

基于混沌精英和 Lévy 飞行策略的鲸鱼优化算法

夏超¹, 欧阳平¹, 李明^{2,3}, 屈盈飞^{2,3*}, 郭玮峰¹

(1. 重庆工商大学 废油资源化技术与装备教育部工程研究中心, 重庆 400067;

2. 重庆工商大学 检测控制集成系统工程实验室, 重庆 400067;

3. 重庆工商大学 人工智能学院, 重庆 400067)

摘要:针对鲸鱼优化算法(Whale Optimization Algorithm, WOA)存在的收敛速度慢、精度低的问题,提出了基于 Tent 混沌精英和 Lévy 飞行策略的鲸鱼优化算法(TELWOA)。使用 Tent 混沌映射初始化鲸鱼种群,保持种群的多样性,并通过引入精英反向学习策略,对初始种群的精英个体生成反向解,选取适应度高的种群作为下一代鲸鱼种群,加快算法收敛速度。其次,通过使用非线性收敛因子,缓解算法全局搜索和局部搜索能力不平衡的现象。最后,在鲸鱼位置寻优过程中使用 Lévy 飞行策略,避免算法陷入局部最优,提升算法的全局搜索能力。通过对不同改进策略的有效性分析、与其他智能算法的对比分析,证明了 TELWOA 算法在收敛精度、算法稳定性和全局寻优能力上与对比算法有显著提升,具有一定的实际工程应用能力。

关键词:鲸鱼优化算法; Tent 混沌映射; 反向学习策略; 非线性收敛因子; Lévy 飞行策略

中图分类号: TP18

文献标识码: A

文章编号: 1673-629X(2024)04-0180-07

doi:10.20165/j.cnki.ISSN1673-629X.2024.0027

Whale Optimization Algorithm Based on Chaotic Elite and Lévy Flight Strategy

XIA Chao¹, OUYANG Ping¹, LI Ming^{2,3}, QU Ying-fei^{2,3*}, GUO Wei-feng¹

(1. Engineering Research Center for Waste Oil Recovery Technology and Equipment of Ministry of Education,

Chongqing Technology and Business University, Chongqing 400067, China;

2. Chongqing Engineering Laboratory for Detection, Control and Integrated System, Chongqing Technology and

Business University, Chongqing 400067, China;

3. School of Artificial Intelligence, Chongqing Technology and Business University, Chongqing 400067, China)

Abstract: For the problems of slow convergence and low accuracy of Whale Optimization Algorithm (WOA), the WOA based on Tent chaotic Elite and Lévy flight strategy (TELWOA) is proposed. The whale population is initialized by Tent chaotic mapping to maintain the population diversity, and the algorithm convergence speed is accelerated by introducing an elite opposition-based learning strategy to generate the inverse solution for the elite individuals of the initial population and select the population with high adaptation as the next generation whale population. Secondly, by using a nonlinear convergence factor, the imbalance between the algorithm's global search and local search ability is alleviated. Finally, the Lévy flight strategy is used in the whale location search process to avoid the algorithm from falling into local optimum and to improve the global search ability of the algorithm. By analyzing the effectiveness of different improvement strategies and comparing with other intelligent algorithms, it is proved that TELWOA has significant improvement in convergence accuracy, algorithmic stability and global optimization searching ability with comparison algorithms, and it has certain practical engineering application ability.

Key words: whale optimization algorithm; Tent chaotic mapping; opposition-based learning; nonlinear convergence factor; Lévy flight strategy

收稿日期: 2023-06-15

修回日期: 2023-10-18

基金项目: 重庆市教委重大科技项目(KJZD-M202200801); 重庆市教委科技项目(KJQN202200828); 重庆市研究生创新科研项目(yjcx2023-211-120)

作者简介: 夏超(1998-), 男, 硕士研究生, 研究方向为智能化收运、智能优化算法、机器学习等; 欧阳平(1979-), 男, 副研究员, 研究方向为废弃物循环利用技术; 李明(1982-), 男, 副教授, 研究方向为传感器网络理论及应用、智能计算及其应用研究等; 通信作者: 屈盈飞(1989-), 男, 讲师, 研究方向为复杂网络、物联网系统、智能系统等。

0 引言

元启发式算法是一种基于直观或经验构造的算法,因其依赖于相对简单的概念、解决局部最优易于实现以及涵盖范围较为广泛等特点,在工程应用领域越来越受欢迎。鲸鱼优化算法^[1] (Whale Optimization Algorithm, WOA) 是一种优秀的自然启发算法,由 Mirjalili 于 2016 年提出。该算法使用数学公式模仿鲸鱼捕食的行为,寻优过程包含三个阶段:包围猎物、气泡网攻击和随机搜索猎物。相比较于一般的启发式算法,鲸鱼优化算法具有参数少、全局搜索能力强、算法简单易于理解等特点,因此广泛应用于系统参数设计^[2]、传感器网络覆盖^[3]、神经网络参数优化^[4-6]等方面。与其他启发式算法类似,WOA 同样存在求解精度低、收敛速度慢以及算法后期会陷入局部最优解等缺陷。因此,国内外许多学者提出了一系列解决方案。Sayed 等^[7]将混沌映射引入到鲸鱼优化算法中,并对比了 10 种混沌映射方式,结果表明圆形混沌映射对于 WOA 算法的参数优化具有显著效果。G. Kaur 和 S. Arora^[8]将多种不同的混沌映射引入到 WOA 算法中,实现结果验证:基于 Tent 映射的 WOA 算法为所有结果中效果最好的。Chatterjee 等^[9]将混沌映射函数和基于对立的学习与 WOA 算法相结合,对于模糊神经网络的优化效果显著。耿召里等^[10]针对 WOA 算法在求解高维且复杂问题时收敛速度慢等问题,引入自适应惯性权重,调节鲸群寻优速度以及维护种群多样性,同时结合混合反向学习策略以及非线性收敛策略,平衡种群局部搜索和全局搜索能力。Sun 等^[11]提出了一种基于 Lévy 飞行^[12]和二次插值的改进 WOA 算法,避免算法陷入局部最优的同时,结合二次插值算法增强了 WOA 算法的求解精度。以上文献的改进算法通过不同策略在一定程度上提升了 WOA 算法的寻优能力,但其优化算法仍存在收敛速度慢精度低、后期陷入局部最优解等问题。

该文提出了基于多种改进策略的鲸鱼优化算法。首先,由于初始化种群对于算法的效果有着很大影响,使用 Tent 混沌映射初始化鲸鱼种群,保持种群的分布均匀性。其次,使用精英反向学习策略,从初始化种群中的精英个体中挑选寻优能力强的作为下一代种群,加快算法收敛速度。并且引入非线性收敛因子,缓解算法全局搜索和局部搜索能力不平衡的现象。最后,在鲸鱼寻优位置更新过程中引入 Lévy 飞行策略,在保持鲸鱼局部搜索和全局搜索能力的同时,避免算法陷入局部最优值。

1 基本鲸鱼优化算法

WOA 是澳大利亚学者 Mirjalili 通过模仿座头鲸

群体围捕猎物、气泡网觅食、螺旋收缩包围靠近猎物所提出的启发式算法。WOA 的数学模型主要包括三个部分:包围猎物、气泡网攻击和随机搜索猎物。

1.1 包围猎物

WOA 中鲸鱼能够识别猎物的位置并将其包围。在起始阶段,WOA 假设当前最优解是目标猎物或者接近最优解,其他鲸鱼将改变自己位置,逐渐向最优解靠拢。包围猎物的行为用以下数学方式表达:

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

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

其中, t 表示当前迭代次数, $\vec{X}^*(t)$ 是当前最优解的位置向量, \vec{A} , \vec{C} 是系数向量,由公式 3、公式 4 确定。

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad (3)$$

$$\vec{C} = 2 \cdot \vec{r}_2 \quad (4)$$

其中, \vec{r}_1 和 \vec{r}_2 为 $[0, 1]$ 之间的随机数, \vec{a} 为收敛因子,值从 2 线性变化减少到 0,如公式 5 所示:

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

其中, t 表示当前迭代次数, T_{\max} 表示最大的迭代次数。

1.2 气泡网攻击

鲸鱼在包围猎物的过程中以螺旋式上升靠近,不断缩小包围范围来捕获猎物,鲸鱼的气泡网攻击方式分为两种:收缩包围和螺旋更新。

收缩包围:鲸鱼的收缩包围行为主要通过式 5 中的线性收敛因子实现,随着迭代次数增加 a 的值逐渐减小,鲸鱼个体更新自身位置逐渐向猎物接近,实现包围。

螺旋更新:鲸鱼在收缩包围猎物过程中,使用螺旋方式逐渐靠近目标,用以下数学方式表达:

$$\vec{D} = |\vec{X}^*(t) - \vec{X}(t)| \quad (6)$$

$$\vec{X}(t+1) = \vec{D} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) \quad (7)$$

其中, \vec{D} 为鲸鱼个体到猎物之间的距离, b 为常数,用于定义对数螺旋形状, l 是 $[-1, 1]$ 中的随机数。

座头鲸在一个不断缩小的圆圈内围绕猎物移动,同时沿着螺旋形的路径游动。为了模拟这种同时的行为,假设有 50% 的概率在收缩环绕机制或螺旋更新之间进行选择,以更新鲸鱼的位置,其数学方式表达如下:

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D} & p < 0.5 \\ \vec{D} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) & p \geq 0.5 \end{cases} \quad (8)$$

其中, p 是 $[0, 1]$ 之间的随机数。

1.3 随机搜索猎物

在元启发式算法中, 算法的开发阶段和探索阶段的平衡十分重要。在 WOA 中, 通过向量 \vec{A} 的大小来体现算法的开发和探索平衡能力。当 $|\vec{A}| \geq 1$ 时, WOA 不再选择最优个体更新位置向量, 而是使用随机选择的鲸鱼进行位置更新, 数学方式表达如下:

$$\vec{D}_{\text{rand}} = |\vec{C} \cdot \vec{X}_{\text{rand}} - \vec{X}(t)| \quad (9)$$

$$\vec{X}(t+1) = \vec{X}_{\text{rand}} - \vec{A} \cdot \vec{D}_{\text{rand}} \quad (10)$$

其中, \vec{X}_{rand} 是随机选取的鲸鱼位置。

2 改进鲸鱼优化算法

通过分析鲸鱼优化算法发现, WOA 虽然具有参数少、结构简单、收敛效果较好等优点, 但仍存在收敛速度慢、全局收敛精度不理想等问题。因此, 该文将多种算法优化策略引入 WOA, 以提升算法寻优能力。

2.1 Tent 混沌映射初始化种群

初始化种群的多样性会在很大程度上影响智能优化算法的收敛速度和精度, 但是基本的鲸鱼优化算法都是随机初始化种群, 种群的多样性很难保证, 会导致算法的搜索效果和速度不佳。混沌具有随机性和遍历性等特点, 能使算法有更快的收敛速度并可在一定程度上避免陷入局部最优, 因此被广泛应用于智能算法的优化中。常见的混沌映射方式有 Tent 映射、Logistic 映射以及圆形映射等等。

由于 Tent 混沌映射有着更高的优化效率, 因此该文使用 Tent 混沌映射方式初始化鲸鱼种群, 其数学表达式如下:

$$Z_{i+1} = \begin{cases} \frac{Z_i}{0.7} & Z_i < 0.7 \\ \frac{10}{3}(1 - Z_i) & Z_i \geq 0.7 \end{cases} \quad (11)$$

Tent 混沌初始化种群步骤如下:

(1) 在搜索空间内初始化 $[0, 1]$ 区间内的 D 维鲸鱼个体;

(2) 使用式 11 的混沌映射策略将初始化的个体生成其混沌个体;

(3) 按照式 12 映射到原始搜索空间。

$$X_{ij} = \text{Tent}_{ij} \cdot (X_{\max j} - X_{\min j}) + X_{\min j} \quad (12)$$

($i = 1, 2, \dots, N; j = 1, 2, \dots, D$)

其中, N 为种群规模, D 为搜索维度, X_{ij} 为映射到搜索空间内的混沌变量, Tent_{ij} 为 Tent 混沌映射序列, $X_{\max j}$ 为每个维度的最大值, $X_{\min j}$ 为每个维度的最小值。

2.2 精英反向学习策略

反向学习 (Opposition-Based Learning, OBL)^[13] 是

2005 年提出的一种优化算法, 本身是机器学习的一种概念。其基本思想是以当前解为基础, 找到其对应的反向解, 并对比原始解和反向解的适应度值, 保存较好的解。反向解的引入可以扩大算法的搜索空间, 但由于反向学习具有一定的盲目性, 过多的计算反向解会浪费算法搜索时间。因此引入精英策略, 将初始化后的鲸鱼种群按适应度排序, 选取适应度较高的个体作为精英个体, 并计算其反向解, 选取精英解及其反向解中较优异的个体作为下一代种群的精英个体。在保证种群不断向最优值逼近的同时, 又减少了不必要的计算。精英反向策略公式如下:

$$\tilde{S}_{ij} = r_3(\text{lb}_j + \text{ub}_j) - S_{ij} \quad (13)$$

其中, S_{ij} 为精英解, r_3 为 $[0, 1]$ 上的随机值, lb_j , ub_j 为精英群体在 j 维的最小值和最大值, \tilde{S}_{ij} 为精英反向解。

2.3 非线性收敛因子

在基本的鲸鱼优化算法中, 鲸群的全局探索和局部开发能力主要依赖于收敛因子 a , 在 WOA 前期, 鲸群中最优个体较少, 需要加强种群的全局探索能力; 在算法后期, 种群更可能出现局部最优问题, 需要改变收敛因子变化量, 加强局部开发能力以跳出局部最优值。但由于收敛因子 a 是从 2 到 0 线性递减的, 无法较好地平衡算法全局探索能力和局部开发能力。因此, 该文提出一种非线性的收敛因子, 公式如下:

$$a = 2 - b \cdot \frac{T}{T_{\max}} - c \cdot (1 - e^{-10 \cdot \frac{T}{T_{\max}}}) \quad (14)$$

其中, T 为当前迭代次数, T_{\max} 为最大迭代次数, b 和 c 为控制因子, 用于控制收敛因子的变化程度, 选取 $b = 1.5$, $c = 0.5$ 。

在算法前期收敛因子减小速度较快, 增强种群全局搜索能力, 而迭代后期值缓慢收敛, 更有利于局部开发, 提升算法收敛精度。

2.4 Lévy 飞行策略更新种群位置

Lévy 飞行策略是一种连续跳跃的随机游走策略, 其步长是一种长尾分布, 概率密度函数是幂次形式的。

Lévy 飞行的轨迹的步长总是很小, 偶尔存在一个大步长。由于其高频小步长跳跃和低频长距离行走的特点, 能够在最优值附近进行范围精细搜索, 有效地提升了算法的探索范围, 在一定程度上可以缓解种群陷入局部最优解的问题。在 WOA 中使用 Lévy 飞行策略更新种群位置, 可以用以下公式来表示:

$$\vec{X}(t+1) = \vec{X}(t) + \text{Levy} \oplus (\vec{X}^s(t) - \vec{X}(t)) \quad (15)$$

其中, \oplus 表示点乘, t 为当前迭代次数; Levy 为步长系数, 服从 Lévy 飞行分布, 可以表示为:

$$\text{Levy} \sim u = t^{-\lambda}, \quad 1 < \lambda \leq 3 \quad (16)$$

$$s = \frac{\mu}{|v|^{1/\beta}}, \quad \mu \sim N(0, \sigma_\mu^2), \quad v \sim N(0, \sigma_v^2) \quad (17)$$

$$\begin{cases} \sigma_\mu = \left[\frac{\Gamma(1+\beta) \sin(\pi \cdot \beta/2)}{\Gamma[(1+\beta)/2] \sin(\pi \cdot \beta/2)} \right]^{1/\beta} \\ \sigma_v = 1 \end{cases} \quad (18)$$

式 17 中, s 表示 $\text{Levy}(\eta)$, μ 和 v 满足正态分布, $\beta = 1.5$, σ_μ 和 σ_v 的计算公式见式 18, 其中 Γ 是标准 Gamma 函数。

2.5 多策略改进鲸鱼优化算法流程

改进鲸鱼优化算法的具体步骤如下:

(1) 使用 Tent 混沌映射初始化鲸鱼种群, 并计算适应度;

(2) 使用精英反向学习策略选取精英鲸鱼个体, 计算其反向解, 比较精英个体及其反向种群, 选取适应度高的个体加入下一代鲸鱼种群中, 继续迭代;

(3) 根据公式 3、公式 4 更新 A, C , 并使用公式 14 更新收敛因子 a ;

(4) 当 $p < 0.5$ 时, 算法执行搜索包围阶段, $|\vec{A}| \geq 1$ 时, 种群随机选取一个鲸鱼个体寻优, 通过公式 10 更新位置; $|\vec{A}| < 1$ 时, 使用公式 2 进行位置更新;

(5) 当 $p \geq 0.5$ 时, 算法通过公式 7 进行螺旋方式位置更新;

(6) 通过 Lévy 飞行策略优化鲸鱼个体位置, 使用公式 15 更新位置, 并对比优化前后的适应度值, 若使用 Lévy 飞行策略优化后大于优化前的适应度值, 则保留 Lévy 飞行优化后的个体位置, 否则舍弃, 继续下一次迭代;

(7) 判断是否到达最大迭代次数, 若达到最大迭代次数, 则输出最优鲸鱼位置和适应度函数值, 否则继续执行步骤 3, 继续迭代。

算法的具体流程如图 1 所示。

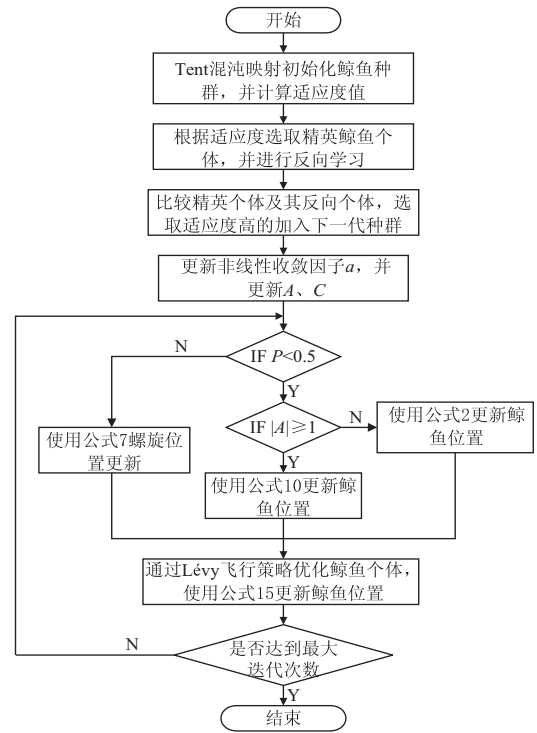


图 1 算法流程

3 实验分析

3.1 测试函数与参数设置

为验证所提算法的有效性, 选取 GWO^[14]、PSO、WOA 以及改进算法 TIWOA^[15]、IMWOA^[16] 和 GSWOA^[17] 进行性能分析。选取 10 个标准测试函数^[18] 验证算法性能, 所有基准函数见表 1。其中 $F_1 \sim F_4$ 为单峰函数, $F_5 \sim F_7$ 为多峰函数, $F_8 \sim F_{10}$ 为定维多峰函数。

设置最大迭代次数为 500, 种群个数为 30, 所有算法均使用 MATLAB R2018b 软件, 运行环境为 AMD Ryzen 5 5600U 处理器, 16G 内存, 操作系统为 Windows 11。

表 1 基准测试函数

函数	维度	区间	理论最优值
$F_1(x) = \sum_{i=1}^n x_i^2$	30	$[-100, 100]$	0
$F_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	$[-10, 10]$	0
$F_3(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j^2)$	30	$[-100, 100]$	0
$F_4(x) = \max_i \{ x_i \}, 1 \leq i \leq n$	30	$[-100, 100]$	0
$F_5(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]$	0
$F_6(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30	$[-600, 600]$	0

续表 1

函数	维度	区间	理论最优值
$F_7(x) = 0.1 \{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \} + \sum_{i=1}^n u(x_i, 10, 100, 4)$	30	$[-50, 50]$	0
$F_8(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	$[-5, 5]$	-1.031 6
$F_9(x) = - \sum_{i=1}^4 c_i \exp(- \sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2)$	6	$[0, 1]$	-3.32
$F_{10}(x) = - \sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	$[0, 10]$	-10.536 3

3.2 实验结果与分析

使用均值、标准差和最优值衡量算法性能,其中最优值和均值反映算法的寻优精度,标准差反映算法的稳定性。将每个算法独立运行 30 次,记录下测试函数的均值、标准差和最优值, $F_1 \sim F_4$ 为单峰函数,主要用于测试算法的最优寻优能力, $F_5 \sim F_7$ 为多峰函数,主要用于测试算法的适应性和全局寻优能力, $F_8 \sim F_{10}$ 为定维多峰函数,主要用于测试算法对固定输入的响应,评估算法的性能稳定性。

由表 2 数据可知,标准 WOA、PSO 和 GWO 的寻优能力都很有限, TIWOA 和 IMWOA 相比较于基本 WOA 有一定程度提升, GSWOA 和 TELWOA 在收敛精度上明显强于其他几种优化算法。在单峰测试函数

中, GSWOA 和 TELWOA 均收敛到理论最优值,大幅度领先其他优化算法,其中在函数 F_2 测试中, TELWOA 平均值为 0,表明 TELWOA 在具有较强寻优能力的同时,也保持了较好的算法稳定性。在多峰测试函数中, TELWOA 在 F_5 和 F_6 测试中均达到理论最优值,在函数 F_7 测试中虽未达到理论最优值,但是 TELWOA 的标准差和平均值都大幅度领先其他算法,证明该改进策略对 WOA 的寻优能力和算法稳定性上起到了积极作用。在定维多峰函数测试中, TELWOA 总体上领先其他优化函数,稳定性较强。

为了更为直观地反映算法的收敛性能,该文针对不同测试函数给出了 WOA、PSO、GWO、TIWOA、IMWOA、GSWOA 和 TELWOA 的收敛曲线,见图 2。

表 2 仿真实验结果

函数	指标	WOA	PSO	GWO	TIWOA	IMWOA	GSWOA	TELWOA
F_1	平均值	1.16E-74	2.06E-07	1.86E-27	3.67E-120	9.41E-95	0	0
	标准差	4.52E-74	6.51E-07	2.08E-27	1.80E-119	5.06E-94	0	0
	最优值	2.91E-86	3.54E-14	9.16E-29	5.15E-133	2.70E-112	0	0
F_2	平均值	8.72E-52	0.117 805 56	9.43E-17	1.18E-71	4.02E-61	4.79E-317	0
	标准差	2.48E-51	0.231 496 29	6.81E-17	4.53E-71	1.62E-60	0	0
	最优值	6.92E-58	0.001 020 96	1.18E-17	4.25E-80	7.59E-68	0	0
F_3	平均值	44 329.421 9	99.702 729	6.61E-05	6 066.038 81	5 705.531 82	0	0
	标准差	9 849.776 81	96.682 369 3	2.10E-04	15 819.123 1	3 058.996 84	0	0
	最优值	23 864.644 3	14.874 985 9	1.68E-08	3.521 3E-17	629.590 29	0	0
F_4	平均值	43.412 851	2.464 457 6	9.98E-07	0.849 425	0.118 694 3	0	0
	标准差	27.879 496	1.004 349 9	1.16E-06	1.251 739	0.152 343 8	0	0
	最优值	0.103 362	0.789 204 5	1.37E-07	1.752 E-05	2.65E-06	0	0
F_5	平均值	0	50.477 604	2.645 376 6	7.997 112	14.572 074	0	0
	标准差	0	15.411 616	3.397 105 6	20.838 562	27.921 92	0	0
	最优值	0	21.889 089	5.684 E-14	0	0	0	0

续表 2

函数	指标	WOA	PSO	GWO	TIWOA	IMWOA	GSWOA	TELWOA
F_6	平均值	0.012 021 4	0.026 714 5	0.003 307	0	0.003 988 7	0	0
	标准差	0.045 129 9	0.027 507 4	0.008 529 6	0	0.019 127	0	0
	最优值	0	1.613 E-12	0	0	0	0	0
F_7	平均值	0.487 467	0.200 948 3	0.681 842 1	0.321 272 5	1.95E-05	0.000 259 8	1.49E-07
	标准差	0.209 841 8	0.473 690 2	0.269 825 4	0.276 910 7	3.08E-05	0.000 433 8	5.66E-07
	最优值	0.112 148 3	1.071 E-09	0.199 324 7	0.026 830 9	2.85E-08	4.48E-07	5.13E-16
F_8	平均值	-1.031 628 5	-1.031 628 5	-1.031 628 4	-1.031 618 7	-1.031 628 4	-0.999 937 9	-1.031 628 5
	标准差	7.688 4E-10	6.409 9E-16	3.062 6E-08	1.534 8E-05	6.99E-08	0.000 156 15	4.95E-09
	最优值	-1.031 628 5	-1.031 628 5	-1.031 628 5	-1.031 628 5	-1.031 628 5	-0.999 998 5	-1.031 628 5
F_9	平均值	-3.217 085 8	-3.282 364 1	-3.243 234 5	-3.147 879 4	-2.989 307 9	-2.958 752 4	-3.299 830 4
	标准差	0.097 294 64	0.056 046 75	0.076 649 53	0.083 983 3	0.158 538 31	0.219 106 35	0.037 952 55
	最优值	-3.321 698 5	-3.321 995 2	-3.321 990 5	-3.320 090 5	-3.279 341 7	-3.264 178 2	-3.321 390 6
F_{10}	平均值	-7.142 560 3	-6.466 980 6	-10.263 698	-10.044 018	-10.265 042	-10.281 263	-10.530 365
	标准差	3.190 001 1	3.848 315 29	1.456 229 97	0.521 837 49	1.460 605 88	0.496 957 04	0.009 201 5
	最优值	-10.535 437	-10.536 41	-10.535 979	-10.530 063	-10.536 403	-10.535 693	-10.536 3

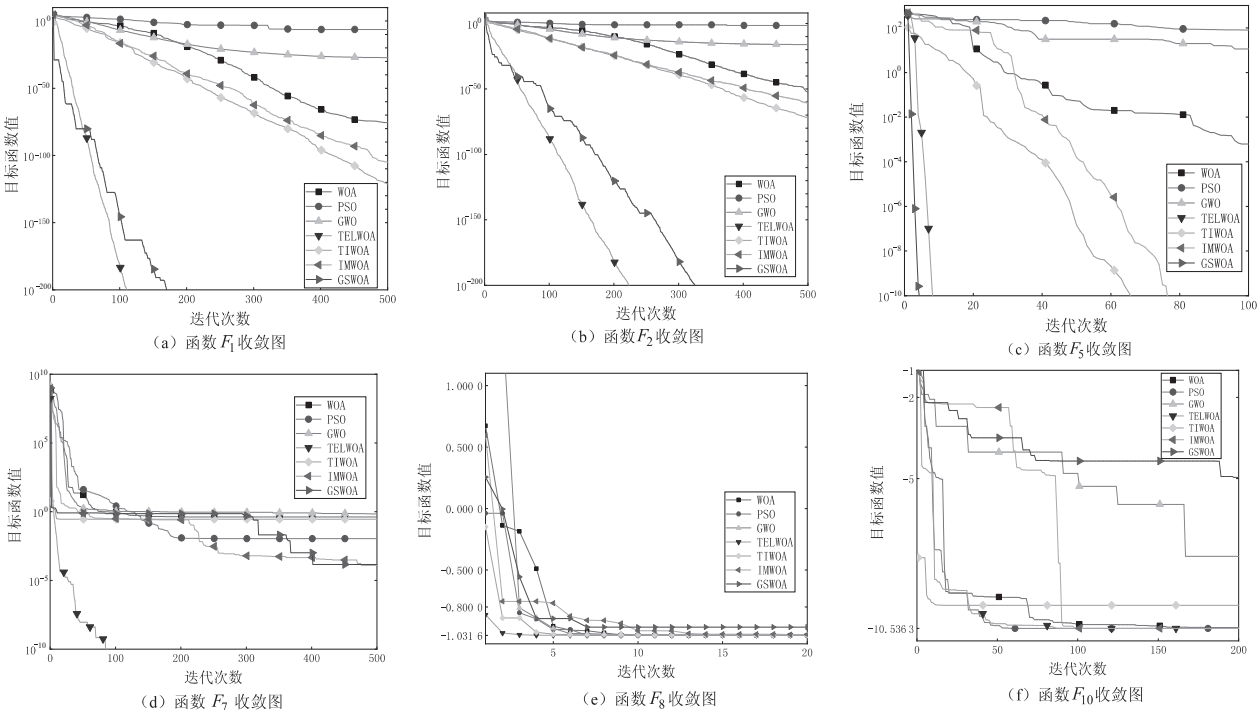


图 2 与其他算法比较收敛图

从收敛曲线可知,在 F_1 和 F_2 为代表的单峰函数测试中,TELWOA 的收敛速度都是最好的且都收敛到理论最优值,其收敛能力明显好于标准 WOA 及其改进算法,具有很强的寻优能力。在 F_5 和 F_7 为代表的多峰函数测试中,TELWOA 收敛速度很快,在测试函

数 F_5 中前 20 次迭代就已经收敛到理论最优,在测试函数 F_7 收敛精度大幅领先其它算法。最后,在 F_8 和 F_{10} 为代表的定维多峰函数测试中,TELWOA 均在前 20 次迭代以及收敛到理论最优,在 F_8 上甚至第 2 次迭代就收敛到最优值,表明 TELWOA 在保持很高收敛

精度的同时具有较好的算法稳定性。

通过以上对比分析,TELWOA 在复杂函数的最优值求解中,收敛精度高,全局搜索能力强,算法稳定性好,各方面性能优于标准鲸鱼算法及大部分改进算法,是一种性能较好的改进鲸鱼算法。

4 结束语

针对 WOA 收敛速度慢、收敛精度低等问题,提出了基于多种改进策略的鲸鱼优化算法。该算法采用 Tent 混沌映射初始化种群,保持种群的分布均匀性,提升种群多样性;使用精英反向学习策略,从初始化种群以及精英个体中挑选寻优能力强的作为下一代种群,加快算法收敛速度;通过增加非线性收敛因子,缓解算法全局搜索和局部搜索能力不平衡的现象。最后,在鲸鱼寻优位置更新过程中引入 Lévy 飞行策略,在最优位置附近进行精细搜索,提升算法全局搜索能力,避免陷入局部最优。通过仿真实验分析,该算法具有较好的收敛精度和全局搜索能力,同时收敛速度较快,可应用于实际工程问题优化中,具有较好的实用价值。

参考文献:

- [1] MIRJALILI S, LEWIS A. The whale optimization algorithm [J]. *Advances in Engineering Software*, 2016, 95: 51–67.
- [2] RIYABI M, RAFSANJANI M K, GUPTA B B, et al. Multiobjective whale optimization algorithm-based feature selection for intelligent systems [J]. *International Journal of Intelligent Systems*, 2022, 37 (11): 9037–9054.
- [3] DEEPA R, VENKATARAMAN R. Enhancing whale optimization algorithm with levy flight for coverage optimization in wireless sensor networks [J]. *Computers & Electrical Engineering*, 2021, 94: 107359.
- [4] WANG J, DU P, NIU T, et al. A novel hybrid system based on a new proposed algorithm—multi-objective whale optimization algorithm for wind speed forecasting [J]. *Applied Energy*, 2017, 208: 344–360.
- [5] MURALI P, REVATHY R, BALAMURALI S, et al. Integration of RNN with GARCH refined by whale optimization algorithm for yield forecasting: a hybrid machine learning approach [J]. *Journal of Ambient Intelligence and Humanized Computing*, 2020, 11 (3): 1–13.
- [6] ALJARAH I, FARIS H, MIRJALILI S. Optimizing connection weights in neural networks using the whale optimization algorithm [J]. *Soft Computing*, 2018, 22 (1): 1–15.
- [7] SAYED G I, DARWISH A, HASSANIEN A E. A new chaotic whale optimization algorithm for features selection [J]. *Journal of Classification*, 2018, 35 (2): 300–344.
- [8] KAUR G, ARORA S. Chaotic whale optimization algorithm [J]. *Journal of Computational Design and Engineering*, 2018, 5 (3): 275–284.
- [9] CHATTERJEE R, MUKHERJEE R, ROY P K, et al. Chaotic oppositional-based whale optimization to train a feed forward neural network [J]. *Soft Computing*, 2022, 26 (22): 12421–12443.
- [10] 耿召里, 李 目, 曹淑睿, 等. 基于混合反向学习策略的鲸鱼优化算法 [J]. *计算机工程与科学*, 2022, 44 (2): 355–363.
- [11] SUN Y, WANG X, CHEN Y, et al. A modified whale optimization algorithm for large-scale global optimization problems [J]. *Expert Systems with Applications*, 2018, 114: 563–577.
- [12] BARTHELEMY P, BERTOLOTI J, WIERSMA D S. A Lévy flight for light [J]. *Nature*, 2008, 453 (7194): 495–498.
- [13] TIZHOOSH H R. Opposition-based learning: a new scheme for machine intelligence [C]//*Proceedings of the international conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CI-MCA-IAWTIC'06)*. Vienna: IEEE, 2005: 695–701.
- [14] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer [J]. *Advances in Engineering Software*, 2014, 69: 46–61.
- [15] 黄 飞, 吴泽忠. 基于阈值控制的一种改进鲸鱼算法 [J]. *系统工程*, 2020, 38 (2): 133–148.
- [16] 吴泽忠, 宋 菲. 基于改进螺旋更新位置模型的鲸鱼优化算法 [J]. *系统工程理论与实践*, 2019, 39 (11): 2928–2944.
- [17] 刘 磊, 白克强, 但志宏, 等. 一种全局搜索策略的鲸鱼优化算法 [J]. *小型微型计算机系统*, 2020, 41 (9): 1820–1825.
- [18] MOLGA M, SMUTNICKI C. Test functions for optimization needs [J]. *Test Functions for Optimization Needs*, 2005, 101: 48.