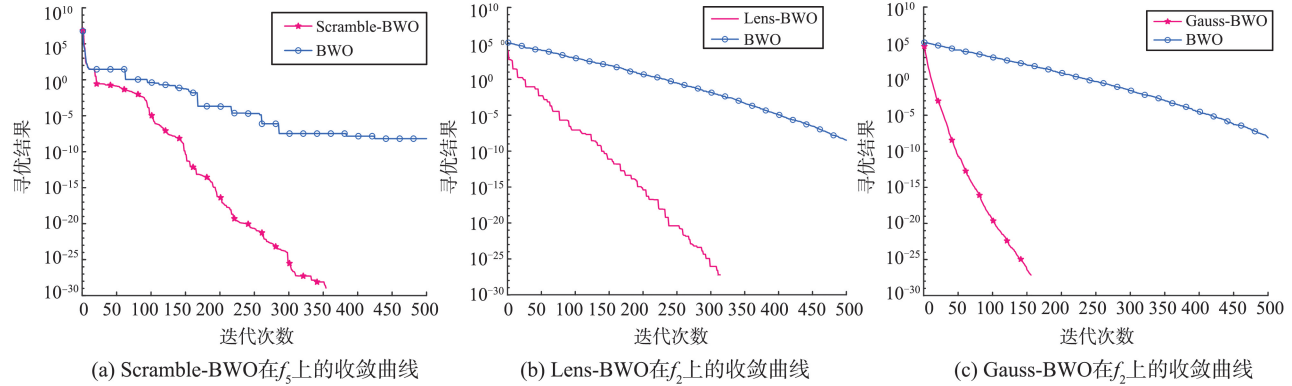


图4 改进策略在部分函数上的收敛曲线

Fig.4 Convergence curves of the improved strategy on partial benchmark function

3.4 算法性能对比分析

将 DBO^[7]、标准 BWO^[10]、基于 Fuch 映射的改进白鲸优化算法 (CPOEBWO)^[16]、WOA^[21]、PSO^[22] 及 EBWO 算法做基准测试函数寻优对比。分别通过 30 次独立试验后,各算法的寻优结果如表 4 所示,每一个基准测试函数上的算法最优结果用粗字体显示。

由表 4 可知:EBWO 在 10 组测试函数中的寻优能力和收敛精度明显优于其他 5 种对比算法。对于单峰函数,EBWO 在求解 $f_1 \sim f_4$ 时均能搜索到理论最优值;在 f_5 上,EBWO 的求解精度更高,且相比于其他算法,所提出算法的平均值和标准差指标结果更小,说明 EBWO 在单峰函数上表现出较强的优势和求解稳定性;在 f_6 上,所提算法寻优的结果收敛到理论最优值。由表 4 的 3 项指标所得算法的综合排名如图 5 所示,EBWO 的综合寻优能力最佳。各算法在部分基准测试函数上的收敛过程如图 6 所示(EBWO 在 f_4 上的收敛过程与 f_2 接近,因此省略 f_4 收敛曲线图),观察到 EBWO 的收敛速度相比于其他算法优势显著,由于自适应高斯扰动将种群逐渐调整到最优值附近,凸透镜成像反向学习策略避免算法在寻优过程种陷入局部最优,因此大大提高了寻优精度的收敛速度。

多峰函数具有多个局部极值,可进一步评估算法的探索能力和挣脱局部极值的能力。由表 4 可知,EBWO 与 BWO、CPOEBWO 在 f_7 、 f_8 上具有相同的收敛精度,但观察图 6 的寻优过程,EBWO 的收敛速度要优于这 2 个算法,说明所采用的改进策略在帮助算法跳出局部极值上具有正向作用。在 f_9 、 f_{10} 上,相比于其他算法,由于引入基于权重的抢食白鲸增大了 EBWO 在开发时的寻优能力,因此表现出更高的收敛精度,同时具有一定的稳定性。

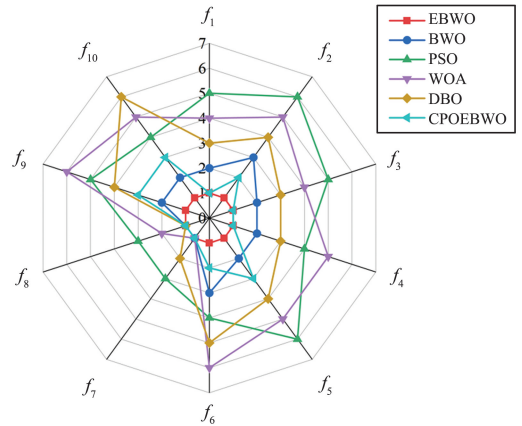


图5 对比算法在基准函数上的排名分析
Fig.5 Comprehensive analysis of rankings against algorithms

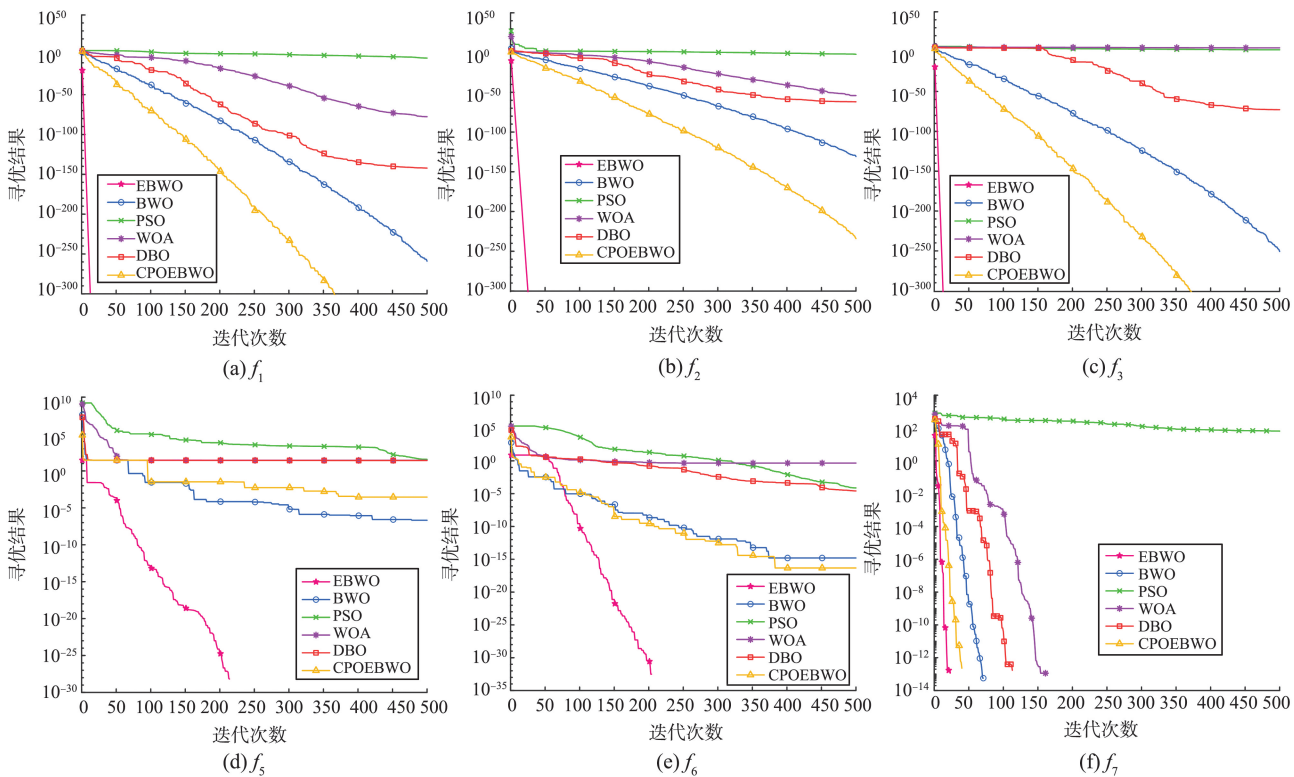
表4 不同算法在基准测试函数上的结果统计

Table 4 Statistics of results of different algorithms on benchmarked functions

函数	算法	最优值	平均值	标准差	函数	算法	最优值	平均值	标准差
f_1	EBWO	0	0	0	f_2	EBWO	0	0	0
	BWO	8.05×10^{-271}	4.01×10^{-259}	0		BWO	1.27×10^{-136}	1.74×10^{-131}	4.67×10^{-131}
	PSO	2.47×10^{-5}	3.40×10^4	3.74×10^4		PSO	3.92×10^{-3}	5.19×10^{-2}	7.51×10^{-2}
	WOA	8.62×10^{-88}	2.20×10^{-68}	1.20×10^{-67}		WOA	1.55×10^{-54}	2.25×10^{-48}	1.08×10^{-47}
	DBO	2.09×10^{-166}	1.94×10^{-110}	1.06×10^{-109}		DBO	6.26×10^{-81}	5.89×10^{-58}	3.23×10^{-57}
	CPOEBWO	0	0	0		CPOEBWO	9.64×10^{-239}	6.00×10^{-229}	0

表4(续)

函数	算法	最优值	平均值	标准差	函数	算法	最优值	平均值	标准差
f_3	EBWO	0	0	0	f_4	EBWO	0	0	0
	BWO	4.19×10^{-280}	9.29×10^{-261}	0		BWO	7.02×10^{-255}	1.29×10^{-243}	0
	PSO	1.12×10^{-10}	2.42×10^{-6}	7.64×10^{-6}		PSO	7.27×10^1	1.92×10^2	5.91×10^1
	WOA	2.18×10^{-131}	1.90×10^{-107}	1.02×10^{-106}		WOA	1.23×10^3	7.88×10^4	4.29×10^4
	DBO	8.46×10^{-205}	1.70×10^{-117}	9.17×10^{-117}		DBO	3.46×10^{-157}	8.75×10^{-71}	4.79×10^{-70}
	CPOEBWO	0	0	0		CPOEBWO	0	0	0
f_5	EBWO	0	2.16×10^{-29}	4.40×10^{-28}	f_6	EBWO	0	0	0
	BWO	5.21×10^{-10}	6.99×10^{-7}	1.07×10^{-6}		BWO	1.69×10^{-16}	3.42×10^{-14}	6.43×10^{-14}
	PSO	2.52×10^1	9.27×10^1	1.39×10^2		PSO	3.28×10^{-5}	3.22×10^{-4}	3.51×10^{-4}
	WOA	2.70×10^1	2.79×10^1	4.63×10^{-1}		WOA	1.21×10^{-1}	3.98×10^{-1}	2.45×10^{-1}
	DBO	2.53×10^1	2.57×10^1	2.11×10^{-1}		DBO	7.05×10^{-6}	8.65×10^{-3}	4.53×10^{-2}
	CPOEBWO	3.23×10^{-7}	4.92×10^{-3}	2.45×10^{-2}		CPOEBWO	1.41×10^{-16}	2.39×10^{-14}	3.32×10^{-14}
f_7	EBWO	0	0	0	f_8	EBWO	8.88×10^{-16}	8.88×10^{-16}	0
	BWO	0	0	0		BWO	8.88×10^{-16}	8.88×10^{-16}	0
	PSO	3.21×10^1	6.06×10^1	1.38×10^1		PSO	1.98×10^1	1.98×10^1	2.44×10^{-2}
	WOA	0	0	0		WOA	8.88×10^{-16}	5.74×10^{-15}	2.18×10^{-15}
	DBO	0	4.64×10^{-1}	1.84		DBO	8.88×10^{-16}	8.88×10^{-16}	0
	CPOEBWO	0	0	0		CPOEBWO	8.88×10^{-16}	8.88×10^{-16}	0
f_9	EBWO	1.57×10^{-32}	1.57×10^{-32}	5.57×10^{-48}	f_{10}	EBWO	1.34×10^{-32}	1.34×10^{-32}	5.56×10^{-48}
	BWO	2.32×10^{-16}	7.02×10^{-14}	1.00×10^{-13}		BWO	2.41×10^{-16}	1.79×10^{-13}	3.87×10^{-13}
	PSO	1.66×10^{-7}	1.73×10^{-2}	3.93×10^{-2}		PSO	6.85×10^{-6}	5.25×10^{-3}	5.70×10^{-3}
	WOA	3.22×10^{-3}	2.66×10^{-1}	1.33		WOA	1.55×10^{-1}	5.78×10^{-1}	2.93×10^{-1}
	DBO	8.32×10^{-8}	3.96×10^{-4}	1.49×10^{-3}		DBO	9.75×10^{-2}	7.19×10^{-1}	4.74×10^{-1}
	CPOEBWO	6.15×10^{-17}	8.21×10^{-14}	1.36×10^{-13}		CPOEBWO	4.67×10^{-15}	4.69×10^{-13}	4.69×10^{-13}



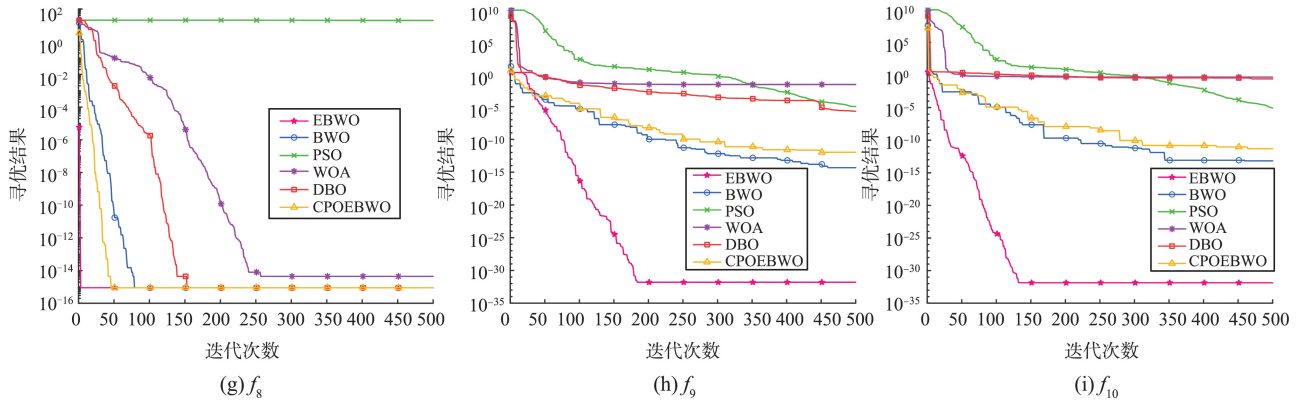


图6 不同算法在部分基准测试函数上的收敛曲线
Fig.6 Convergence curves of different algorithms on some benchmark test functions

3.5 CEC2020 测试函数试验分析

为更全面的评估 EBWO 的算法有效性和稳定性,本节将算法在 CEC2020 函数集上进行优化求解。CEC2020 函数集由挑战难度较大的 10 个优化问题组成,包含 1 个组合的单峰函数 f_{11} 、3 个经过旋转和位移的多峰函数 $f_{12} \sim f_{14}$ 、3 个混合函数 $f_{15} \sim f_{17}$ 、3 个复合函数 $f_{18} \sim f_{20}$ 。函数集的信息如表 5 所示。设种群规模为 50,总迭代次数为 500。各算法均独立运行 30 次。

表 5 CEC2020 测试集函数信息

Table 5 CEC2020 test set function information

函数	维度	搜索范围	最优值
f_{11}	10	$[-100, 100]$	100
f_{12}	10	$[-100, 100]$	1 100
f_{13}	10	$[-100, 100]$	700
f_{14}	10	$[-100, 100]$	1 900
f_{15}	10	$[-100, 100]$	1 700
f_{16}	10	$[-100, 100]$	1 600
f_{17}	10	$[-100, 100]$	2 100
f_{18}	10	$[-100, 100]$	2 200
f_{19}	10	$[-100, 100]$	2 400

EBWO 及对比算法在 CEC2020 测试集的结果如表 6 所示。每一个测试函数上的算法最优结果用

粗字体显示。可知:EBWO 在大部分函数上收敛的平均精度均优于其他 5 种对比算法;在经过旋转和位移的 $f_{12} \sim f_{13}$ 函数上,EBWO 寻优的结果排名第一, f_{14} 中,EBWO 略低于 PSO,但其收敛到的平均结果同样几乎接近最优解;各算法在 CEC2020 测试集函数的收敛平均值的排名分析如图 7 所示。由图 7 可知,EBWO 在 7 个函数上均排名第一,进一步说明本研究提出的算法鲁棒性较强和具有解决复杂问题的能力。

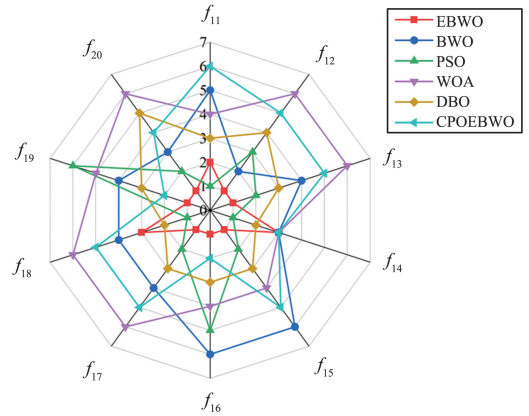


图7 对比算法在 CEC2020 上的排名分析
Fig.7 Ranking analysis of comparison algorithms at CEC2020

表 6 不同算法在 CEC2020 上的结果统计

Table 6 Statistical results of different algorithms at CEC2020

函数	算法	平均值	标准差	函数	算法	平均值	标准差
f_{11}	EBWO	1.11×10^6	5.91×10^6	f_{12}	EBWO	1.43×10^3	1.90×10^2
	BWO	7.75×10^7	4.15×10^7		BWO	1.79×10^3	1.31×10^2
	PSO	1.74×10^3	2.17×10^3		PSO	1.89×10^3	2.46×10^2
	WOA	1.28×10^7	1.65×10^7		WOA	2.23×10^3	3.35×10^2
	DBO	3.01×10^6	1.29×10^7		DBO	1.90×10^3	3.58×10^2
	CPOEBWO	9.24×10^7	4.50×10^7		CPOEBWO	1.95×10^3	1.33×10^2

表6(续)

函数	算法	平均值	标准差	函数	算法	平均值	标准差
f_{13}	EBWO	7.27×10^2	1.08×10^1	f_{14}	EBWO	1.91×10^3	5.39
	BWO	7.46×10^2	6.21		BWO	1.91×10^3	3.72
	PSO	7.32×10^2	2.07×10^1		PSO	1.90×10^3	5.80×10^{-1}
	WOA	7.79×10^2	2.43×10^1		WOA	1.91×10^3	3.71
	DBO	7.35×10^2	1.03×10^1		DBO	1.90×10^3	2.38
	CPOEBWO	7.53×10^2	7.70		CPOEBWO	1.91×10^3	2.23
f_{15}	EBWO	3.94×10^3	2.41×10^3	f_{16}	EBWO	1.60×10^3	2.60×10^{-1}
	BWO	2.70×10^5	1.29×10^5		BWO	1.62×10^3	4.09
	PSO	5.06×10^3	4.21×10^3		PSO	1.61×10^3	8.44
	WOA	1.77×10^5	2.52×10^5		WOA	1.61×10^3	8.20
	DBO	1.15×10^4	1.67×10^4		DBO	1.60×10^3	4.16
	CPOEBWO	2.53×10^5	1.32×10^5		CPOEBWO	1.60×10^3	7.47×10^{-1}
f_{17}	EBWO	2.69×10^3	1.71×10^3	f_{18}	EBWO	2.31×10^3	6.69
	BWO	2.61×10^4	2.48×10^4		BWO	2.32×10^3	1.88×10^2
	PSO	5.17×10^3	1.80×10^3		PSO	2.30×10^3	1.74×10^2
	WOA	1.65×10^5	1.75×10^5		WOA	2.35×10^3	1.46×10^2
	DBO	7.69×10^3	9.96×10^3		DBO	2.30×10^3	1.08×10^2
	CPOEBWO	4.35×10^4	3.17×10^4		CPOEBWO	2.32×10^3	1.99×10^2
f_{19}	EBWO	2.68×10^3	1.06×10^1	f_{20}	EBWO	2.91×10^3	8.72×10^1
	BWO	2.72×10^3	1.12×10^2		BWO	2.94×10^3	1.59×10^1
	PSO	2.79×10^3	8.32×10^1		PSO	2.93×10^3	2.18×10^1
	WOA	2.77×10^3	4.92×10^1		WOA	2.95×10^3	3.13×10^1
	DBO	2.69×10^3	1.10×10^2		DBO	2.94×10^3	2.88×10^1
	CPOEBWO	2.68×10^3	8.18×10^2		CPOEBWO	2.94×10^3	1.80×10^1

3.6 Wilcoxon 秩和检验

为深入评估 EBWO 的性能,本节将其与对比算法在表 1 基准测试函数集上进行 Wilcoxon 符号秩检验。当 $P < 0.05$ 时,表明 2 种算法存在显著性差异;当 $P > 0.05$ 时,2 种算法不存在显著性差异;若 P 为 NaN,则表明 2 种算法寻优性能相当。

统计结果如表 7 所示,S 为显著性判断结果,

“+/-/=”分别表示“显著性差异/不显著性差异/相当”。由表 7 可知:EBWO 与对比算法在 f_7 、 f_8 函数(与 PSO 有显著性差异)上的寻优能力相当,在其余测试函数上与其他算法具有显著性差异。

综上分析,本研究提出的增强型白鲸优化算法具有一定的优越寻优能力。

表 7 Wilcoxon 符号秩检验结果
Table 7 Results of Wilcoxon signed rank test

函数	BWO		PSO		WOA		DBO		CPOEBWO	
	P	S	P	S	P	S	P	S	P	S
f_1	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+
f_2	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+
f_3	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+
f_4	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+
f_5	9.25×10^{-12}	+	7.85×10^{-12}	+	1.09×10^{-11}	+	1.39×10^{-11}	+	9.35×10^{-2}	+
f_6	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+
f_7	NaN	=	1.21×10^{-12}	+	NaN	=	NaN	=	NaN	=
f_8	NaN	=	1.21×10^{-12}	+	NaN	=	NaN	=	NaN	=
f_9	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+
f_{10}	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+	1.21×10^{-12}	+

4 工程应用

4.1 减速器设计问题

减速器是齿轮箱的重要部件之一,减速器质量最小化可以使齿轮箱工作速度更高效。减速器的设计模型如图8所示,齿轮设计问题共7个变量:齿面宽度 x_1 , 齿轮齿数 x_2, x_3 , 轴承间的第一根轴长度 x_4 , 轴承间的第二根轴长度 x_5 , 第一根轴的直径 x_6 , 第二根轴的直径 x_7 。

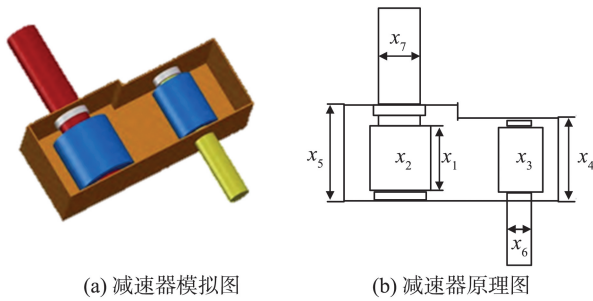


图8 减速器设计问题模型

Fig.8 Model of speed reducer design problem

减速器设计问题的数学模型如下。

目标函数为

$$f(x) = 0.785 4x_1x_2(3.333 3x_3^2 + 14.933 4x_3 - 43.093 4) - 1.508x_1(x_6^2 + x_7^2) + 0.785 4(x_4x_6^2 + x_5x_7^2) \quad (14)$$

约束条件为

$$g_1(x) = \frac{27}{x_1x_2x_3} - 1 \leq 0,$$

$$g_2(x) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \leq 0,$$

$$g_3(x) = \frac{1.93x_4^3}{x_2x_6^4x_3} - 1 \leq 0,$$

$$g_4(x) = \frac{1.93x_5^3}{x_2x_7^4x_3} - 1 \leq 0,$$

$$g_5(x) = \frac{\sqrt{[745x_4/(x_2x_3)]^2 + 16.9 \times 10^6}}{110x_6^3} - 1 \leq 0,$$

$$g_6(x) = \frac{\sqrt{[745x_5/(x_2x_3)]^2 + 157.5 \times 10^6}}{85x_7^3} - 1 \leq 0,$$

$$g_7(x) = \frac{x_2x_3}{40} - 1 \leq 0,$$

$$g_8(x) = \frac{5x_2}{x_1} - 1 \leq 0,$$

$$g_9(x) = \frac{x_1}{12x_2} - 1 \leq 0,$$

$$g_{10}(x) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0,$$

$$g_{11}(x) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0,$$

式中:变量边界约束为 $2.6 \leq x_1 \leq 3.6$, $0.7 \leq x_2 \leq 0.8$, $17 \leq x_3 \leq 28$, $7.3 \leq x_4 \leq 8.3$, $7.3 \leq x_5 \leq 8.3$, $2.9 \leq x_6 \leq 3.9$, $5 \leq x_7 \leq 5.5$ 。

EBWO 与对比算法在该工程问题分别迭代寻优 30 次的结果如表 8 所示。由表 8 可知,本研究提出的 EBWO 算法取得减速器的最小质量为 2 444.749 8,与其他算法对比,本研究改进的算法优化效果显著。

表8 减速器设计问题结果统计

Table 8 Statistics on the results of speed reducer design problem

算法	x_1	x_2	x_3	x_4	x_5	x_6	x_7	最优质量
EBWO	4.061 0	0.684 0	14.461 8	8.370 2	8.131 9	3.356 6	5.286 8	2 444.74
BWO	3.600 0	0.700 0	17.000 0	7.300 0	7.800 0	3.379 3	5.295 5	2 877.73
PSO	2.600 0	0.739 7	17.000 0	7.300 0	7.800 0	3.349 8	5.286 4	2 844.88
WOA	2.806 7	0.700 1	17.001 9	7.300 8	7.800 9	3.392 0	5.286 5	2 856.93
DBO	2.802 2	0.700 0	18.354 5	7.951 0	7.828 4	3.350 7	5.286 3	2 881.40
CPOEBWO	2.600 0	0.800 0	17.000 0	7.300 0	7.800 0	3.394 1	5.287 8	2 905.05

4.2 压力容器设计问题

压力容器的设计目标是在满足生产需要的同时使总设计费用最少。压力容器设计问题的模型如图9所示,该模型共有4个设计变量:容器内半径 z_1 、容器长度(不含封头) z_2 、容器厚度 z_3 、封头厚度 z_4 。

压力容器的数学设计模型如下。

目标函数为

$$f(z) = 0.622 4z_1z_3z_4 + 1.778 1z_2z_3^2 + 3.166 1z_1^2z_4 + 19.84z_1^2z_3 \quad (15)$$

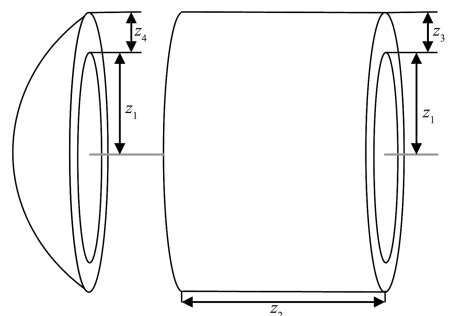


图9 压力容器设计问题模型

Fig.9 Model of pressure vessel design problem

约束条件为

$$g_1(z) = -z_1 + 0.019 3z_3 \leq 0,$$

$$g_2(z) = -z_2 + 0.0095 4z_3 \leq 0,$$

$$g_3(z) = -\pi z_3^2 z_4 - \frac{4}{3} \pi z_3^2 + 1 296 000 \leq 0,$$

$$g_4(z) = z_4 - 240 \leq 0,$$

式中:变量的边界约束为 $0 \leq z_1 \leq 99, 0 \leq z_2 \leq 99, 10 \leq z_3 \leq 200, 10 \leq z_4 \leq 200$ 。EBWO 与对比算法在该工程问题分别迭代寻优 30 次的结果如表 9 所示。可知,经 EBWO 优化后的最低费用为 5 837.63 元,明显优于其他 5 种对比算法。

表 9 压力容器设计问题结果统计

Table 9 Statistics on the results of pressure vessel design problem

算法	z_1	z_2	z_3	z_4	最优费用
EBWO	1.107 6	0.547 5	57.388 6	48.739 5	5 837.63
BWO	1.162 6	0.420 1	42.211 3	175.356 1	6 879.73
PSO	0.893 1	0.441 5	46.277 0	130.927 7	5 947.70
WOA	1.256 6	0.631 7	63.172 5	19.141 0	7 192.20
DBO	0.778 2	0.384 6	40.319 6	200.000 0	5 885.33
CPOEBWO	1.186 1	1.220 9	54.839 8	95.750 0	7 328.98

通过减速器和压力容器 2 个工程应用设计问题的试验证明了本研究提出的增强型白鲸优化算法的有效性以及具有优异的寻优能力。

5 结束语

为了提高白鲸优化算法在优化问题的寻优能力,本研究提出增强型白鲸优化算法(EBWO)。引入了抢食型的白鲸个体,并根据白鲸与最优、最差位置的距离赋予不同的权重,在提高算法开发能力的同时,丰富 BWO 在该阶段的位置信息;使用自适应的高斯扰动在最优值附近调整进入鲸坠阶段的白鲸,提高算法的收敛速度;引入凸透镜成像反向学习策略,利用成像原理提供反向位置,改善算法易陷入局部最优值的问题。试验仿真结果证明,本研究提出的 EBWO 在较快的收敛速度下可得到更优精度的问题结果。下一步考虑将 EBWO 算法应用到现实问题中,进一步检验算法的实用性。

参考文献:

- [1] 杜晓昕,周薇,王浩,等. 智能算法的亚群优化策略综述[J]. 计算机应用, 2024, 44(3): 819-830.
DU Xiaoxin, ZHOU Wei, WANG Hao, et al. Survey of subgroup optimization strategies for intelligent algorithms [J]. Journal of Computer Applications, 2024, 44(3): 819-830.
- [2] 田野,陈津津,张兴义. 面向约束多目标优化的进化计算与梯度下降联合优化方法[J]. 计算机应用, 2024, 44(5): 1386-1392.
TIAN Ye, CHEN Jinjin, ZHANG Xingyi. Hybrid optimizer combining evolutionary computation and

gradient descent for constrained multi-objective optimization [J]. Journal of Computer Applications, 2024, 44(5): 1386-1392.

- [3] 任洁,彭建文. 求解多目标优化问题的邻近牛顿法[J]. 应用数学学报, 2022, 45(2): 222-237.
REN Jie, PENG Jianwen. Proximal Newton methods for multiobjective optimization problems [J]. Acta Mathematicae Applicatae Sinica, 2022, 45(2): 222-237.
- [4] 张林,沈佳颖,胡传陆,等. 基于信噪比的学习型哈里斯鹰算法[J/OL]. 北京航空航天大学学报. 2024. <https://doi.org/10.13700/j.bh.1001-5965.2023.0433>
ZHANG Lin, SHEN Jiaying, HU Chuanlu, et al. Learning Harris hawks optimization algorithm with signal-to-noise ratio [J/OL]. Journal of Beijing University of Aeronautics and Astronautics. 2024. <https://doi.org/10.13700/j.bh.1001-5965.2023.0433>
- [5] BRAIK M S. Chameleon swarm algorithm: a bio-inspired optimizer for solving engineering design problems [J]. Expert Systems with Applications, 2021, 174: 114685.
- [6] BENYAMIN A, SOLEIMANIAN G F, SEYEDALI M. African vultures optimization algorithm: a new nature-inspired metaheuristic algorithm for global optimization problems [J]. Computers & Industrial Engineering, 2021, 158: 107408.
- [7] XUE J K, SHEN B. Dung beetle optimizer: a new metaheuristic algorithm for global optimization [J]. The Journal of Supercomputing, 2023, 79(7): 7305-7336.
- [8] SEYYEDABBASI A, KIANI F. Sand cat swarm optimization: a nature-inspired algorithm to solve global optimization problems [J]. Engineering with Computers, 2023, 39(4): 2627-2651.
- [9] 刘庆鑫,齐琦,贾鹤鸣,等. 混合改进策略的阿奎拉鹰优化算法[J]. 山东大学学报(工学版), 2023, 53(4):

- 93-103.
- LIU Qingxin, QI Qi, JIA Heming, et al. Aquila optimizer based on hybrid improved strategies[J]. Journal of Shandong University (Engineering Science), 2023, 53(4): 93-103.
- [10] ZHONG C, LI G, MENG Z. Beluga whale optimization: a novel nature-inspired metaheuristic algorithm [J]. Knowledge-Based Systems, 2022, 251: 109215.
- [11] 梁天添, 王英东, 杨健雄, 等. 测量丢失下时滞惯性系统故障估计的优化鲁棒 Kalman 滤波[J]. 中国惯性技术学报, 2023, 31(6): 627-636.
- LIANG Tiantian, WANG Yingdong, YANG Jianxiong, et al. Optimized robust Kalman filtering for fault estimation of time-delay inertial systems with missing measurement [J]. Journal of Chinese Inertial Technology, 2023, 31(6): 627-636.
- [12] 汪业正午, 季亮, 常潇, 等. 故障下新能源场站间主动电压支撑协调控制策略[J]. 太阳能学报, 2023, 44(12): 560-567.
- WANG Yezhengwu, JI Liang, CHANG Xiao, et al. Active voltage support coordination control strategy between new energy field stations under failure[J]. Acta Energetica Solaris Sinica, 2023, 44(12): 560-567.
- [13] 孔云, 周学良, 冷杰武. 基于改进白鲸优化算法的低碳柔性工艺规划[J]. 现代制造工程, 2024(1): 80-88.
- KONG Yun, ZHOU Xueliang, LENG Jiewu. Low-carbon flexible process planning based on improved beluga whale optimization algorithm [J]. Modern Manufacturing Engineering, 2024(1): 80-88.
- [14] 陈元健, 黄靖, 孙晓, 等. 基于 BWO 优化 VMD 联合小波阈值的管道泄漏次声波去噪方法[J]. 机电工程技术, 2024, 53(3): 54-59.
- CHEN Yuanjian, HUANG Jing, SUN Xiao, et al. Research on infrasound denoising method for pipeline leakage based on BWO optimized VMD joint wavelet thresholding [J]. Mechanical & Electrical Engineering Technology, 2024, 53(3): 54-59.
- [15] JIA H M, WEN Q X, WU D, et al. Modified beluga whale optimization with multi-strategies for solving engineering problems [J]. Journal of Computational Design and Engineering, 2023, 10(6): 2065-2093.
- [16] 陈心怡, 张孟健, 王德光. 基于 Fuch 映射的改进白鲸优化算法及应用[J]. 计算机工程与科学, 2024, 46(8): 1482-1492.
- CHEN Xinyi, ZHANG Mengjian, WANG Deguang. Improved beluga whale optimization algorithms based on Fuch mapping and applications [J]. Computer Engineering & Science, 2024, 46(8): 1482-1492.
- [17] 王亚辉, 张虎晨, 王学兵, 等. 基于混沌反向学习和水波算法改进的白鲸优化算法[J]. 计算机应用研究, 2024, 41(3): 729-735.
- WANG Yahui, ZHANG Huchen, WANG Xuebing, et al. Improved beluga whale optimization algorithm based on chaotic inverse learning and water wave algorithm[J]. Application Research of Computers, 2024, 41(3): 729-735.
- [18] 陈曦明, 张军伟. 融合 FDB 策略和切线飞行的改进白鲸优化算法[J]. 计算机时代, 2023(11): 46-51.
- CHEN Ximing, ZHANG Junwei. Improved BWO algorithm combining FDB strategy and tangent flight[J]. Computer Era, 2023(11): 46-51.
- [19] 潘勃成, 李少波, 周鹏, 等. 改进正弦算法引导的蜣螂优化算法[J]. 计算机工程与应用, 2023, 59(22): 92-110.
- PAN Jincheng, LI Shaobo, ZHOU Peng, et al. Dung beetle optimization algorithm guided by improved sine algorithm[J]. Computer Engineering and Applications, 2023, 59(22): 92-110.
- [20] 肖怡心, 刘三阳. 融合模式搜索的蝗虫优化算法及其应用[J]. 西安电子科技大学学报, 2024, 51(2): 137-156.
- XIAO Yixin, LIU Sanyang. Integration of pattern search into the grasshopper optimization algorithm and its applications[J]. Journal of Xidian University, 2024, 51(2): 137-156.
- [21] MIRJALILI S, LEWIS A. The whale optimization algorithm[J]. Advances in Engineering Software, 2016, 95: 51-67.
- [22] KENNEDY J, EBERHART R. Particle swarm optimization [C]// Perth, Australia: IEEE, 1995, 4: 1942-1948.
- [23] 向君幸, 吴永红. 基于邻域重心反向学习的混合樽海鞘群蝴蝶优化算法[J]. 计算机应用, 2023, 43(3): 820-826.
- XIANG Junxing, WU Yonghong. Hybrid salp swarm and butterfly optimization algorithm combined with neighborhood centroid opposition-based learning [J]. Journal of Computer Applications, 2023, 43(3): 820-826.

(编辑:熊小原)