

一种基于多策略改进的鲸鱼算法

吕嘉婧¹ 李 磊¹
LYU Jiajing LI Lei

摘要

针对标准鲸鱼算法 (whale optimization algorithm, WOA) 寻优精度低、可能陷入局部最优解等问题, 提出一种基于多策略改进的鲸鱼算法 (MSWOA)。首先使用 Tent 混沌映射与反向学习策略初始化鲸鱼种群; 然后, 引入自适应收敛因子根据适应度值动态调整个体在迭代过程中的包围步长, 平衡算法的探索 and 开发能力; 最后引入随机差分变异策略, 加强算法跳出局部最优的能力。实验结果显示, MSWOA 算法有效地提高了鲸鱼算法在稳定性、求解精度和收敛速度等方面的表现, 展现了更优秀的求解效果。

关键词

鲸鱼算法; 混沌映射; 自适应收敛因子; 随机差分策略

doi: 10.3969/j.issn.1672-9528.2024.02.009

0 引言

鲸鱼优化算法 (whale optimization algorithm, WOA) 是 Mirjalili M 等人^[1]于 2016 年提出的一种群智能优化算法, 算法模拟了鲸鱼个体不断寻找更优解的行为, 并通过合作和信息传递来提高整体的搜索性能。与其他优化算法相比, 鲸鱼算法的实现相对简单, 不需要复杂的数学模型或参数调整^[2]。然而, 传统的鲸鱼优化算法依旧存在一些缺陷, 例如算法初始种群多样性低、全局探索和局部开发能力难以协调、可能会陷入局部最优解而难以跳出等问题。

针对算法初始种群多样性低的问题, 部分学者采用混沌序列、精英反向学习等方式进行改进, 如 Garip^[3]、付接递^[4]、孟宪猛^[5]、王仲雨等人^[6]。部分学者仅对初始化方式进行改进不足以提升算法的全局搜索与局部搜索能力, 尹梅等人^[7]、王梓辰^[8]、Huang M 等人^[9]、Kun Y 等人^[10]均是对收敛因子进行非线性改进以平衡全局搜索与局部搜索能力。此外, 部分学者在上述基础上引入其他策略, 如秋兴国等人^[11]、郑威迪等^[12]引入莱维飞行策略以跳出局部最优, 张达敏等人^[13]提出一种逐维小孔成像反向学习策略, 柴岩等人^[14]引入了黄金正弦策略以扩大种群搜索范围。

以上研究成果虽然对 WOA 进行了一定程度上的改进, 但是在寻优精度以及收敛速度方面仍有提升空间。针对鲸鱼算法中存在的问题, 本文提出了一种融合混沌反向学习策略、自适应收敛因子策略和随机差分策略的多策略改进的鲸鱼优化算法。Rahmat M 等^[15]将鲸鱼算法与模拟退火算法和人工蜂群算法相结合, 以提升收敛性能。

1 基本鲸鱼优化算法

WOA 的搜索过程可划分为三个阶段: 收缩包围猎物、气泡网攻击、随机搜索猎物。

1.1 收缩包围猎物

WOA 假设当前种群中的最优解位于猎物位置或接近目标猎物的区域。其他种群个体则通过当前最优解来更新其位置。这一过程可以用数学模型描述为:

$$D = |C \cdot X^*(t) - X(t)| \quad (1)$$

$$X(t+1) = X^*(t) - A \cdot D \quad (2)$$

式中: t 为当前迭代次数; $X^*(t)$ 为当前最优解, D 为 $X^*(t)$ 与 $X(t)$ 之间的距离, A 、 C 定义为:

$$A = 2a \cdot r - a \quad (3)$$

$$C = 2r \quad (4)$$

式中: r 为区间 $[0,1]$ 的随机数, a 为收敛因子, 定义为:

$$a = 2 - 2t / T_{\max} \quad (5)$$

式中: $T_{\max}$ 为最大迭代次数。

1.2 气泡网攻击阶段

气泡网攻击模拟鲸鱼向猎物收缩包围的过程, WOA 中存在两种位置更新策略: 收缩式包围策略和螺旋式策略, 且两种策略的几率各占一半, 即两者概率约为 50%。

(1) 收缩式包围策略

收缩式包围通过式 (3) 中的收敛因子的线性递减来实现。

(2) 螺旋式策略

计算鲸鱼与猎物间的距离, 并模拟鲸鱼捕食时的螺旋运动方式, 其数学模型可表达为:

$$X(t+1) = D' \cdot e^{bm} \cdot \cos(2\pi m) + X^*(t) \quad (6)$$

$$D' = |X^*(t) - X(t)| \quad (7)$$

式中: m 为 $[-1,1]$ 内的随机数。

1.3 随机搜索猎物阶段

在随机搜索时, 鲸鱼个体向随机选中的鲸鱼个体移动, 数学模型如下:

$$D = |C \cdot X_{rand}(t) - X(t)| \quad (8)$$

$$X(t+1) = X_{rand}(t) - A \cdot D \quad (9)$$

式中: $X_{rand}(t)$ 表示该鲸鱼个体的位置。

2 基于多策略改进的鲸鱼优化算法

针对鲸鱼算法存在的不足与问题, 本文采用多种策略进行改进。首先, 为增强初始种群多样性, 使用混沌反向学习策略初始化鲸鱼种群。然后, 为平衡算法的局部搜索能力与全局开发能力, 本文引入自适应收敛因子根据适应度值动态调整个体在迭代过程中的包围步长。最后, 引入随机差分变异策略, 加强算法跳出局部最优的能力。

2.1 混沌反向学习策略

在基本鲸鱼优化算法中, 初始化阶段的种群质量对算法的性能有着直接的影响。传统的随机初始化方法随机性较大, 多样性较差, 在某些情况下可能会导致种群聚集在搜索空间的某些区域, 无法充分探索整个搜索空间, 导致算法的优异程度受限。因此, 为了使初始种群分布更均匀, 本文将混沌映射和反向学习策略结合, 形成混沌反向学习方法, 通过该方法生成鲸鱼算法的初始种群。

混沌序列采用 Tent 混沌映射的方法, 与 Cubin 映射、Singer 映射等方法相比, Tent 混沌映射具有更好的深度搜索能力^[16], 其数学模型为:

$$x_{i+1} = \begin{cases} \delta x_i & 0 < x_i < 0.5 \\ \delta(1 - x_i) & 0.5 < x_i < 1 \end{cases} \quad (10)$$

反向学习是一种扰动手段, 其主要原理是通过对目标空间中已知解的反向学习, 得到该解在空间中的反向解, 并选择更优的解作为当前解。其数学模型可以描述如下:

$$x' = r(x_{\max} - x_{\min}) - x \quad (11)$$

式中: r 为随机数, $x_{\max}$ 、 $x_{\min}$ 为初始解中的最大值与最小值。

采用混沌反向学习方法生成初始种群的步骤如下。

步骤 1: 通过 Tent 映射生成 n 个初始解 x_i 。

步骤 2: 根据式 (12) 获取各个初始解的反向解 x'_i 。

步骤 3: 将 x_i 和 x'_i 进行合并处理后, 选取适应度值排在前二分之一的 n 个个体作为新种群 N 。

2.2 自适应收敛因子策略

在鲸鱼算法中, 收敛因子 a 从 2 线性递减到 0, 这种方式没有很好地平衡算法的全局寻优和局部开发能力。此外, 对适应度不同的鲸鱼个体, 其包围步长的计算方法是一致的, 没有反映出鲸鱼个体间的差异。因此, 本文提出一种自适应收敛因子, 计算公式为:

$$a = 2 - (2 - 2 \cdot \frac{rank_i}{N}) \cdot \sin(\frac{t}{2 \cdot T_{\max}} \pi) \quad (12)$$

式中: t 为当前迭代次数, $T_{\max}$ 为最大迭代次数, $rank_i$ 为个体的适应度值在种群中的排名。

2.3 随机差分策略

差分算法 (differential evolution algorithm) 是一种基于演化算法的随机优化模型, 模拟生物进化的过程来求解连续变量的优化问题, 通过不断迭代, 利用变异和交叉操作来产生新的解, 并根据适应度评价来筛选和保留适应环境的个体。

为避免鲸鱼算法陷入局部最优, 本文受差分算法变异策略的启发, 对当前个体、最优个体、随机选择的鲸鱼个体进行随机差分产生新的个体, 其数学模型如下:

$$X(t+1) = r_1(X'(t) - X(t)) - r_2(X^*(t) - X(t)) \quad (13)$$

式中: r_1 、 r_2 为 $[0,1]$ 区间内的随机数, $X'(t)$ 为当前种群中的最优个体, $X^*(t)$ 为随机选中的鲸鱼个体。

当个体更新位置后, 使用随机差分策略再次更新个体位置, 并保留最优位置。这种策略能够有效地避免种群陷入局部最优解, 使得算法具有更强的全局搜索能力和寻优效果。

综上所述, 本文所提出的基于多策略改进的鲸鱼算法的具体步骤如下。

步骤 1: 设置初始参数。

步骤 2: 利用混沌反向学习策略生成初始鲸鱼种群。

步骤 3: 评估每个鲸鱼个体的适应度, 并标记最佳个体位置。

步骤 4: 计算收敛因子 a , 并更新参数 A 、 C 的值。

步骤 5: 产生 $[0,1]$ 内的随机数 p , 当 $p < 0.5$, $|A| < 1$ 时, 根据式 (2) 进行收缩包围; 当 $p \geq 0.5$ 时, 根据式 (6) 进行螺旋更新位置; 当 $p < 0.5$, $|A| \geq 1$ 时, 根据公式 (9) 进行随机搜索。

步骤 6: 根据式 (13) 对鲸鱼个体进行随机差分再次更新位置。

步骤 7: 先确定算法的迭代数是否已到达最大值, 如果达到最大值, 则跳出循环并终止运算, 输出最佳结果, 否则回到步骤 3。

3 实验结果与分析

3.1 实验环境和测试函数

本文的仿真实验环境为: 操作系统是 Windows11, CPU 为 Inter(R) Core(TM) i5-12 500 h, 主频率为 2.5 Hz, 内存为 16 GB。各算法的种群数均为 30, 最大迭代次数 $T_{\max}$ 为 1000, 各算法单独执行 30 次, 使用 10 个测试函数进行 Python 实验对比, 其中包括 4 个单峰函数、4 个多峰函数与 2 个混合函数。测试函数如表 1 所示。

表 1 测试函数

函数	维度	范围
$f_1(x) = \sum_{i=1}^n x_i^2$	30	[-100,100]
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10,10]
$f_3(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	30	[-100,100]
$f_4(x) = \sum_{i=1}^n ix_i^4 + random[0,1)$	30	[-1.28,1.28]
$f_5(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5,5]
$f_6(x) = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(-0.2 \sqrt{\frac{1}{n} \cos(2\pi x_i)}) + 20 + e$	30	[-32,32]
$f_7(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30	[-600,600]
$f_8(x) = 0.1 \{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] \} + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] + \sum_{i=1}^n u(x_i, 5, 100, 4)$	30	[-50,50]
$f_9(x) = (\frac{1}{500} + \sum_{i=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6})^{-1}$	2	[-65,65]
$f_{10}(x) = \sum_{i=1}^n [a_i - \frac{x_i(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	4	[-5,5]

3.2 实验结果分析

在不同基准测试函数上比较 MSWOA 与其他优化算法，如标准鲸鱼算法（WOA）、粒子群算法（PSO）、嵌入 Circle 映射和逐维小孔成像反向学习的鲸鱼优化算法（MWOA）、基于自适应权重和模拟退火的鲸鱼优化算法^[17]（W-SA-WOA）进行对比，对比结果见表 2，最优结果已用粗体标出。

可以看出，MSWOA 算法在 10 个基准函数中寻优效果最佳。在单峰函数 f_1 、 f_2 、 f_3 上 MSWOA 算法均达到了理论最优值，并且标准差也为 0，在单峰函数 f_4 上虽未达到最优值，但相比于其他算法具有一定的优越性和稳定性。在多峰函数 f_5 、 f_6 与 f_7 上获得更理想的平均精度与稳定性， f_8 的平均值差于 PSO 算法但优于其他算法，说明 MSWOA 算法具有较强的全局探索能力以及较好的稳定性。此外，在混合函数 f_9 上，MSWOA 的平均值差于 W-SA-WOA 算法与 PSO 算法，但优于其余两种算法，标准差差于 W-SA-WOA 算法，但优于其余三种算法。在混合函数 f_{10} 上，MSWOA 算法的平均精度最优，稳定性也仅次于 MWOA 算法，说明 MSWOA 算法具有更加平衡的全局搜索和局部勘测能力。

表 2 算法性能对比

函数	测试指标	MSWOA	WOA	PSO	MWOA	W-SA-WOA
f_1	平均值	0.00E+00	1.71E-42	1.02E+00	0.00E+00	0.00E+00
	标准差	0.00E+00	6.23E-83	2.83E-01	0.00E+00	0.00E+00
f_2	平均值	0.00E+00	1.43E-30	2.02E+01	0.00E+00	2.58E-121
	标准差	0.00E+00	2.20E-59	3.00E+01	0.00E+00	6.68E-121
f_3	平均值	0.00E+00	3.26E-11	7.28E+01	0.00E+00	0.00E+00
	标准差	0.00E+00	8.64E-11	2.41E+01	0.00E+00	0.00E+00
f_4	平均值	3.28E-05	8.84E-05	4.82E-01	3.57E-05	1.17E-04
	标准差	2.97E-05	7.75E-05	2.94E-01	3.66E-05	1.00E-04
f_5	平均值	0.00E+00	0.00E+00	1.18E+02	0.00E+00	0.00E+00
	标准差	0.00E+00	0.00E+00	2.79E+01	0.00E+00	0.00E+00
f_6	平均值	4.43E-16	3.76E-15	2.78E+00	4.44E-16	3.02E-15
	标准差	0.00E+00	1.28E-15	4.58E-01	0.00E+00	1.77E-15
f_7	平均值	0.00E+00	0.00E+00	1.18E-01	0.00E+00	1.50E-03
	标准差	0.00E+00	0.00E+00	3.90E-02	0.00E+00	7.30E-03
f_8	平均值	2.47E+00	1.65E+00	2.80E-01	3.34E-03	4.21E-02
	标准差	1.12E-01	1.87E-01	2.80E-01	4.51E-07	2.89E-02
f_9	平均值	3.32E+00	3.47E+00	2.45E+00	7.85E+00	1.03E+00
	标准差	1.47E+00	3.15E+00	1.64E+00	2.47E+00	1.68E-01
f_{10}	平均值	3.75E-04	3.75E-04	1.48E-03	2.47E+00	4.92E-04
	标准差	1.67E-04	4.69E-04	5.46E-04	1.55E-04	2.08E-04

为了更直观展示 MSWOA 的收敛速度，此处将 MSWOA 算法与 WOA 算法、MWOA 算法、PSO 算法在单峰函数 f_4 与多峰函数 f_7 上进行对比。

图 1 为单峰函数 f_4 的收敛曲线，从中可以看出，MSWOA 算法整体寻优效果最佳，明显优于 WOA 算法、MWOA 算法、PSO 算法，收敛速度与寻优精度均得到了显著提高。此外，MSWOA 算法在收敛过程中出现了明显拐点，说明 MSWOA 算法在引入随机差分策略后，成功跳出了局部最优。

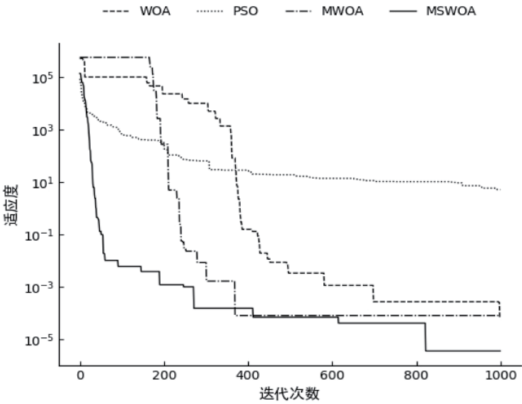


图 1 f_4 函数变化曲线

由图 2 可知在 WOA 算法、MWOA 算法、PSO 算法在

开始时,都陷入了局部最优解,而MSWOA的收敛曲线则近乎垂直,收敛速度明显快于其他算法,最先达到了最优值。因此,与WOA算法、MWOA算法、PSO算法相比较,本文提出的MSWOA改进算法在寻优精度和收敛速度上更胜一筹。

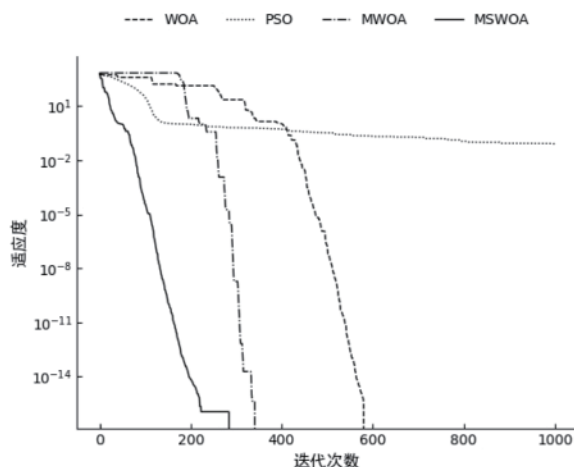


图2 f_7 函数变化曲线

4 结语

本文针对鲸鱼算法存在的问题与不足,提出一种基于多策略改进的鲸鱼算法(MSWOA)。首先使用 Tent 混沌映射与反向学习策略初始化鲸鱼种群;然后,引入自适应收敛因子根据适应度值动态调整个体在迭代过程中的包围步长,平衡算法的探索和开发能力;最后引入随机差分变异策略,加强算法跳出局部最优的能力。通过在10个基准函数上进行对比实验,结果表明,本文所提出的MSWOA算法显著提升了鲸鱼算法在稳定性、求解精度及收敛速度方面的性能,展现出更好的求解效果。

参考文献:

- [1] MIRJALILI S, LEWIS A. The whale optimization algorithm[J]. Advances in engineering software, 2016, 95: 56-17.
- [2] 李雅丽, 王淑琴, 陈倩茹, 等. 若干新型群智能优化算法的对比研究[J]. 计算机工程与应用, 2020, 56(22): 1-12.
- [3] GARIP Z, CIMEN M, KARAYEL D, et al. The Chaos-based Whale Optimization Algorithms Global Optimization[EB/OL]. (2019-11-01)[2023-08-25]. https://www.researchgate.net/publication/337481871_The_Chaos-based_Whale_Optimization_Algorithms_Global_Optimization.
- [4] 付接递, 李振东, 郭辉. 基于教与学和逐维柯西变异的鲸鱼优化算法[J]. 计算机工程与科学, 2023, 45(5): 940-950.
- [5] 孟宪猛, 蔡翠翠. 基于精英反向学习和 Lévy 飞行的鲸鱼优化算法[J]. 电子测量技术, 2021, 44(20): 82-87.
- [6] 王仲雨, 高美凤. 基于改进鲸鱼优化算法的近红外光谱波长变量选择方法及其应用[J]. 分析测试学报, 2023, 42(1): 37-44.
- [7] 尹梅, 柯鹏, 章呈瑞. 一种融合多种策略的改进鲸鱼优化算法[J]. 武汉科技大学学报, 2023, 46(2): 145-152.
- [8] 王梓辰, 窦震海, 董军, 等. 多策略改进的自适应动态鲸鱼优化算法[J]. 计算机工程与设计, 2022, 43(9): 2638-2645.
- [9] HUANG M, ZHAN X, LIANG X. Improvement of Whale Algorithm and Application[C]//2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT). Piscataway: IEEE, 2019: 23-25.
- [10] KUN Y, KAN Y. Improved whale algorithm for economic load dispatch problem in hydropower plants and comprehensive performance evaluation[J]. Water resources management, 2022, 36(15): 5823-5838.
- [11] 秋兴国, 王瑞知, 张卫国, 等. 基于混合策略改进的鲸鱼优化算法[J]. 计算机工程与应用, 2022, 58(1): 70-78.
- [12] 郑威迪, 李志刚, 贾涵中, 等. 基于改进型鲸鱼优化算法和最小二乘支持向量机的炼钢终点预测模型研究[J]. 电子学报, 2019, 47(3): 700-706.
- [13] 张达敏, 徐航, 王依柔, 等. 嵌入 Circle 映射和逐维小孔成像反向学习的鲸鱼优化算法[J]. 控制与决策, 2021, 36(5): 1173-1180.
- [14] 柴岩, 朱玉, 任生. 多策略协同的改进鲸鱼优化算法[J]. 计算机工程与科学, 2023, 45(7): 1308-1319.
- [15] RAHMAT M A, SHAHRAM H E, JAVAD M M. A novel multi-objective optimization algorithm for Pareto design of a fuzzy full state feedback linearization controller applied on a ball and wheel system[J]. Transactions of the institute of measurement and control, 2022, 44(7): 1388-1409.
- [16] KAUR G, ARORA S. Chaotic whale optimization algorithm[J]. Journal of computational design and engineering, 2018, 5(3): 275-284.
- [17] 褚鼎立, 陈红, 王旭光. 基于自适应权重和模拟退火的鲸鱼优化算法[J]. 电子学报, 2019, 47(5): 992-999.

【作者简介】

吕嘉婧(2000—),女,黑龙江双鸭山人,硕士研究生,研究方向:大数据分析、优化算法。

李磊(1973—),女,新疆乌鲁木齐人,教授,博士,研究方向:计量经济、数据挖掘。

(收稿日期:2023-11-02)