

控制与决策

Control and Decision

基于围攻机制的改进鲸鱼优化算法

王延峰, 廖荣航, 梁恩豪, 孙军伟

引用本文:

王延峰, 廖荣航, 梁恩豪, 孙军伟. 基于围攻机制的改进鲸鱼优化算法[J]. *控制与决策*, 2023, 38(10): 2773–2782.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2022.0425>

您可能感兴趣的其他文章

Articles you may be interested in

[嵌入Circle映射和逐维小孔成像反向学习的鲸鱼优化算法](#)

Whale optimization algorithm for embedded Circle mapping and one-dimensional oppositional learning based small hole imaging
控制与决策. 2021, 36(5): 1173–1180 <https://doi.org/10.13195/j.kzyjc.2019.1362>

[基于平衡鲸鱼优化算法的无人车路径规划](#)

Path planning of unmanned ground vehicle based on balanced whale optimization algorithm
控制与决策. 2021, 36(11): 2647–2655 <https://doi.org/10.13195/j.kzyjc.2020.0416>

[基于自适应正态云模型的灰狼优化算法](#)

Grey wolf optimization algorithm based on adaptive normal cloud model
控制与决策. 2021, 36(10): 2562–2568 <https://doi.org/10.13195/j.kzyjc.2020.0233>

[求解约束优化问题的改进果蝇优化算法及其工程应用](#)

Improved fruit fly optimization algorithm for solving constrained optimization problems and engineering applications
控制与决策. 2021, 36(2): 314–324 <https://doi.org/10.13195/j.kzyjc.2019.0557>

[基于反时限混沌郊狼优化算法的BP神经网络参数优化](#)

Parameter optimization of BP neural network based on coyote optimization algorithm with inverse time chaotic
控制与决策. 2021, 36(10): 2339–2349 <https://doi.org/10.13195/j.kzyjc.2020.0408>

基于围攻机制的改进鲸鱼优化算法

王延峰, 廖荣航, 梁恩豪, 孙军伟[†]

(郑州轻工业大学 电气信息工程学院, 郑州 450002)

摘要: 针对传统鲸鱼优化算法全局探索能力不足、收敛精度低、速度慢等问题, 提出一种基于围攻机制的改进鲸鱼优化算法. 首先, 使用 Tent 混沌映射和非线性参数, 使种群的分布更均匀, 并且协调了鲸鱼优化算法的探索与开发能力; 其次, 考虑算法适应度在寻优中的重要作用, 提出限制适应度控制和高斯检测机制; 最后, 结合哈里斯鹰优化算法的围攻机制, 提升鲸鱼优化算法的全局探索和局部寻优的能力. 将改进的算法与多种算法在 13 个可变维基准函数上进行仿真测试, 结果表明, 基于围攻机制的改进鲸鱼优化算法, 在拥有较好鲁棒性和稳定性的同时, 能够保证收敛精度与速度.

关键词: 鲸鱼优化算法; Tent 混沌映射; 非线性参数; 适应度控制; 围攻机制

中图分类号: TP18 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.0425

引用格式: 王延峰, 廖荣航, 梁恩豪, 等. 基于围攻机制的改进鲸鱼优化算法[J]. 控制与决策, 2023, 38(10): 2773-2782.

Improved whale optimization algorithm based on siege mechanism

WANG Yan-feng, LIAO Rong-hang, LIANG En-hao, SUN Jun-wei[†]

(School of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China)

Abstract: An improved whale optimization algorithm based on siege mechanism is proposed to solve the problems of the traditional whale optimization algorithm, such as its insufficient global exploration ability, low convergence accuracy and slow speed. Firstly, the algorithm uses Tent chaos mapping and nonlinear parameters to make the population distribution more uniform and coordinate the exploration and development ability of whale optimization algorithm. Then, considering the important role of algorithm fitness in optimization, the limited fitness control and Gaussian detection mechanism are proposed. Finally, combined with the siege mechanism of the Harris hawks optimization algorithm, the global exploration and local optimization capabilities of the whale optimization algorithm are improved. The results show that the improved whale optimization algorithm based on the siege mechanism has good robustness and stability, while ensuring the convergence accuracy and speed.

Keywords: whale optimization algorithm; Tent chaos mapping; nonlinear parameters; fitness control; siege mechanism

0 引言

鲸鱼优化算法 (whale optimization algorithm, WOA) 是 Mirjalili 等^[1]提出的一种基于群体的智能优化算法. 该算法利用数学公式, 形象地模拟了鲸鱼捕食的行为, 与传统的元启发式优化算法相比, 鲸鱼优化算法具有原理简单、参数设置少、寻优能力强等特点^[2], 同时也存在陷入局部最优、收敛速度缓慢、收敛精度低等缺陷. 鲸鱼优化算法的优点与缺点同时存在, 使得鲸鱼优化算法的适用领域变得有限, 因此, 提出一种可以充分发挥鲸鱼优化算法优点, 同时可以

改进鲸鱼优化算法中已知缺点的算法, 具有十分重要的学术意义.

哈里斯鹰优化算法 (Harris hawks optimization, HHO) 是 Heidari 等^[3]提出的一种模拟哈里斯鹰捕食行为的群智能算法. 该算法融入了有关逃逸、群体中心、随机种群划分等相关理念^[4], 对许多连续非约束问题都具有较好的求解能力. 哈里斯鹰在觅食过程中, 会与鹰群的其他个体进行合作, 哈里斯鹰优化算法把这种合作行为, 形象地利用数学公式表达出来, 建立了拥有快速收敛和种群交流性质的软、硬围攻

收稿日期: 2022-03-18; 录用日期: 2022-06-24.

基金项目: 国家自然科学基金河南联合基金重点项目 (U1804262); 中原科技创新领军人才项目 (204200510003).

责任编委: 夏元清.

[†]通讯作者. E-mail: junweisun@yeah.net.

机制,与鲸鱼优化算法相比,哈里斯鹰优化算法拥有收敛速度快和全局寻优的优势。

针对传统鲸鱼优化算法易陷入局部最优、收敛精度低等缺陷,学者们提出了许多改进策略。例如, Sun 等^[5]通过引入二次插值和莱维飞行策略提高求解精度和跳出局部最优,并改进非线性递减收敛因子以更好地解决非线性问题。林杰等^[6]通过引入混沌映射让种群分布更均匀,并加入正余弦算法筛选领导者和自适应权重的方式来改进基本的鲸鱼优化算法。黄清宝等^[7]加入同步余弦惯性权值和多项式变异,增强了传统鲸鱼优化算法跳出局部最优的能力。本文提出一种基于围攻机制的改进鲸鱼优化算法,改进方法如下。

首先,通过 Tent 混沌映射让种群分布更均匀^[8],使算法的收敛速度加快,在此基础上,引进非线性收敛参数 a ,使改进后的鲸鱼算法能够利用改进的参数适应非线性问题;其次,在每次迭代开始时,使用限制种群控制方法,当算法适应度多次循环未更新时,使用螺旋位置更新方法,防止算法出现局部最优;然后,借鉴哈里斯鹰优化算法的种群围攻策略,改进鲸鱼优化算法的收缩包围机制,加快所有鲸鱼个体对最优位置的寻找,尽可能地避免某一个体在无用位置探索,浪费计算资源;最后,在算法每次迭代结束时,利用高斯检测机制对鲸鱼位置的适应度进行更新判断^[9],使鲸鱼算法拥有更优的探索位置,加快鲸鱼优化算法的收敛速度。

1 基本 WOA 算法

鲸鱼优化算法模拟了鲸鱼的捕食动作,根据鲸鱼的捕食特点,将鲸鱼的捕食过程分为3个步骤,即3种位置更新方式:收缩包围、螺旋位置更新以及随机搜寻^[10]。

1.1 收缩包围

鲸鱼能感知猎物所在的区域并将其包围。由于最优设计在狩猎或搜索空间中的位置与之前的位置不一致,WOA 优化算法假设当前的最佳候选解是目标猎物或接近于最优解。在这种情况下,鲸鱼定义了最好的搜索代理;然后,其他搜索代理将试图改变位置,向最好的搜索代理靠拢。收缩包围的捕猎行为可用以下公式描述:

$$X(t+1) = X^*(t) - A \cdot D_1, \quad (1)$$

$$D_1 = |C \cdot X^*(t) - X(t)|. \quad (2)$$

其中: t 表示当前迭代次数, A 和 C 是向量系数, $X(t)$ 是当前时刻位置, $X(t+1)$ 是下一时刻位置, D_1 是 C

倍猎物位置与当前鲸鱼位置的差值绝对值, $X^*(t)$ 是目前得到最优解的位置向量。每次迭代结果若有更好的解决方案,即此时位置的适应度值小于 $X^*(t)$ 的适应度值,则将此时鲸鱼位置向量设置为新的 X^* 。

式(1)、(2)中 A 和 C 的计算公式如下:

$$A = 2a \cdot r_1 - a, \quad (3)$$

$$C = 2r_2, \quad (4)$$

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

其中: r_1 、 r_2 是 $[0, 1]$ 之内的随机数; T_{\max} 是最大迭代次数,本文取 $T_{\max} = 500$; a 的取值范围是 $[0, 2]$,随着 t 的增加而线性改变。

1.2 位置更新

鲸鱼位置探索更新的方式分为2种,一是螺旋位置更新,二是随机搜寻。为了模拟鲸鱼在某时刻的位置更新方式,保证鲸鱼在同一时刻拥有相等的概率选择螺旋位置更新或随机搜寻方式,设置取值范围为 $[0, 1]$ 的随机数字 p 。通过 p 值大小随机选择鲸鱼位置的更新方式。

1.2.1 螺旋位置更新

当 $p \geq 0.5$ 时,选择螺旋位置更新的方法,通过模拟鲸鱼螺旋包围猎物的方式,建立螺旋位置更新方程,用来更新下一次鲸鱼的位置。其计算公式如下:

$$X(t+1) = D_2 \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t), \quad (6)$$

$$D_2 = |X^*(t) - X(t)|, \quad (7)$$

$$a_1 = -1 - \frac{t}{T_{\max}}, \quad (8)$$

$$l = (a_1 - 1) \cdot r_3 + 1. \quad (9)$$

其中: D_2 表示猎物与鲸鱼之间的距离; b 是控制螺旋形状的参数,本文设置为1; a_1 是 $[-2, -1]$ 线性变化的参数; r_3 是 $[0, 1]$ 之间随机数, l 的取值范围是 $[-2, 1]$ 。

1.2.2 随机搜寻

当 $p < 0.5$ 时,选择随机搜寻位置更新公式。随机搜寻分为两种方式:当 $|A| < 1$ 时,表示鲸鱼在朝向猎物位置移动,此时使用收缩包围公式模拟鲸鱼的动作行为,即使用公式(1)来包围猎物;当 $|A| \geq 1$ 时,表示鲸鱼移动位置越过了猎物存在的位置,此时鲸鱼会放弃之前的移动方向,随机向其他方向搜索新的更新位置,避免陷入局部极值。有

$$D_{\text{rand}} = |C \cdot X_{\text{rand}}(t) - X(t)|, \quad (10)$$

$$X(t+1) = X_{\text{rand}}(t) - A \cdot D_{\text{rand}}. \quad (11)$$

其中: X_{rand} 表示随机选择的鲸鱼位置向量, D_{rand} 表示 C 倍 X_{rand} 与 $X(t)$ 差值的绝对值。

2 改进鲸鱼优化算法

在基本鲸鱼优化算法中, 鲸鱼位置的更新过程是通过随机选择 3 种位置更新机制实现的, 因此, 存在鲸鱼位置更新中无法选择最有效更新方式的问题; 而且在算法的搜索过程中存在多次迭代但未改变领导者 $X^*(t)$ 的位置, 导致收敛过程提前结束的问题, 即在求解优化问题时可能会迅速收敛到局部最优, 最终使优化算法解的质量下降. 针对传统鲸鱼优化算法中存在的问题, 本文提出基于围攻机制的改进鲸鱼优化算法.

首先, 通过混沌 Tent 映射随机生成算法的初始种群位置, 让种群分布更均匀, 加快算法的收敛速度; 提出新的非线性参数 a , 使得鲸鱼优化算法可以适应复杂的非线性问题; 引入适应度控制机制, 通过控制种群位置更新, 防止更新停滞, 提高算法跳出陷入局部最优的能力; 引入哈里斯鹰围攻机制, 加快鲸鱼的捕猎速度; 最后, 在每次鲸鱼捕猎迭代结束时, 加入高斯检测的位置控制机制以增加算法的寻优精度. 基于围攻机制的改进鲸鱼优化算法 (IWOA) 的流程如图 1 所示.

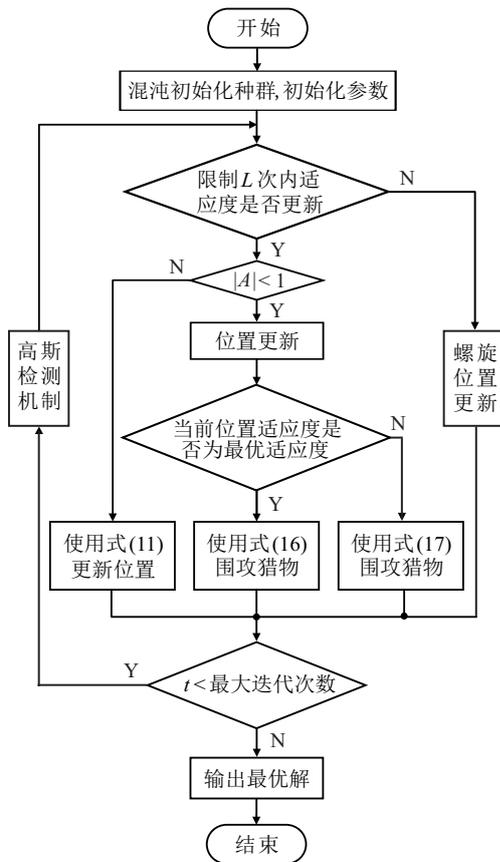


图 1 基于围攻机制的改进鲸鱼优化算法流程

2.1 Tent混沌映射

鲸鱼算法在解决函数优化问题中, 通常利用随机产生的数据作为初始种群位置, 因而难以保证种群的

多样性, 会造成算法的寻优结果较差. 一方面, 混沌映射具有随机性、规律性以及遍历性的特征; 另一方面, 对于基于群体迭代的智能优化算法而言, 初始种群的好坏影响着算法的求解精度和收敛速度^[11]. 因此, 在求解函数优化问题时, 利用混沌映射的这些特性能够使算法有效逃离局部最优, 从而维持种群的多样性, 同时提高全局搜索能力. 本文利用 Tent 映射初始化种群, 参考文献 [12] 取 $u = 0.7$, 其限制参数表达式为

$$X(t+1) = \begin{cases} \frac{X(t)}{0.7}, & 0 \leq X(t) < 0.7; \\ \frac{1-X(t)}{0.3}, & 0.7 \leq X(t) \leq 1. \end{cases} \quad (12)$$

2.2 非线性参数

对于群智能优化算法而言, 探索 and 开发能力对于其优化性能至关重要. 而对于 WOA, 其位置更新中的收缩包围和随机搜寻都与 a 的取值有关, 如何选取合适的收敛因子 a 来协调 WOA 的探索与开发能力是一个值得深究的问题. 探测能力是群体利用位置更新方式对广泛搜索区域探索, 避免算法陷入局部最优的关键; 开发能力主要是利用群体已有的信息对解空间的某些邻域进行局部搜索, 对算法的收敛速度产生决定性影响. 变化较大的收敛因子 a 具有较好的全局搜索能力, 避免算法陷入局部最优; 较小的收敛因子 a 具有较强的局部搜索能力, 可以加快算法的收敛速度^[13]. 然而, 鲸鱼优化算法中的收敛因子 a 是随着迭代次数从 2 线性递减到 0 的, 并不能完全体现出 WOA 的探索和开发过程.

本文设计一个前期变化迅速而后后期变化相对缓慢的非线性递减收敛因子 a , 用于平衡 WOA 的探索与开发, 其表达式为

$$a = 2 \left(1 - \sqrt{\frac{t}{T_{\max}}} \right). \quad (13)$$

其中: T_{\max} 是最大迭代次数, t 是当前迭代次数. 参数 A 由系数 a 直接控制, 系数 a 的改变导致随机搜寻和收缩包围机制都发生一定变化.

2.3 限制适应度控制

在多次算法的迭代过程中, 存在最优位置在多次迭代中未更新的情况. 当算法多次在已探索位置持续探索时, 既浪费计算资源, 又容易陷入局部最优的不良状态.

本文提出一种限制适应度控制方法, 其主要思想是: 首先, 利用此时最优位置代入算法的适应度函数, 计算出其对应的适应度值; 其次, 根据最大迭代次数, 通过下式计算得到限制次数 L :

$$L = \lfloor T_{\max} / 100 \rfloor. \quad (14)$$

其中: L 表示限制次数, $\lfloor \cdot \rfloor$ 表示向下取整.

当迭代更新后鲸鱼位置的适应度值连续 L 次仍未发生变化时, 使用螺旋位置更新公式对当前位置进行更新计算; 否则, 根据 $|A|$ 来判断下次位置更新公式.

2.4 哈里斯鹰围攻机制

哈里斯鹰优化算法通过模拟哈里斯鹰的捕食动作, 利用数学公式对其动作进行仿真模拟, 该算法生动地模拟了哈里斯鹰的围攻捕食机制, 使得算法具有极强的全局搜索能力. 在传统的鲸鱼优化算法中, 寻找最优位置的过程常常是单个鲸鱼个体的随机探索, 个体与群体之间缺乏交流, 使得存在某些个体在距离猎物较远处进行多次无用的探索. 因此, 参考文献[3]中哈里斯鹰的围攻策略, 对鲸鱼优化算法位置进行一定的改进, 改进公式如下:

$$X(t+1) = \begin{cases} Y, & f(Y) < f(X(t)); \\ Z, & f(Z) < f(X(t)). \end{cases} \quad (15)$$

$$Y = X^*(t) - A \cdot D_1. \quad (16)$$

$$Z = Y + S \cdot \text{LF}(D). \quad (17)$$

其中: S 为 $(1, D)$ 均匀分布上的 D 维随机向量; $f(x)$ 为 x 的位置适应值, 表示将某一位置代入适应度函数计算其适应度值; $\text{LF}(D)$ 为莱维飞行产生的 D 维随机向量, 莱维飞行公式为

$$\text{LF}(D) = 0.01 \times \frac{u \cdot \sigma}{|v|^{\frac{1}{\beta}}}, \quad (18)$$

$$\sigma = \left(\frac{\Gamma(1+\beta) \cdot \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \cdot \beta \cdot 2^{\frac{\beta-1}{2}}}\right)^{\frac{1}{\beta}}. \quad (19)$$

u 和 v 为 $(0,1)$ 之间的随机值, $\Gamma(x)$ 为 Gamma 函数, β 设为 1.5.

2.5 高斯检测机制

该部分对当前位置使用高斯变异, 并对变异后的位置适应度与检测前的位置适应度进行比较判断, 选取最优位置. 主要目的是为了提高算法跳出局部最优的能力, 提升算法的寻优能力. 高斯检测机制公式如下:

$$X(N) = X(t) + X(t) \cdot N(0,1). \quad (20)$$

$$X_{t+1} = \begin{cases} X(N), & f(X(N)) < f(X(t)); \\ X(t), & f(X(N)) > f(X(t)). \end{cases} \quad (21)$$

其中: $N(0,1)$ 生成一个 $0 \sim 1$ 之间高斯分布的随机数, $X(N)$ 为高斯变异后生成的位置向量.

3 函数测试实验设计

3.1 基准测试函数

本文选择 13 个可变维的基准函数, 用来对改进算法进行测试评估. 其中: $F_1 \sim F_7$ 为单峰基准函数, 只有一个峰值; $F_8 \sim F_{13}$ 为多峰基准函数, 有多个峰值^[14]. 所有基准函数都有理论最优值, 即此测试函数的极值. 基准函数如表 1 所示.

表 1 单峰、多峰基准函数

编号	函数名称	维度	区间	理论最优值
F_1	Sphere	30	$[-100, 100]$	0
F_2	Schwefel 2.22	30	$[-10, 10]$	0
F_3	Schwefel 1.2	30	$[-100, 100]$	0
F_4	Schwefel 2.21	30	$[-100, 100]$	0
F_5	Rosenbrock	30	$[-30, 30]$	0
F_6	Step	30	$[-100, 100]$	0
F_7	Quartic	30	$[-1.28, 1.28]$	0
F_8	Schwefel 2.26	30	$[-500, 500]$	-418.9829×5
F_9	Rastrigin	30	$[-5.12, 5.12]$	0
F_{10}	Ackley	30	$[-32, 32]$	0
F_{11}	Griewank	30	$[-600, 600]$	0
F_{12}	Penalized1	30	$[-50, 50]$	0
F_{13}	Penalized2	30	$[-50, 50]$	0

3.2 实验方法

通过 3 组实验测试对本文所提出的 IWOA 算法的收敛精度和稳定性进行分析: 1) 与原始优化算法比较; 2) 与不同优化算法比较; 3) 与不同改进策略的 WOA 算法比较.

选取基本算法: 鲸鱼优化算法 (WOA)、哈里斯鹰优化算法 (HHO)、灰狼优化算法 (GWO)^[15]、飞蛾扑火优化算法 (MFO)^[16]、蚁狮优化算法 (ALO)^[17]、樽海鞘群算法 (SSA)^[18] 以及其他改进鲸鱼优化算法 EWOA^[19]、GSWOA^[20]、IWOA-1^[21]、WOABAT^[22], 在 $F_1 \sim F_{13}$ 可变维基准函数上, 与本文所提出的基于围攻机制的改进鲸鱼优化算法进行对比实验.

3.3 参数与实验设计

为了保证对比实验的公平性, 对比算法的种群和迭代次数统一设置为相同数值: 种群设置为 30, 迭代次数设置为 500. 通过测试多维度基准函数, 分析不同优化算法在不同维度下的收敛性能.

为了避免随机性, 保证实验的准确性, 对所有优化算法均进行 30 次独立运行. 因为均值 (mean) 可以反映每个算法的寻优精度, 标准差 (std) 可以直观地反映每个算法的鲁棒性和稳定性^[23], 所以通过对比各优化算法单独运行 30 次所得到最优适应度的平均值和标准差, 分析优化算法的平均收敛精度和稳定性. 首先, 通过符号检验分析不同维度上 IWOA 算法的平均结果是否优于、等于或劣于所对比的优化算法; 其次, 通过 Friedman 检验和 Wilcoxon 检验分析优

化算法的优异性和显著差异性.

4 实验结果与分析

4.1 与原始优化算法比较

通过统一设置相同的维度、种群和迭代次数,在保证算法对比公平性的基础上,在 30 维、100 维、300 维和 500 维上使用可变维基准函数 $F_1 \sim F_{13}$ 对基本 WOA、HHO、IWOA 算法进行对比实验验证.

表 2 为 3 种原始算法在 30 维、100 维、300 维和 500

维下单独运行 30 次最优值的平均值和标准差计算结果. 图 2 给出 3 种原始优化算法,在 4 种维度下 F_5 、 F_7 、 F_9 、 F_{12} 的收敛曲线. 其中:横轴为迭代次数,纵轴表示最优适应度值. 从图 2 中可以明显看出:在不同维度上,3 种优化算法的收敛特性并无较大变化;IWOA 算法仅在 F_7 测试函数上效果略低于 HHO 算法,在 F_5 、 F_9 、 F_{12} 上 IWOA 算法表现出优秀的收敛精度与收敛速度.

表 2 WOA、HHO、IWOA 算法对比

函数	算法	30		100		300		500	
		mean	std	mean	std	mean	std	mean	std
F_1	WOA	1.52e-72	5.45e-72	1.02e-71	4.48e-71	4.09e-72	9.63e-72	2.22e-66	9.74e-66
	HHO	3.44e-97	1.10e-96	3.40e-94	1.32e-93	6.97e-100	1.83e-99	4.90e-91	2.20e-90
	IWOA	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
F_2	WOA	3.67e-51	1.27e-50	1.12e-49	4.80e-49	9.80e-50	1.87e-49	6.52e-48	3.37e-47
	HHO	1.98e-48	1.05e-47	8.26e-51	3.39e-50	3.26e-50	6.07e-50	1.02e-49	4.79e-49
	IWOA	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
F_3	WOA	4.29e+04	1.22e+04	1.12e+06	3.41e+05	1.08e+07	3.69e+06	2.77e+07	6.97e+06
	HHO	2.26e-76	1.21e-75	1.14e-59	5.55e-59	1.75e-46	5.26e-46	6.15e-37	1.84e-36
	IWOA	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
F_4	WOA	5.31e+01	2.55e+01	7.92e+01	1.65e+01	8.18e+01	1.91e+01	7.65e+01	2.35e+01
	HHO	2.32e-48	8.39e-48	7.95e-48	4.25e-47	1.10e-49	1.59e-49	1.81e-48	6.64e-48
	IWOA	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
F_5	WOA	2.83e+01	4.03e-01	9.83e+01	3.31e-02	2.97e+02	1.59e-01	4.96e+02	4.54e-01
	HHO	2.23e-03	1.88e-03	6.14e-02	5.27e-02	1.46e-01	1.94e-01	1.56e-01	2.34e-01
	IWOA	7.07e-11	5.81e-11	3.55e-11	2.38e-11	2.50e-09	4.93e-09	9.11e-10	3.76e-09
F_6	WOA	4.08e-01	2.63e-01	4.58e+00	1.65e+00	1.62e+01	3.00e+00	3.23e+01	9.44e+00
	HHO	1.94e-04	3.82e-04	6.16e-04	7.87e-04	1.71e-03	1.95e-03	3.53e-03	6.12e-03
	IWOA	2.34e-13	5.00e-13	9.68e-14	1.64e-13	1.90e-14	3.03e-14	1.04e-12	2.87e-12
F_7	WOA	2.66e-03	1.87e-03	3.00e-03	3.62e-03	5.35e-03	5.80e-03	3.00e-03	7.12e-03
	HHO	1.37e-04	9.54e-05	1.01e-04	6.68e-05	1.39e-04	8.31e-05	1.16e-04	7.74e-05
	IWOA	1.95e-04	1.51e-04	3.39e-04	4.07e-04	1.66e-04	9.87e-05	3.95e-04	2.85e-04
F_8	WOA	-8.90e+03	1.78e+01	-2.59e+04	3.69e+03	-1.04e+05	1.41e+04	-1.87e+05	2.79e+04
	HHO	-1.26e+04	1.11e+00	-4.19e+04	2.51e+00	-1.26e+05	4.51e+00	-2.09e+05	1.16e+01
	IWOA	-1.26e+04	7.15e-08	-4.19e+04	4.03e-06	-1.26e+05	1.57e-04	-2.09e+05	7.62e-04
F_9	WOA	1.89e-15	1.02e-14	0.00e+00	0.00e+00	4.55e-14	1.36e-13	9.09e-14	2.73e-13
	HHO	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
	IWOA	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
F_{10}	WOA	4.20e-15	2.58e-15	4.91e-15	3.14e-15	4.09e-15	1.91e-15	4.80e-15	2.49e-15
	HHO	8.88e-16	0.00e+00	8.88e-16	0.00e+00	8.88e-16	0.00e+00	8.88e-16	0.00e+00
	IWOA	8.88e-16	0.00e+00	8.88e-16	0.00e+00	8.88e-16	0.00e+00	8.88e-16	0.00e+00
F_{11}	WOA	1.10e-02	4.12e-02	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
	HHO	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
	IWOA	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
F_{12}	WOA	3.59e-02	3.88e-02	4.64e-02	1.72e-02	7.65e-02	3.99e-02	9.61e-02	4.72e-02
	HHO	1.00e-05	1.19e-05	3.51e-06	3.33e-06	4.90e-06	5.46e-06	1.03e-06	1.02e-06
	IWOA	3.87e-15	6.18e-15	4.35e-16	7.15e-16	3.74e-16	6.50e-16	2.76e-16	6.42e-16
F_{13}	WOA	4.47e-01	2.86e-01	2.89e+00	8.63e-01	1.22e+01	3.11e+00	1.82e+01	3.69e+00
	HHO	7.13e-05	1.37e-04	1.51e-04	2.64e-04	2.29e-04	1.69e-04	4.05e-04	4.84e-04
	IWOA	1.39e-13	3.58e-13	6.08e-13	1.93e-12	2.11e-13	3.69e-13	1.97e-13	4.40e-13

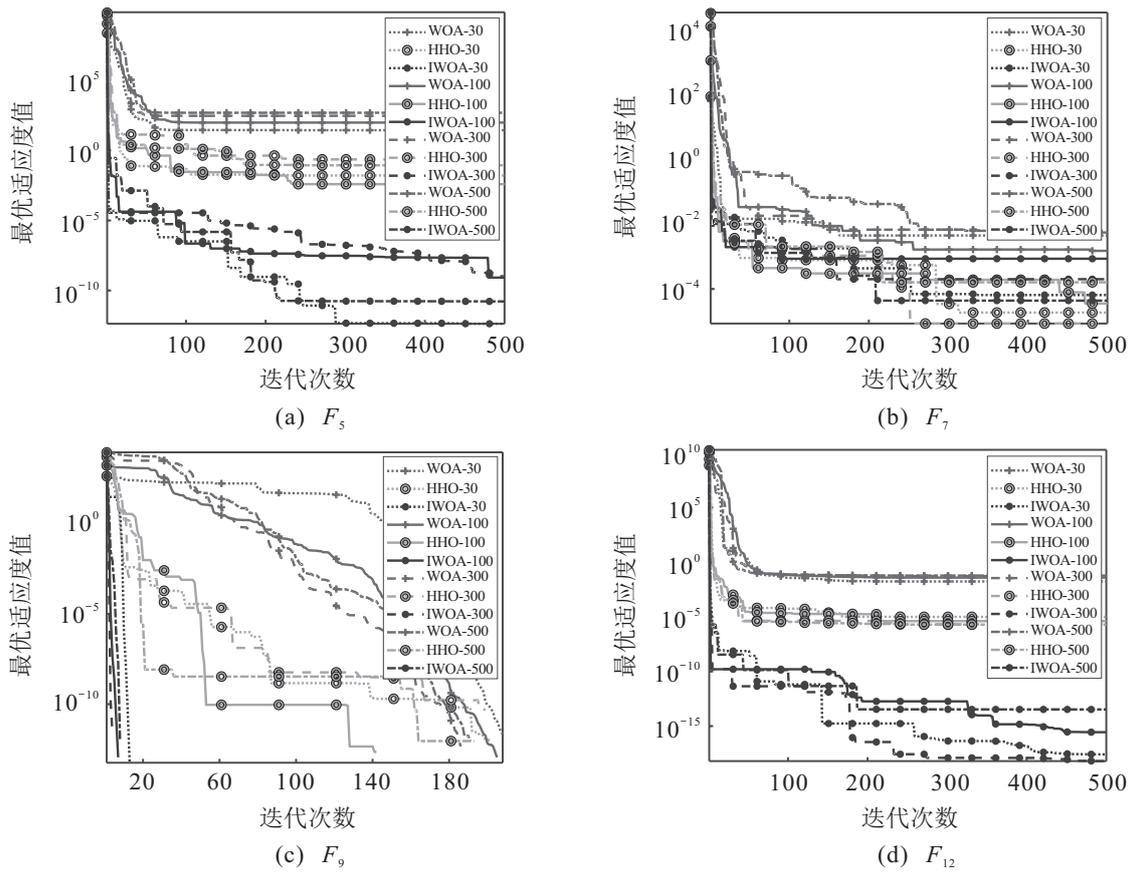


图2 不同维度下WOA、HHO和IWOA的收敛曲线

表3为使用Friedman检验和Wilcoxon检验对表2中数据进行分析的结果. 通过Friedman检验可知, 3种算法在不同维度的平均值和标准差数值大小排序结果均为IWOA < HHO < WOA, 表明IWOA对比原始WOA和HHO, 在4种不同维度中具有最好的整体寻优精度和稳定性; 通过Wilcoxon检验可知, 原始WOA和HHO与IWOA的渐近显著性均小于0.05, 表明IWOA与原始的WOA和HHO之间具有显著性差

距; 通过符号检验可知, 在不同维度上, IWOA取得的最优值均优于或等于WOA算法, 在 F_7 测试函数上IWOA算法的最优值低于HHO算法, 在3个测试函数上与HHO算法取得相同结果, 在9个测试函数上取得优于HHO算法的实验结果. 由上述3项检验结果可知, 在4种不同维度下IWOA算法仅在 F_7 测试函数上的测试结果劣于HHO算法. IWOA算法在13种测试函数的整体收敛精度与整体稳定性最优.

表3 WOA、HHO、IWOA算法对比检验

检验	WOA				HHO				IWOA			
	30D	100D	300D	500D	30D	100D	300D	500D	30D	100D	300D	500D
Friedman mean	2.92	2.85	2.92	2.92	1.88	1.88	1.85	1.85	1.19	1.27	1.23	1.23
Friedman std	2.92	2.77	2.85	2.92	1.88	1.96	1.92	1.85	1.19	1.27	1.23	1.23
Wilcoxon	1.47e-03	3.35e-03	2.22e-03	2.22e-03	2.84e-02	3.67e-02	2.84e-02	2.84e-02	-	-	-	-
+/-/=	13/0/0	11/0/2	12/0/1	12/0/1	9/1/3	9/1/3	9/1/3	9/1/3	-	-	-	-

4.2 与不同优化算法比较

选取4种近年流行且收敛效果优秀的优化算法: 灰狼优化算法(GWO)、飞蛾扑火优化算法(MFO)、蚁狮优化算法(ALO)、樽海鞘群算法(SSA), 在30维和100维的 $F_1 \sim F_{13}$ 可变维基准函数上, 与本文所提出的IWOA算法进行寻优对比.

图3包含了本文选取的4种不同优化算法与IWOA在不同维度下 F_3 、 F_7 、 F_{11} 、 F_{13} 的收敛曲线. 对比5种算法的收敛曲线可知, 在30维与100维的单峰、双峰基准测试函数中, 本文所改进的鲸鱼优化算法都有着较好的收敛精度和收敛速度.

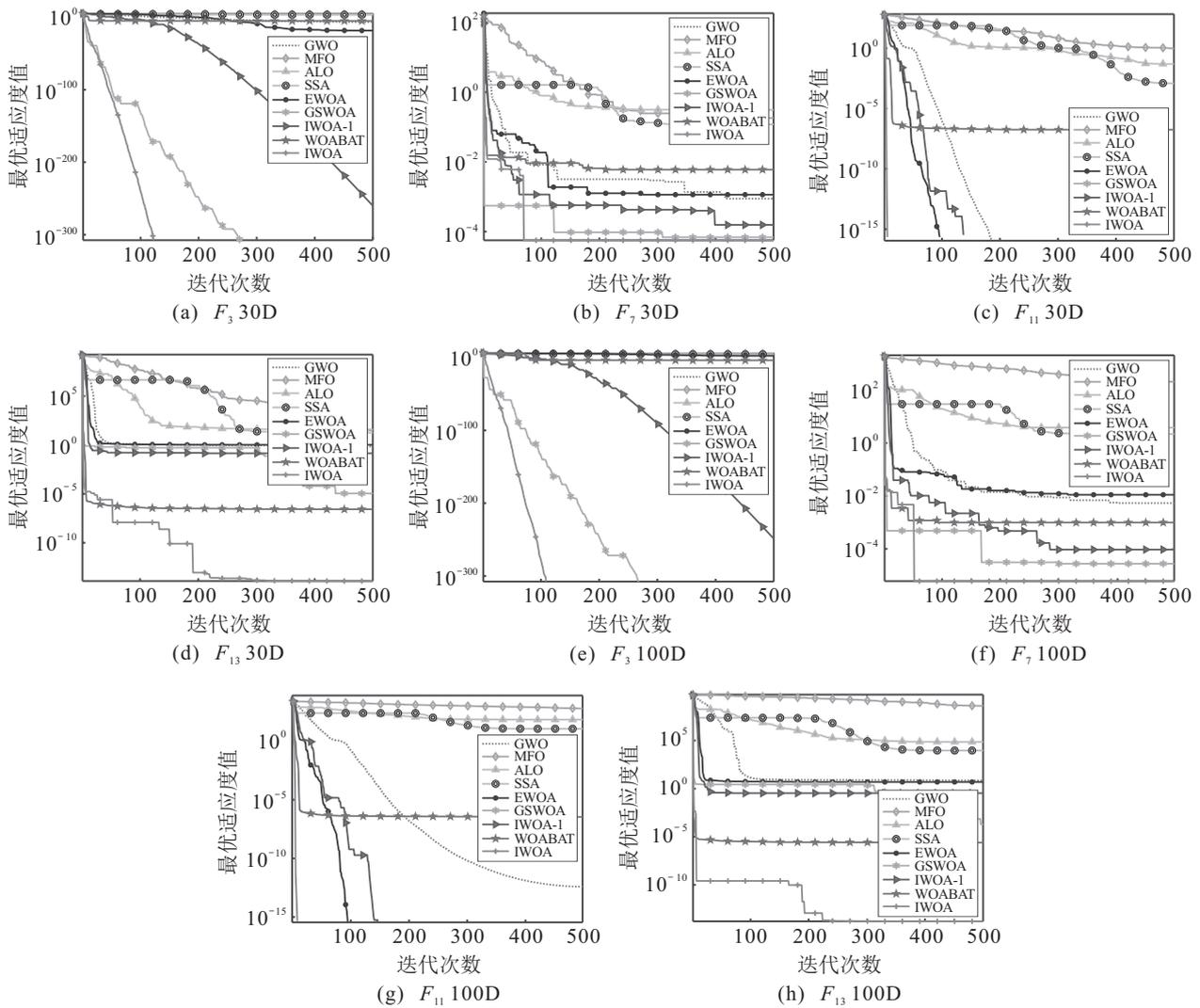


图3 多维度下不同改进优化算法的收敛曲线

表4为不同的优化算法在相同维度下,单独运行30次最优适应度值计算结果对比.分析对比实验数据可知:在30维与100维时,对所有不同算法的平均值和标准差进行Friedman检验,结果排序均为 $IWOA < GWO < SSA < ALO < MFO$;通过Wilcoxon检验可知,IWOA在2种不同维度上与4种优化算法的渐近显著性均小于0.05,表明IWOA与上述4种不同算法之间存在显著差异性;通过符号检验可知,在不同维度上,IWOA解的质量均优于上述4种算法.上述实验数据表明,本文所提出的改进鲸鱼算法的平均值、标准差均优于SSA、ALO、MFO、GWO算法,说明IWOA具有十分优秀的寻优精度与稳定性.

4.3 与不同改进策略的鲸鱼优化算法比较

将IWOA与文献[19-22]中4种优秀的改进算法进行对比,对比算法的参数按照原文献设置.由图3中不同维度下 F_3 、 F_7 、 F_{11} 、 F_{13} 的收敛曲线分析可知,IWOA具有优秀的收敛速度与精度.

表5为5种改进算法在30维、100维上,单独运行30次最优适应度值的平均值和标准差.由表5中Friedman检验结果可知,在不同维度上,IWOA的平均值和标准差检验结果均为最优值,表明IWOA算法在30维与100维中具有整体最优的收敛精度与稳定性;由Wilcoxon检验可知,IWOA与4种不同的改进优化算法之间存在显著差异性;通过符号检验分析,IWOA算法在不同维度的13种测试函数上的平均结果均优于或等于EWOA算法和WOABAT算法,在 F_7 测试函数上IWOA算法的平均最优值低于GSWOA算法和IWOA-1算法,结合收敛曲线可知,IWOA算法在 F_7 测试函数上存在寻得最优值的情况但平均最优值低于IWOA-1算法和GSWOA算法.综合上述3项检验结果,IWOA算法在30维与100维 $F_1 \sim F_{13}$ 测试函数上相较于EWOA、GSWOA、IWOA-1、WOABAT算法,整体上具有一定的优越性与稳定性.

表4 5种算法的寻优结果比较

函数	统计结果	GWO		MFO		ALO		SSA		IWOA	
		30	100	30	100	30	100	30	100	30	100
F_1	mean	5.83e-28	1.51e-12	1.00e+03	7.36e+04	1.24e-03	4.90e+03	3.95e-07	1.31e+03	0.00e+00	0.00e+00
	std	6.69e-28	1.05e-12	3.00e+03	1.60e+04	1.13e-03	1.88e+03	6.73e-07	3.02e+02	0.00e+00	0.00e+00
F_2	mean	1.09e-16	4.47e-08	3.01e+01	2.51e+02	6.64e+01	3.01e+02	2.14e+00	4.76e+01	0.00e+00	0.00e+00
	std	9.26e-17	1.43e-08	2.28e+01	2.84e+01	4.73e+01	1.48e+02	1.81e+00	6.44e+00	0.00e+00	0.00e+00
F_3	mean	3.83e-06	3.40e+02	2.62e+04	2.52e+05	3.78e+03	7.79e+04	1.82e+03	3.87e+04	0.00e+00	0.00e+00
	std	4.82e-06	2.17e+02	1.18e+04	5.04e+04	1.30e+03	2.29e+04	1.17e+03	1.61e+04	0.00e+00	0.00e+00
F_4	mean	1.03e-06	9.82e-01	6.74e+01	9.39e+01	1.74e+01	3.26e+01	1.17e+01	2.88e+01	0.00e+00	0.00e+00
	std	6.01e-07	7.22e-01	7.15e+00	1.58e+00	5.34e+00	3.05e+00	3.47e+00	3.60e+00	0.00e+00	0.00e+00
F_5	mean	2.67e+01	9.80e+01	1.02e+04	1.69e+08	1.07e+02	6.42e+05	5.30e+02	1.72e+05	7.07e-11	3.55e-11
	std	4.95e-01	4.63e-01	2.69e+04	8.73e+07	5.16e+01	6.19e+05	6.58e+02	6.76e+04	5.81e-11	2.38e-11
F_6	mean	6.61e-01	1.00e+01	4.02e+03	6.42e+04	9.19e-04	5.49e+03	1.07e-07	1.52e+03	2.34e-13	9.68e-14
	std	2.78e-01	1.01e+00	9.20e+03	1.19e+04	4.74e-04	2.72e+03	5.50e-08	4.03e+02	5.00e-13	1.64e-13
F_7	mean	2.20e-03	7.63e-03	8.10e-01	3.11e+02	2.53e-01	5.09e+00	1.43e-01	2.66e+00	1.95e-04	3.39e-04
	std	1.19e-03	2.30e-03	1.22e+00	1.11e+02	8.65e-02	1.84e+00	3.39e-02	6.92e-01	1.51e-04	4.07e-04
F_8	mean	-6.21e+03	-1.61e+04	-8.53e+03	-2.28e+04	-5.50e+03	-1.85e+04	-7.54e+03	-2.15e+04	-1.26e+04	-4.19e+04
	std	6.93e+02	2.78e+03	1.01e+03	2.22e+03	7.86e+01	1.55e+03	6.29e+02	1.51e+03	7.15e-08	4.03e-06
F_9	mean	5.52e+00	4.93e+00	1.66e+02	8.52e+02	7.09e+01	3.77e+02	4.69e+01	2.35e+02	0.00e+00	0.00e+00
	std	3.45e+00	3.71e+00	4.81e+01	5.87e+01	1.35e+01	4.69e+01	1.61e+01	5.00e+01	0.00e+00	0.00e+00
F_{10}	mean	9.93e-14	1.27e-07	1.54e+01	1.99e+01	5.66e+00	1.32e+01	2.49e+00	1.08e+01	8.88e-16	8.88e-16
	std	1.23e-14	2.95e-08	7.03e+00	4.40e-02	4.15e+00	1.34e+00	5.33e-01	1.32e+00	0.00e+00	0.00e+00
F_{11}	mean	3.51e-03	4.96e-03	9.41e-01	4.93e+02	6.20e-02	4.77e+01	1.74e-02	1.14e+01	0.00e+00	0.00e+00
	std	7.05e-03	9.99e-03	1.42e-01	1.02e+02	3.10e-02	2.48e+01	1.08e-02	2.36e+00	0.00e+00	0.00e+00
F_{12}	mean	3.92e-02	2.96e-01	2.59e+07	3.78e+08	1.35e+01	1.68e+03	7.17e+00	3.45e+01	3.87e-15	4.35e-16
	std	1.27e-02	7.88e-02	7.67e+07	1.72e+08	5.15e+00	3.25e+03	2.90e+00	1.00e+01	6.18e-15	7.15e-16
F_{13}	mean	6.40e-01	6.94e+00	8.20e+07	7.82e+08	2.79e+01	1.05e+05	2.41e+01	8.30e+03	1.39e-13	6.08e-13
	std	1.92e-01	6.00e-01	1.64e+08	3.07e+08	1.54e+01	7.59e+04	1.58e+01	1.74e+04	3.58e-13	1.93e-12
Friedmanmean		2.31	2.23	4.69	4.69	4.00	4.08	3.00	3.00	1.00	1.00
Friedman std		2.31	2.23	4.92	4.54	3.62	4.00	3.15	3.23	1.00	1.00
Wilcoxon		1.47e-03	-	-							
+/-/=		13/0/0	13/0/0	13/0/0	13/0/0	13/0/0	13/0/0	13/0/0	13/0/0	-	-

5 结论

针对传统鲸鱼优化算法的性能不足,本文提出了一种基于围攻机制的改进鲸鱼优化算法,在初始化鲸鱼种群、使用非线性参数、适应度更新控制的改进基础上,引进了哈里斯鹰优化算法的围攻机制与高斯检测机制.通过Friedman检验、Wilcoxon检验和符号检验对IWOA算法进行了分析:首先,对比了鲸鱼优化算法(WOA)、哈里斯鹰优化算法(HHO)、基于围攻机制的改进鲸鱼优化算法(IWOA),结果表明本文的改

进算法与原始算法具有显著差异性;其次,对比了灰狼优化算法(GWO)、飞蛾扑火优化算法(MFO)、蚁狮优化算法(ALO)、樽海鞘群算法(SSA),以及不同的改进WOA算法EWOA、GSWOA、IWOA-1、WOABAT,验证了IWOA在13个不同基准函数上拥有整体最优的稳定性和收敛精度.3组实验结果表明,IWOA相较于原始鲸鱼优化算法具有明显的优化效果,在拥有较好的稳定性的同时保证了收敛精度和速度,体现了改进算法的有效性.

表 5 不同改进优化算法的寻优结果比较

函数	统计结果	EWOA		GSGWOA		IWOA-1		WOABAT		IWOA	
		30	100	30	100	30	100	30	100	30	100
F_1	mean	1.44e-140	8.31e-140	0.00e+00	0.00e+00	6.25e-276	3.96e-277	1.42e-06	2.18e-05	0.00e+00	0.00e+00
	std	4.31e-140	2.49e-139	0.00e+00	0.00e+00	0.00e+00	0.00e+00	5.02e-07	5.47e-06	0.00e+00	0.00e+00
F_2	mean	8.63e-82	9.31e-81	6.51e-307	9.34e-295	7.03e-140	1.61e-141	8.35e-03	4.53e-02	0.00e+00	0.00e+00
	std	2.59e-81	2.78e-80	0.00e+00	0.00e+00	2.11e-139	4.67e-141	1.81e-03	4.26e-03	0.00e+00	0.00e+00
F_3	mean	1.13e-03	1.83e+01	0.00e+00	0.00e+00	8.94e-252	3.52e-247	1.02e+00	1.25e-04	0.00e+00	0.00e+00
	std	3.38e-03	3.60e+01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	3.07e+00	1.47e-05	0.00e+00	0.00e+00
F_4	mean	1.08e-42	4.28e-46	4.57e-304	7.54e-302	1.07e-131	8.13e-131	1.01e-03	1.52e-03	0.00e+00	0.00e+00
	std	3.24e-42	9.68e-46	0.00e+00	0.00e+00	3.10e-131	2.37e-130	9.42e-05	1.83e-04	0.00e+00	0.00e+00
F_5	mean	2.79e+01	9.81e+01	3.46e-02	9.55e-02	2.81e+01	9.80e+01	8.49e+00	1.95e+01	7.07e-11	3.55e-11
	std	5.63e-01	3.64e-01	5.01e-02	1.07e-01	2.23e-01	8.38e-02	1.30e+01	3.91e+01	5.81e-11	2.38e-11
F_6	mean	8.45e-01	7.52e+00	3.26e-04	8.09e-04	2.43e-01	1.12e+00	1.84e-06	2.24e-05	2.34e-13	9.68e-14
	std	3.86e-01	1.58e+00	4.37e-04	9.59e-04	8.50e-02	2.20e-01	8.71e-07	6.25e-06	5.00e-13	1.64e-13
F_7	mean	2.55e-03	3.00e-03	1.16e-04	6.73e-05	8.94e-05	8.31e-05	6.74e-04	7.07e-04	1.95e-04	3.39e-04
	std	3.24e-03	2.50e-03	1.12e-04	7.87e-05	6.31e-05	5.81e-05	9.18e-04	6.95e-04	1.51e-04	4.07e-04
F_8	mean	-7.85e+03	-2.49e+04	-1.22e+04	-4.14e+04	-1.24e+04	-4.18e+04	-1.22e+04	-3.95e+04	-1.26e+04	-4.19e+04
	std	9.09e+02	4.30e+03	6.11e+02	9.94e+02	3.24e+02	9.05e+01	1.07e+03	7.15e+03	7.15e-08	4.03e-06
F_9	mean	1.15e+01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	2.99e+00	9.95e+00	0.00e+00	0.00e+00
	std	3.46e+01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	8.95e+00	2.98e+01	0.00e+00	0.00e+00
F_{10}	mean	3.02e-15	3.73e-15	8.88e-16	8.88e-16	8.88e-16	8.88e-16	9.30e-04	1.98e-03	8.88e-16	8.88e-16
	std	1.74e-15	1.42e-15	0.00e+00	0.00e+00	0.00e+00	0.00e+00	2.29e-04	1.86e-04	0.00e+00	0.00e+00
F_{11}	mean	6.29e-03	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	6.68e-08	3.30e-07	0.00e+00	0.00e+00
	std	1.89e-02	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00	2.50e-08	4.74e-08	0.00e+00	0.00e+00
F_{12}	mean	5.97e-02	1.13e-01	1.71e-05	2.28e-06	7.65e-03	7.38e-03	1.86e-08	4.69e-08	3.87e-15	4.35e-16
	std	3.17e-02	3.71e-02	3.55e-05	2.03e-06	2.74e-03	3.81e-03	6.30e-09	8.11e-09	6.18e-15	7.15e-16
F_{13}	mean	8.27e-01	4.45e+00	1.71e-04	3.06e-04	1.11e-01	4.64e-01	1.68e-07	2.68e-06	1.39e-13	6.08e-13
	std	2.93e-01	1.58e+00	2.21e-04	3.69e-04	2.75e-02	1.30e-01	6.82e-08	7.78e-07	3.58e-13	1.93e-12
Friedman mean		4.54	4.31	2.31	2.23	2.92	3.00	3.77	3.92	1.46	1.54
Friedman std		4.46	4.15	2.23	2.38	2.62	2.62	4.08	4.15	1.62	1.69
Wilcoxon		1.47e-03	3.35e-03	4.99e-02	4.99e-02	2.18e-02	2.18e-02	1.47e-03	1.47e-03	-	-
+/-/=		13/0/0	11/0/2	7/1/5	7/1/5	9/1/3	9/1/3	13/0/0	13/0/0	-	-

参考文献(References)

[1] Mirjalili S, Lewis A. The whale optimization algorithm[J]. Advances in Engineering Software, 2016, 95: 51-67.

[2] 王坚浩, 张亮, 史超, 等. 基于混沌搜索策略的鲸鱼优化算法[J]. 控制与决策, 2019, 34(9): 1893-1900. (Wang J H, Zhang L, Shi C, et al. Whale optimization algorithm based on chaotic search strategy[J]. Control and Decision, 2019, 34(9): 1893-1900.)

[3] Heidari A A, Mirjalili S, Faris H, et al. Harris hawks optimization: Algorithm and applications[J]. Future Generation Computer Systems, 2019, 97: 849-872.

[4] 刘小龙, 梁彤纓. 基于方形邻域和随机数组的哈里斯

鹰优化算法[J]. 控制与决策, 2022, 37(10): 2467-2476. (Liu X L, Liang T Y. Harris hawk optimization algorithm based on square neighborhood and random array[J]. Control and Decision, 2022, 37(10): 2467-2476.)

[5] Sun Y J, Wang X L, Chen Y H, et al. A modified whale optimization algorithm for large-scale global optimization problems[J]. Expert Systems with Applications, 2018, 114: 563-577.

[6] 林杰, 何庆, 王茜, 等. 基于混沌的正余弦鲸鱼优化算法[J]. 智能计算机与应用, 2020, 10(9): 43-48. (Lin J, He Q, Wang Q, et al. Chaos-based sine-cosine whale optimization algorithm[J]. Intelligent Computer and Applications, 2020, 10(9): 43-48.)

[7] 黄清宝, 李俊兴, 宋春宁, 等. 基于余弦控制因子和

- 多项式变异的鲸鱼优化算法[J]. 控制与决策, 2020, 35(3): 559-568.
(Huang Q B, Li J X, Song C N, et al. Whale optimization algorithm based on cosine control factor and polynomial mutation[J]. Control and Decision, 2020, 35(3): 559-568.)
- [8] 兰娅勋. 混沌和柯西变异的蝗虫优化算法及特征选择[J]. 微电子学与计算机, 2021, 38(11): 21-30.
(Lan Y X. Grasshopper optimization algorithm based on chaos and cauchy mutation and feature selection[J]. Microelectronics & Computer, 2021, 38(11): 21-30.)
- [9] Song S M, Wang P J, Heidari A A, et al. Dimension decided Harris Hawks optimization with Gaussian mutation: Balance analysis and diversity patterns[J]. Knowledge-Based Systems, 2021, 215: 106425.
- [10] 张铸, 饶盛华, 张仕杰. 基于自适应正态云模型的灰狼优化算法[J]. 控制与决策, 2021, 36(10): 2562-2568.
(Zhang Z, Rao S H, Zhang S J. Grey wolf optimization algorithm based on adaptive normal cloud model[J]. Control and Decision, 2021, 36(10): 2562-2568.)
- [11] 龙文, 蔡绍洪, 焦建军, 等. 求解大规模优化问题的改进鲸鱼优化算法[J]. 系统工程理论与实践, 2017, 37(11): 2983-2994.
(Long W, Cai S H, Jiao J J, et al. Improved whale optimization algorithm for large scale optimization problems[J]. Systems Engineering — Theory & Practice, 2017, 37(11): 2983-2994.)
- [12] Tharwat A, Hassanien A E. Chaotic antlion algorithm for parameter optimization of support vector machine[J]. Applied Intelligence, 2018, 48(3): 670-686.
- [13] 魏政磊, 赵辉, 李牧东, 等. 控制参数值非线性调整策略的灰狼优化算法[J]. 空军工程大学学报: 自然科学版, 2016, 17(3): 68-72.
(Wei Z L, Zhao H, Li M D, et al. A grey wolf optimization algorithm based on nonlinear adjustment strategy of control parameter[J]. Journal of Air Force Engineering University: Natural Science Edition, 2016, 17(3): 68-72.)
- [14] 张霓, 曾乐襄, 何熊熊, 等. 基于滚动时域粒子群优化的视频去雾算法[J]. 控制与决策, 2021, 36(9): 2218-2224.
(Zhang N, Zeng L X, He X X, et al. Receding horizon particle swarm optimization based video defogging algorithm[J]. Control and Decision, 2021, 36(9): 2218-2224.)
- [15] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.
- [16] Mirjalili S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm[J]. Knowledge-Based Systems, 2015, 89: 228-249.
- [17] Mirjalili S. The ant lion optimizer[J]. Advances in Engineering Software, 2015, 83: 80-98.
- [18] 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.
- [19] 冯文涛, 宋科康. 一种增强型鲸鱼优化算法[J]. 计算机仿真, 2020, 37(11): 275-279.
(Feng W T, Song K K. An enhanced whale optimization algorithm[J]. Computer Simulation, 2020, 37(11): 275-279.)
- [20] 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.
(Liu L, Bai K Q, Dan Z H, et al. Whale optimization algorithm with global search strategy[J]. Journal of Chinese Computer Systems, 2020, 41(9): 1820-1825.)
- [21] 赵晶, 祝锡晶, 孟小玲, 等. 改进鲸鱼优化算法在机械臂时间最优轨迹规划的应用[J]. 机械科学与技术, DOI: 10.13433/j.cnki.1003-8728.20200596.
(Zhao J, Zhu X J, Meng X L, et al. Application of improved whale optimization algorithm in time-optimal trajectory planning of manipulator[J]. Mechanical Science and Technology for Aerospace Engineering, DOI: 10.13433/j.cnki.1003-8728.20200596.)
- [22] Mohammed H M, Umar S U, Rashid T A. A systematic and meta-analysis survey of whale optimization algorithm[J]. Computational Intelligence and Neuroscience, 2019, 2019: 1-25.
- [23] 于建芳, 刘升, 韩斐斐, 等. 基于柯西变异的蚁狮优化算法[J]. 微电子学与计算机, 2019, 36(6): 45-49.
(Yu J F, Liu S, Han F F, et al. Ant lion optimization algorithm based on cauchy variation[J]. Microelectronics & Computer, 2019, 36(6): 45-49.)

作者简介

王延峰(1973—), 男, 教授, 博士, 从事生物启发计算、生物信息处理、复杂系统建模等研究, E-mail: yanfengwang@yeah.net;

廖荣航(1997—), 男, 硕士生, 从事群智能的研究, E-mail: ronghangliao@yeah.net;

梁恩豪(1995—), 男, 硕士, 从事大数据的研究, E-mail: enhaoliang_edu@163.com;

孙军伟(1984—), 男, 副教授, 博士, 从事智能算法、数据处理等研究, E-mail: junweisun@yeah.net.