



融合非线性收敛因子与变异准反射学习的哈里斯鹰优化算法

宋美佳, 贾鹤鸣, 林志兴, 刘庆鑫

引用本文:

宋美佳, 贾鹤鸣, 林志兴, 刘庆鑫. 融合非线性收敛因子与变异准反射学习的哈里斯鹰优化算法[J]. 智能系统学报, 2024, 19(3): 738-748.

SONG Meijia, JIA Heming, LIN Zhixing, et al. Harris Hawks optimization algorithm based on nonlinear convergence factor and mutation quasi-reflected-based learning[J]. *CAA Transactions on Intelligent Systems*, 2024, 19(3): 738-748.

在线阅读 View online: <https://dx.doi.org/10.11992/tis.202205008>

您可能感兴趣的其他文章

具有Levy变异和精英自适应竞争机制的蚁狮优化算法

Ant lion optimizer with levy variation and adaptive elite competition mechanism
智能系统学报. 2018, 13(2): 236-242 <https://dx.doi.org/10.11992/tis.201706091>

求解离散优化问题的元胞量子狼群演化算法

Cellular and quantum-behaved wolf pack evolutionary algorithm for solving discrete optimization problems
智能系统学报. 2018, 13(5): 716-727 <https://dx.doi.org/10.11992/tis.201705007>

基于混沌搜索和权重学习的教与学优化算法及其应用

Teaching-learning-based optimization algorithm based on chaotic search and weighted learning and its application
智能系统学报. 2018, 13(5): 818-828 <https://dx.doi.org/10.11992/tis.201705017>

一种精英反向学习的萤火虫优化算法

Firefly optimization algorithm utilizing elite opposition-based learning
智能系统学报. 2017, 12(5): 710-716 <https://dx.doi.org/10.11992/tis.201706014>

一种增强局部搜索能力的改进人工蜂群算法

Improved artificial bee colony algorithm based on enhanced local search
智能系统学报. 2017, 12(5): 684-693 <https://dx.doi.org/10.11992/tis.201612026>

依概率收敛的改进粒子群优化算法

Improved particle swarm optimization algorithm with probability convergence
智能系统学报. 2017, 12(4): 511-518 <https://dx.doi.org/10.11992/tis.201610004>

DOI: 10.11992/tis.202205008

网络出版地址: <https://link.cnki.net/urlid/23.1538.TP.20231018.0843.002>

融合非线性收敛因子与变异准反射学习的 哈里斯鹰优化算法

宋美佳¹, 贾鹤鸣², 林志兴¹, 刘庆鑫³

(1. 三明学院 网络中心, 福建 三明 365004; 2. 三明学院 信息工程学院, 福建 三明 365004; 3. 海南大学 计算机科学与技术学院, 海南 海口 570228)

摘要: 针对哈里斯鹰优化算法 (Harris Hawks optimization, HHO) 易早熟收敛、寻优精度低、收敛速度慢等问题, 提出一种融合非线性收敛因子与变异准反射学习的哈里斯鹰优化算法 (improved Harris Hawks optimization, IH-HO)。首先, 初始化阶段引入 Circle 混沌映射, 提高初始化种群多样性和种群位置质量; 其次, 引入 Sigmoid 非线性收敛因子, 平衡全局探索和局部开发能力; 最后, 针对 HHO 算法易陷入局部最优问题, 提出变异准反射学习 (quasi-reflection-based learning, QRBL) 策略, 提高种群活力, 进一步提高算法局部收敛能力。仿真实验采用 13 个标准测试函数和 1 个经典工程问题对改进算法进行测试, 结果表明改进算法收敛精度、收敛速度均有较大提高, 适用于解决实际问题。

关键词: 哈里斯鹰优化算法; 非线性收敛因子; 准反射学习; 准反向学习; 混沌映射; 工程问题; 元启发算法; 群智能

中图分类号: TP301.6 文献标志码: A 文章编号: 1673-4785(2024)03-0738-11

中文引用格式: 宋美佳, 贾鹤鸣, 林志兴, 等. 融合非线性收敛因子与变异准反射学习的哈里斯鹰优化算法 [J]. 智能系统学报, 2024, 19(3): 738-748.

英文引用格式: SONG Meijia, JIA Heming, LIN Zhixing, et al. Harris Hawks optimization algorithm based on nonlinear convergence factor and mutation quasi-reflected-based learning[J]. CAAI transactions on intelligent systems, 2024, 19(3): 738-748.

Harris Hawks optimization algorithm based on nonlinear convergence factor and mutation quasi-reflected-based learning

SONG Meijia¹, JIA Heming², LIN Zhixing¹, LIU Qingxin³

(1. Center of Network, Sanming University, Sanming 365004, China; 2. Department of Information Engineering, Sanming University, Sanming 365004, China; 3. School of Computer Science and Technology, Hainan University, Haikou 570228, China)

Abstract: Due to the shortcomings of the Harris Hawks optimization (HHO) algorithm, such as premature convergence, low optimization precision, and slow convergence speed, an improved HHO (IH-HO) algorithm integrating nonlinear convergence factor and mutation quasi-reflection-based learning (QRBL) is proposed. First, circle chaotic mapping is introduced in the initialization stage to improve the diversity of the initialization population and the location and quality of the population. Second, the sigmoid nonlinear convergence factor is introduced to balance the ability of global exploration and partial exploitation. Finally, because the HHO algorithm easily falls into the local optimum, mutation QRBL is proposed to improve the vigor of the population and further improve the local convergence ability of the algorithm. The simulation experiments are conducted by applying 13 standard test functions and one classical engineering problem to the evaluation of the proposed algorithm. The results show that the convergence accuracy and the convergence speed of the IH-HO algorithm are greatly improved, and IH-HO is suitable for solving practical problems.

Keywords: Harris Hawks optimization; nonlinear convergence factor; quasi-reflected-based learning; quasi-inverse learning; chaotic mapping; engineering problems; meta-heuristic algorithms; swarm intelligence

收稿日期: 2022-05-12. 网络出版日期: 2023-10-18.

基金项目: 福建省教育厅中青年教师教育科研项目 (JAT211002, JAT210432, JAT220834); 福建省高校教育技术研究会 2022 年度课题 (FJET202211, FJET202206); 福建省电子商务工程中心开放课题 (KBX2109); 国家教育部教师函〔2021〕13 号第二批人工智能助推教师队伍建设试点项目 (三明学院); 福建省自然科学基金面上项目 (2021J011128).

通信作者: 贾鹤鸣. E-mail: jiaheminglucky99@126.com.

随着科技的不断创新发展, 不同领域产生诸多优化问题。近年来, 众多学者通过分析生物种群或物理现象而提出的群智能优化算法不断涌现, 如 Kennedy 等^[1] 受到鸟群觅食行为的规律性启发提出了粒子群优化算法 (particle swarm optim-

ization, PSO), PSO 具有收敛速度快、参数少且算法简单易实现的优点。Mirjalili 等^[2]根据灰狼种群的等级制度和狩猎机制提出了灰狼优化算法 (grey wolf optimization, GWO), GWO 通过 4 种类型的灰狼模拟领导阶层,并实现了狩猎的 3 个主要步骤:寻找猎物、包围猎物和攻击猎物。Mirjalili 等^[3]通过模拟鲸鱼随机搜索捕食、气泡攻击与收缩包围行为提出的鲸鱼优化算法 (whale optimization algorithm, WOA)。Mirjalili 等^[4]模拟了樽海鞘链的群体行为,通过领导者指导追随者,提出一种模拟链式行为向事物移动的樽海鞘群优化算法 (salp swarm algorithm, SSA)。Mirjalili^[5]为优化飞机机翼设计问题而提出的正余弦优化算法,主要基于数学中正余弦函数图像的概念,使最佳解向内或向外扩展,对应全局搜索和局部开发两个阶段,对最佳位置进行深度挖掘,最终达到全局最优的目的。Li 等^[6]基于黏菌的扩散和觅食行为提出黏菌优化算法 (slime mould algorithm, SMA), SMA 主要模拟了黏菌在觅食过程中的行为和形态变化,通过权值指标模拟黏菌静脉状管的形态变化和收缩模式之间的 3 种相关性。贾鹤鸣等^[7]将遗传算法和乌燕鸥算法混合,应用到数据分类领域,并通过 16 组 UCI 数据和乳腺癌数据集验证算法在数据挖掘领域的应用。

2019 年, Heidari 等^[8]提出了哈里斯鹰优化算法 (Harris Hawks optimization, HHO), HHO 是受自然界哈里斯鹰捕食行为启发的一种种群智能优化算法,包含搜索、搜索与开发的转换、开发 3 个阶段。HHO 是一种基于种群、无梯度的优化技术,具有较强的全局搜索能力,由于其简单易行、参数较少且寻优能力较强等特点,因此在图像分割^[9]、神经网络训练^[10]、电机控制^[11]等领域广泛运用。但是 HHO 与其他种群智能优化算法一样,在求解复杂优化问题时,存在收敛速度慢、寻优精度低、易陷入局部最优等缺陷。因此,很多学者针对 HHO 的不足进行改进研究。马一鸣等^[12]利用 Chan 算法计算初始解,并替换单个个体位置,以减少不必要的全局搜索,提高了算法的收敛速度。Houssein 等^[13]引入交叉和变异合作基因操作符,并基于反向学习和随机反向学习策略提出 HHOCM (Harris Hawks optimization via crossover and mutation) 优化算法,有效提高解决方案的开发能力和种群的多样性。汤安迪等^[14]引入 Tent 混沌映射、精英等级制度、非线性逃逸能量更新策略和高斯随机游走策略,以提升算法的收敛速度和收敛精度。Jia 等^[15]引入动态控制参数策略和突变机

理,应用于 HHO 逃逸能量的计算和全局探索,通过调节不同参数,取得较好效果。上述改进策略对 HHO 算法的性能有一定提升,但对于 HHO 算法收敛速度慢、精度低、易陷入局部最优值的问题,依然具有较大的改进空间。

本文基于 HHO 算法存在的不足,首先,引入 Circle 混沌映射,提升初始化种群位置质量,其次引入 Sigmoid 非线性收敛因子,平衡全局探索和局部开发的转换,最后提出一种变异准反射学习策略,根据变异因子对求解进行变异和转换,并根据贪心策略保留最优解,有效增强了算法的寻优性能,并提升收敛速度和收敛精度。仿真实验部分采用 13 个标准测试函数以及 1 个经典工程问题对算法进行测试,结果表明改进算法寻优性能更佳,对跳出局部最优能力有一定增强。

1 哈里斯鹰优化算法

1.1 探索阶段

HHO 算法的灵感来自于哈里斯鹰探索、攻击猎物的不同攻击策略,主要由探索阶段、探索与开发的转换和开发阶段 3 部分组成。

在探索阶段,哈里斯鹰随机栖息在某些地方,它们栖息的地方基于其他成员或猎物的位置,具体计算方式为

$$X(t+1) = \begin{cases} X_{\text{rand}}(t) - r_1 |X_{\text{rand}}(t) - 2r_2 X(t)|, & q \geq 0.5 \\ (X_{\text{rabbit}}(t) - X_m(t)) - r_3 (L_B + r_4 (U_B - L_B)), & q < 0.5 \end{cases} \quad (1)$$

$$X_m(t) = \frac{1}{N} \sum_{i=1}^N X_i(t) \quad (2)$$

式中: $X(t+1)$ 是哈里斯鹰下一次迭代位置, $X_{\text{rabbit}}(t)$ 是猎物位置, r_1 、 r_2 、 r_3 、 r_4 和 q 是 (0,1) 间的随机数, U_B 和 L_B 为变化范围的上下界, $X_{\text{rand}}(t)$ 为当前种群中随机选择个体, $X_m(t)$ 是当前鹰群的平均位置。

1.2 探索到开发的转换

从全局搜索到局部开发的转换主要基于猎物的逃逸能量 E , 当 $|E| \geq 1$ 时算法进入探索阶段, 当 $|E| < 1$ 时算法进入开发阶段, E 的计算公式为

$$E_1 = 2 \left(1 - \frac{t}{T} \right) \quad (3)$$

$$E = E_0 E_1 \quad (4)$$

式中: E_0 为初始能量, 为 -1 到 1 的随机数; t 为当前迭代次数; T 为最大迭代次数。

1.3 开发阶段

在开发阶段, 哈里斯鹰基于 4 种追捕策略进行狩猎。当 $r \geq 0.5$ 且 $|E| \geq 0.5$ 时, 表示猎物有足够的逃逸能量但是逃跑未成功, 此时哈里斯鹰通过

软包围的方式进行狩猎:

$$X(t+1) = \Delta X(t) - E|JX_{\text{rabbit}}(t) - X(t)| \quad (5)$$

$$\Delta X(t) = X_{\text{rabbit}}(t) - X(t) \quad (6)$$

$$J = 2(1 - r_5) \quad (7)$$

式中: $\Delta X(t)$ 是猎物位置和当前位置的差值, r_5 是 0~1 的随机数, J 表示猎物逃跑过程中的跳跃距离。

当 $r \geq 0.5$ 且 $|E| < 0.5$ 时, 表示猎物没有足够的逃逸能量, 也没有逃脱机会, 此时哈里斯鹰通过硬包围的方式进行狩猎:

$$X(t+1) = X_{\text{rabbit}}(t) - E|\Delta X(t)| \quad (8)$$

当 $r < 0.5$ 且 $|E| \geq 0.5$ 时, 表示猎物有足够的逃逸能量且成功逃脱, 此时哈里斯鹰通过更智能的软包围方式进行狩猎, 主要通过 2 个策略, 首先执行:

$$Y = X_{\text{rabbit}}(t) - E|JX_{\text{rabbit}}(t) - X(t)| \quad (9)$$

当式(9)的策略无效时, 执行如下策略:

$$Z = Y + S \times L_F(D) \quad (10)$$

式中: D 是问题维度, S 是一个 $1 \times D$ 维的随机向量, L_F 为 Levy 飞行函数:

$$L_F(x) = 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}} \quad (11)$$

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin \frac{\pi\beta}{2}}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\frac{\beta-1}{2}}} \right)^{\frac{1}{\beta}} \quad (12)$$

其中: u 、 v 是 $1 \times D$ 的随机正态分布向量, β 是取值为 1.5 的常数, 因此该阶段的更新策略最终为

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ Z, & F(Z) < F(X(t)) \end{cases} \quad (13)$$

当 $r < 0.5$ 且 $|E| < 0.5$ 时, 表示猎物有机会逃脱, 但没有足够的逃逸能量, 此时哈里斯鹰在突袭前形成硬包围以缩小它们和猎物的距离:

$$X(t+1) = \begin{cases} Y, & F(Y) < F(X(t)) \\ Z, & F(Z) < F(X(t)) \end{cases} \quad (14)$$

$$Y = X_{\text{rabbit}}(t) - E|JX_{\text{rabbit}}(t) - X_m(t)| \quad (15)$$

$$Z = Y + S \times L_F(D) \quad (16)$$

2 改进的哈里斯鹰优化算法

2.1 Circle 混沌映射

混沌映射是一种由简单的确定性系统产生的随机性序列, 具有一定的便利性和很高的随机性, 可以产生 (0,1) 分布较为均匀的随机数。经过实验证明^[16], 利用混沌序列进行种群初始化、选择、交叉和变异等操作会影响算法的整个过程, 而且常常能取得比伪随机数更好的效果。产生混

沌序列的映射有很多, Circle 混沌映射是常用的映射之一, 本文引入 Circle 混沌映射^[16] 初始化种群, 以保证种群初始位置的质量:

$$X_{i+1} = \text{mod} \left(X_i + b - \left(\frac{a}{2\pi} \sin 2\pi X_i \right), 1 \right) \quad (17)$$

式中: a 为 0.5, b 为 0.2。

2.2 Sigmoid 非线性收敛因子

原算法的逃逸能量是探索到开发转换的关键性因素, 主要通过式(3)、(4) 计算, 式(3) 产生从 2 到 0 线性下降的值, 虽然可以在一定程度上支持探索与开发的转换, 但在迭代次数超过最大迭代次数一半的时候, 就一定进入局部开发过程。然而实际的搜索过程是非常复杂的非线性过程, 确定的转换边界容易降低算法寻优的性能。为平衡全局搜索和局部搜索, 引入 Sigmoid 非线性收敛因子^[17]。引入后, 式(3) 更新为

$$E_{1_new} = \frac{w}{1 + \exp \left(\frac{t - T/2}{T} \times 10 \right)} \quad (18)$$

式中: w 为调节因子, 本文选取 $w=2$ 。Sigmoid 非线性收敛因子和原文收敛因子对比如图 1 所示。

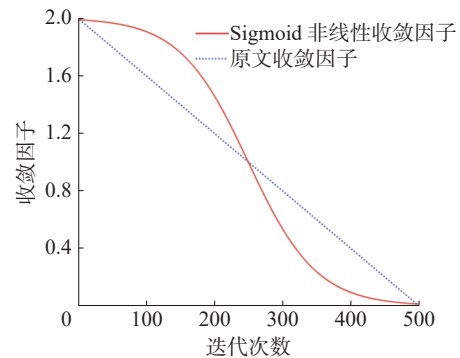
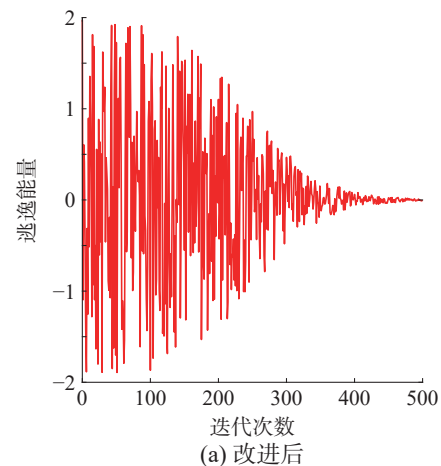


图 1 收敛因子曲线对比

Fig. 1 Curve of convergence factor comparison

更新后逃逸能量对比如图 2 所示。前期种群保持个体较大自由度, 增强全局寻优能力; 后期自由度快速下降, 增加局部开发能力。



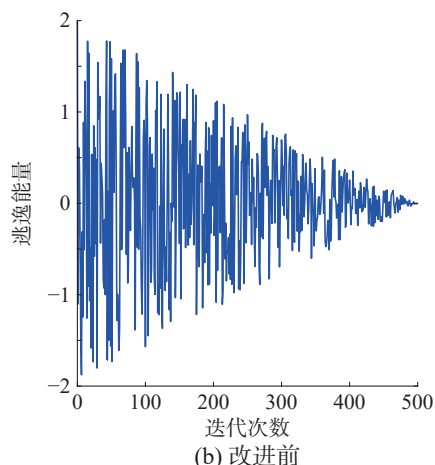


图 2 逃逸能量曲线对比

Fig. 2 Curve of escape energy comparison

2.3 变异准反射学习策略

HHO 算法中, 哈里斯鹰通过 4 种追捕策略更新自身位置, 重新计算适应度并择优进入下一次迭代, 没有对更新位置进行干扰, 容易导致算法陷入局部最优。据此, 本文提出变异准反射学习策略, 增加反射解和反向解对位置的干扰, 增加种群多样性, 提高算法跳出局部最优能力。

反向学习策略 (opposition-based learning, OBL)^[18-19] 主要思想是生成可行解的反向解, 通过评估可行解和反向解, 选择更好的候选解, 具体公式为

$$X_o(t) = L_B + U_B - X(t) \quad (19)$$

OBL 的一种变体为准反向学习 (quasi-opposition-based learning, QOBL)^[20], 以往的研究证明, 在寻找全局最优解时, 使用准反向解比反向解更有效。准反向解的具体表达式为

$$X_{qo}(t) = \text{rand}\left(\frac{L_B + U_B}{2}, X_o(t)\right) \quad (20)$$

基于 OBL 和 QOBL, Ewees 等^[21] 提出了一种新的准反射学习机制 (QRBL), 准反射解是根据准反向解获得的, 具体计算为

$$X_{qr}(t) = \text{rand}\left(\frac{L_B + U_B}{2}, X(t)\right) \quad (21)$$

三者的主要区别, 如图 3 所示。

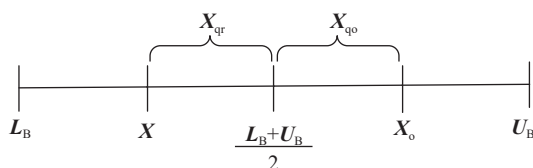


图 3 OBL、QOBL、QRBL 区别

Fig. 3 Differences between OBL, QOBL and QRBL

Fan 等^[22] 通过实验发现, 对于 HHO 的优化, QRBL 比 QOBL 和 OBL 更有优势, 但在个别函数

测试中, 发现 QOBL 得出的解可以获得更好效果。为此提出变异 QRBL 算法, 具体为

$$X_{\text{new}}(t) = \begin{cases} X_{qo}(t), & b \leq p_r \\ X_{qr}(t), & b > p_r \end{cases} \quad (22)$$

式中: b 是 $(0,1)$ 随机数; p_r 变异概率^[23] 是 $(0.01, 0.1)$ 之间的数, 经测试 $p_r=0.08$ 算法性能最好。

经变异反射学习策略后, 使用贪心策略将最好位置保留进入下一次迭代:

$$X(t+1) = \begin{cases} X_{\text{new}}(t+1), & f(X_{\text{new}}(t+1)) < f(X(t+1)) \\ X(t+1), & f(X_{\text{new}}(t+1)) \geq f(X(t+1)) \end{cases} \quad (23)$$

2.4 IHHO 算法实现过程

综上所述, IHHO 算法在原始 HHO 算法的基础上进行改进, 通过 Circle 混沌映射初始化种群, 提升初始化位置质量, 然后引入 Sigmoid 非线性收敛因子, 平衡全局探索和局部开发, 最后利用变异 QRBL 策略产生准反射或准反向位置, 更新种群向最优个体位置进化, 增强算法跳出局部最优的能力, 使得算法获得更好的收敛速度和收敛精度, IHHO 流程如图 4 所示。

IHHO 算法实现流程描述如下:

- 1) 使用 Circle 混沌映射初始化种群。
- 2) 计算个体适应度值, 记录较优个体。
- 3) 如果 $|E| \geq 1$, 个体根据式 (1)、式 (2) 开始探索行为。

- 4) 如果 $|E| < 1$, 个体根据逃逸猎物能量值历史适应度值改善情况选择开发策略。

策略 1 软包围: 当 $r \geq 0.5$ 且 $|E| \geq 0.5$, 子群个体采用式 (5)~(7) 更新位置;

策略 2 硬包围: 当 $r \geq 0.5$ 且 $|E| < 0.5$, 子群个体采用式 (8) 更新位置;

策略 3 渐进式快速俯冲软包围: 当 $r < 0.5$ 且 $|E| \geq 0.5$, 子群个体采用式 (9)~(13) 更新位置;

策略 4 渐进式快速俯冲硬包围: 当 $r < 0.5$ 且 $|E| < 0.5$, 子群个体采用式 (14)~(16) 更新位置;

采用式 (22)~(23) 计算变异反射解, 择优保留最好位置。

- 5) 判断是否满足终止条件, 满足则跳出循环, 否则返回步骤 2。

- 6) 输出最佳位置及适应度值。

2.5 运算复杂度分析

IHHO 算法主要由种群初始化、种群位置更新以及变异准反射解组成, 假设种群规模 N , 维度 D , 最大迭代次数 T , 则 IHHO 初始阶段时间复杂度为 $O(N \times D)$, 计算个体适应度为 $O(N)$, 迭代过程中利用变异 QRBL 生成解, 复杂度为 $O(2 \times N \times D \times T)$, 更新最优解的时间复杂度为 $O(1)$ 。综上所述, IHHO 运算复杂度为 $O(2 \times N \times D \times T)$ 。

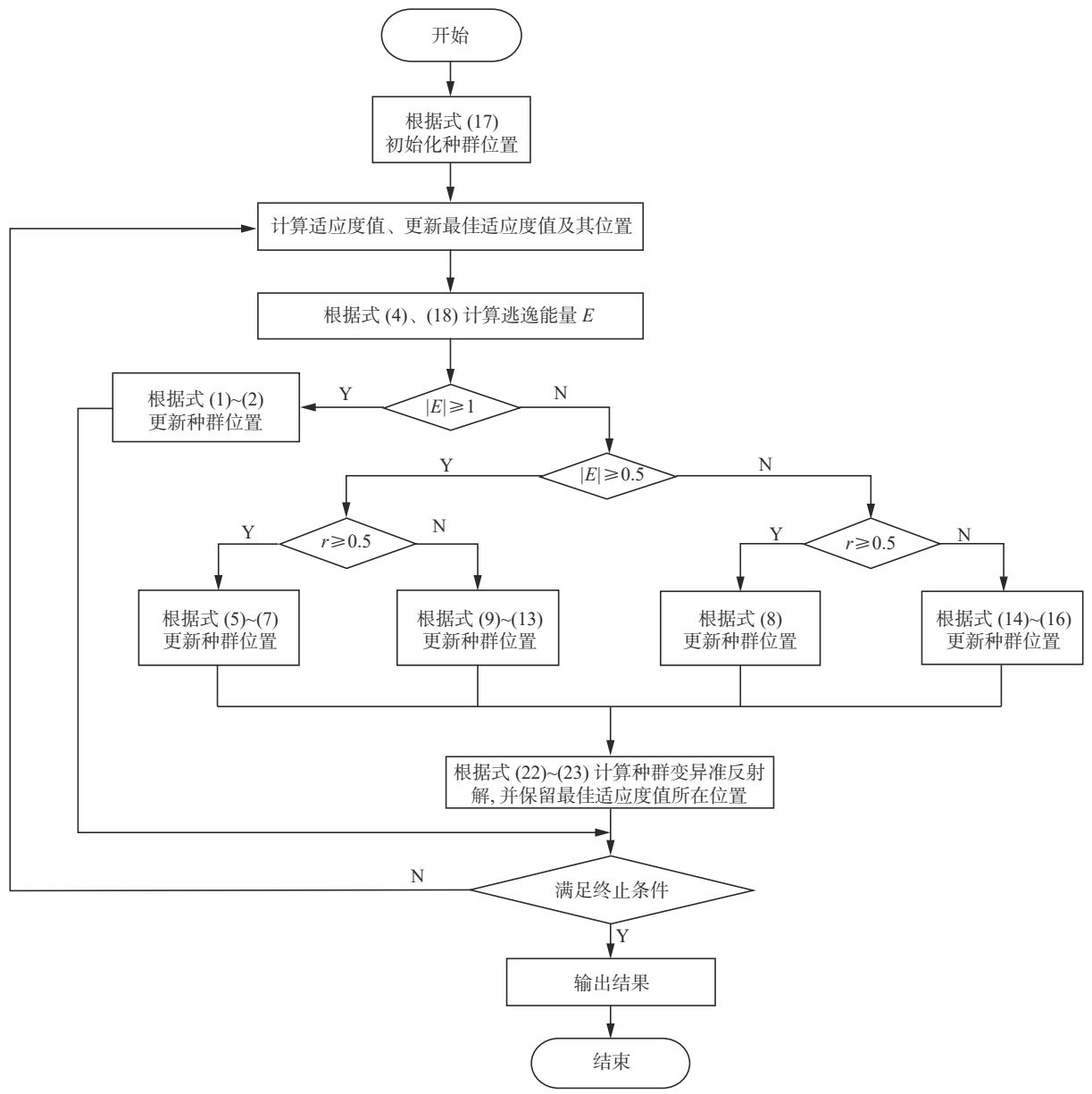


图 4 IHHO 算法流程

Fig. 4 Flowchart of IHHO

表 1 为 IHHO 和 HHO 算法在 13 个标准测试函数^[24](F_1 — F_{13})上(每个函数迭代 500 次)运行时间的对比。从表 1 可以看到各算法用时与测试函数本身复杂度相关, 时间差最大的为 F_{13} 函数, 算法运行时间相差 0.3 s 左右。虽然整体算法运行时间更长, 但是 IHHO 算法在部分测试函数中收敛速度远大于原 HHO 算法, 也就意味着 IHHO 在部分函数中不需要完整执行完 500 次迭代就已经达到最优效果, 这也给 IHHO 在时间上进行了一个有效缩减。

表 1 IHHO 和 HHO 运行时间对比
Table 1 Riming time of IHHO and HHO s

算法	IHHO	HHO
F_1	0.1789	0.0980

续表 1

算法	IHHO	HHO
F_2	0.1210	0.0799
F_3	0.6220	0.3843
F_4	0.0966	0.0852
F_5	0.2055	0.1448
F_6	0.1349	0.1000
F_7	0.3227	0.2082
F_8	0.2042	0.1379
F_9	0.1496	0.1150
F_{10}	0.1614	0.1199
F_{11}	0.1923	0.1382
F_{12}	0.7809	0.4953
F_{13}	0.7845	0.4874

3 实验仿真与分析

为验证 IHHO 算法的稳定性和有效性,使用 13 个标准测试函数对 IHHO 进行性能测试,其中, $F_1—F_7$ 为单峰测试函数, $F_8—F_{13}$ 为多峰测试函数,利用两种不同类型的测试函数,可充分验证算法的寻优性能。

3.1 对比算法与参数设置

为了更好的验证 IHHO 算法性能,选取 6 种性能良好的优化算法进行对比: HHO、SMA、WOA、SSA、GWO、SCA 和其他 HHO 优化算法: CEHHO (chaotic elite Harris Hawk optimization)、HHOCM 和 DHHO/M (dynamic Harris Hawks optimization with a mutation mechanism)。算法选取平均值、标准差与 Wilcoxon 秩和检验作为评价指标。设定种群规模 $N=30$, 维度 $D=30$, 最大迭代次数 $T=500$ 次,各算法独立运行 30 次,算法参数设置如表 2 所示。

表 2 各算法参数设置

Table 2 Parameter setting of each algorithm

算法	参数设置
IHHO	$\beta=1.5; a=0.5; b=0.2; p_r=0.8$
HHO ^[7]	$\beta=1.5$
SMA ^[6]	$z=0.03$
WOA ^[3]	$a_1=[2,0]; a_2=[-2,1]; b=1$
SSA ^[4]	$c_1 \in [0,1]; c_2 \in [0,1]$
GWO ^[2]	$a=[2,0]$
SCA ^[5]	$a=2$
CEHHO ^[13]	$\beta=1.5$
HHOCM ^[12]	$\beta=1.5$
DHHO/M ^[14]	$a=2.5; F=0.5; \beta=1.5$

3.2 求解精度分析

IHHO 与经典算法标准函数测试结果对比如表 3 所示,与其他 HHO 优化算法标准函数测试结果对比如表 4 所示。Mean 代表平均适应度值, Std 为标准差,加粗的实验数据为最佳结果。

表 3 与经典算法标准函数测试结果对比

Table 3 Test results of benchmark functions with classic algorithms

函数	统计值	IHHO	SMA	HHO	WOA	SSA	GWO	SCA
F_1	Mean	0.00	9.28×10^{-308}	9.70×10^{-97}	1.36×10^{-72}	3.23×10^{-7}	1.55×10^{-27}	1.15×10^1
	Std	0.00	0.00	5.14×10^{-96}	6.44×10^{-72}	9.51×10^{-7}	2.55×10^{-27}	2.37×10^1
	用时/s	0.103 3	0.214 5	0.055 3	0.040 3	0.059 7	0.096 0	0.069 7
F_2	Mean	0.00	5.82×10^{-146}	1.93×10^{-50}	2.54×10^{-50}	2.05	1.16×10^{-16}	1.58×10^{-2}
	Std	0.00	3.19×10^{-145}	9.76×10^{-50}	7.38×10^{-50}	1.71	1.22×10^{-16}	2.48×10^{-2}
	用时/s	0.101 2	0.225 8	0.053 6	0.041 4	0.054 5	0.099 0	0.069 2
F_3	Mean	0.00	1.59×10^{-315}	9.01×10^{-68}	4.10×10^4	1.42×10^3	1.03×10^{-5}	9.57×10^3
	Std	0.00	0.00	4.93×10^{-67}	1.49×10^4	8.61×10^2	3.17×10^{-5}	5.96×10^3
	用时/s	0.442 6	0.288 7	0.251 5	0.118 8	0.135 8	0.158 3	0.152 3
F_4	Mean	0.00	6.95×10^{-165}	1.34×10^{-48}	5.10×10^1	1.17×10^1	7.13×10^{-7}	3.83×10^1
	Std	0.00	0.00	6.99×10^{-48}	2.85×10^1	3.14	6.63×10^{-7}	1.34×10^1
	用时/s	0.101 3	0.238 8	0.067 0	0.040 4	0.054 9	0.099 7	0.069 9
F_5	Mean	2.24×10^{-3}	3.73	1.80×10^{-2}	2.80×10^1	1.76×10^2	2.72×10^1	8.17×10^4
	Std	2.12×10^{-3}	7.69	4.05×10^{-2}	4.20×10^{-1}	2.52×10^2	6.85×10^{-1}	3.74×10^5
	用时/s	0.223 2	0.314 9	0.136 31	0.069 0	0.088 7	0.136 1	0.106 9
F_6	Mean	1.05×10^{-5}	4.90×10^{-3}	1.63×10^{-4}	3.70×10^{-1}	1.31×10^{-7}	8.74×10^{-1}	1.68×10^1
	Std	2.29×10^{-5}	3.31×10^{-3}	2.30×10^{-4}	1.99×10^{-1}	1.17×10^{-7}	3.26×10^{-1}	2.95×10^1
	用时/s	0.117 0	0.271 1	0.092 6	0.041 4	0.062 0	0.130 7	0.080 6
F_7	Mean	1.07×10^{-4}	2.07×10^{-4}	1.15×10^{-4}	3.81×10^{-3}	1.60×10^{-1}	2.10×10^{-3}	1.00×10^{-1}
	Std	8.94×10^{-5}	1.74×10^{-4}	1.22×10^{-4}	4.28×10^{-3}	6.59×10^{-2}	1.34×10^{-3}	8.06×10^{-2}
	用时/s	0.307 5	0.267 5	0.179 1	0.089 5	0.105 7	0.149 6	0.120 3
F_8	Mean	-1.26×10^4	-1.26×10^4	-1.26×10^4	-1.04×10^4	-7.48×10^3	-5.81×10^3	-3.79×10^3
	Std	4.26×10^{-1}	3.15×10^{-1}	7.93×10^1	1.71×10^3	7.04×10^2	1.05×10^3	3.43×10^2
	用时/s	0.187 4	0.283 8	0.112 4	0.053 8	0.065 1	0.110 6	0.085 3

续表 3

函数	统计值	IHHO	SMA	HHO	WOA	SSA	GWO	SCA
F_9	Mean	0.00	0.00	0.00	0.00	5.09×10^1	3.12	4.55×10^1
	Std	0.00	0.00	0.00	0.00	1.38×10^1	3.86	3.22×10^1
	用时/s	0.1308	0.2461	0.0846	0.0428	0.0611	0.1003	0.0758
F_{10}	Mean	8.88×10^{-16}	8.88×10^{-16}	8.88×10^{-16}	4.68×10^{-15}	2.31	1.02×10^{-13}	1.48×10^1
	Std	0.00	0.00	0.00	2.46×10^{-15}	7.60×10^{-1}	1.85×10^{-14}	8.64
	用时/s	0.1463	0.2390	0.0901	0.0457	0.0642	0.1060	0.0844
F_{11}	Mean	0.00	0.00	0.00	1.94×10^{-2}	1.57×10^{-2}	2.81×10^{-3}	1.06
	Std	0.00	0.00	0.00	6.12×10^{-2}	1.21×10^{-2}	6.05×10^{-3}	4.68×10^{-1}
	用时/s	0.1743	0.2583	0.1078	0.0554	0.0744	0.1079	0.1087
F_{12}	Mean	6.63×10^{-7}	5.56×10^{-3}	7.65×10^{-6}	2.02×10^{-2}	6.00	4.62×10^{-2}	5.35×10^4
	Std	8.42×10^{-7}	7.88×10^{-3}	7.68×10^{-6}	1.11×10^{-2}	2.71	2.01×10^{-2}	2.26×10^5
	用时/s	0.7282	0.3451	0.4208	0.1752	0.1981	0.2313	0.2105
F_{13}	Mean	1.02×10^{-5}	3.84×10^{-3}	8.08×10^{-5}	5.59×10^{-1}	1.23×10^1	6.65×10^{-1}	3.96×10^5
	Std	1.59×10^{-5}	3.99×10^{-3}	1.37×10^{-4}	2.43×10^{-1}	1.38×10^1	2.40×10^{-1}	1.62×10^6
	用时/s	0.6896	0.6896	0.4443	0.1786	0.2245	0.2339	0.2170

表 4 与其他 HHO 优化算法标准函数测试结果对比

Table 4 Test results of benchmark functions with other HHO optimization algorithms

函数	统计值	IHHO	HHOCM	CEHHO	DHHO/M
F_1	Mean	0.00	0.00	0.00	8.16×10^{-92}
	Std	0.00	0.00	0.00	3.92×10^{-91}
F_2	Mean	0.00	2.06×10^{-198}	0.00	5.26×10^{-48}
	Std	0.00	0.00	0.00	2.39×10^{-47}
F_3	Mean	0.00	0.00	0.00	4.62×10^{-60}
	Std	0.00	0.00	0.00	2.53×10^{-59}
F_4	Mean	0.00	4.90×10^{-192}	0.00	1.89×10^{-45}
	Std	0.00	0.00	0.00	1.02×10^{-44}
F_5	Mean	2.24×10^{-3}	1.80×10^{-2}	4.26×10^{-3}	1.15×10^{-2}
	Std	2.12×10^{-3}	2.47×10^{-2}	7.03×10^{-3}	1.17×10^{-2}
F_6	Mean	1.05×10^{-5}	4.82×10^{-4}	1.65×10^{-5}	6.56×10^{-5}
	Std	2.29×10^{-5}	6.54×10^{-4}	2.32×10^{-5}	9.79×10^{-5}
F_7	Mean	1.07×10^{-4}	1.52×10^{-4}	1.33×10^{-4}	1.58×10^{-4}
	Std	8.94×10^{-5}	1.46×10^{-4}	1.09×10^{-4}	1.31×10^{-4}
F_8	Mean	-1.26×10^4	-1.26×10^4	-1.26×10^4	-1.24×10^4
	Std	4.26×10^{-1}	7.51×10^{-1}	6.03×10^1	9.65×10^2
F_9	Mean	0.00	0.00	0.00	0.00
	Std	0.00	0.00	0.00	0.00
F_{10}	Mean	8.88×10^{-16}	8.88×10^{-16}	8.88×10^{-16}	8.88×10^{-16}
	Std	0.00	0.00	0.00	0.00
F_{11}	Mean	0.00	0.00	0.00	0.00
	Std	0.00	0.00	0.00	0.00
F_{12}	Mean	6.63×10^{-7}	1.66×10^{-5}	1.73×10^{-6}	5.44×10^{-6}
	Std	8.42×10^{-7}	2.41×10^{-5}	2.37×10^{-6}	6.46×10^{-6}

续表 4

函数	统计值	IHHO	HHOCM	CEHHO	DHHO/M
F_{13}	Mean	1.02×10^{-5}	1.50×10^{-4}	1.57×10^{-5}	7.29×10^{-5}
	Std	1.59×10^{-5}	1.42×10^{-4}	3.16×10^{-5}	9.17×10^{-5}

从表 3~4 可以看出, 在求解 F_1 — F_7 单峰测试函数中, IHHO 在 F_1 — F_4 测试函数中达到理论最优值, F_5 和 F_7 中 Mean 和 Std 虽然没有达到理论最优值, 但对比经典算法和其他 HHO 优化算法都取得最好结果。对于 F_6 , IHHO 虽然没有收敛到全局最优值, 但求解精度仅次于 SSA。可见, IHHO 应用变异准反射学习对于提升局部收敛精度和稳定性有一定作用。在多峰测试函数中, 对于 F_9 — F_{13} 函数, IHHO 在所有对比算法中表现最好; 对于 F_8 , IHHO 平均适应度最优, 方差仅次于 SMA。可见 IHHO 算法在多峰测试函数中也取得较好效果, 证明 IHHO 的全局搜索能力和跳出局部最优能力整体优于原始 HHO 和其他对比算法。结合测试函数的复杂性, 证实了 IHHO 在平衡探索阶段和局部阶段的能力确实得到增强。通过表 3 还可以看到, IHHO 对比 HHO 的收敛精度有非常大的提升, 在多项测试函数中可以达到理论最优值, 因此本文在性能上的提升有效弥补了运行时间的增加。

3.3 收敛曲线分析

为了更直观地展示各算法的收敛速度及跳

出局部最优能力, 将 IHHO 与经典算法收敛曲线进行对比分析(图 5), 选取单峰函数 F_3 、 F_5 和多峰函数 F_9 和 F_{11} 。可以看到对于函数 F_3 , IHHO 在前期没有显示巨大优势, 但是随着迭代增加, IHHO 进入局部开发, 收敛速度明显加快, 迅速达到全局最优, 证明变异准反射策略有效提高算法局部寻优的收敛速度。对于多峰测试函数 F_9 和 F_{11} , IHHO 在迭代初期(50 次内)就迅速找到全局最优, 证明算法稳定的全局探索能力。在与其他优秀的 HHO 改进算法的对比中(图 6), 选取单峰函数 F_2 、 F_5 和多峰函数 F_9 和 F_{12} 。对于函数 F_2 , IHHO 在前期没有很大优势, 曲线与 CEHHO 和 DHHO/M 相当, 略逊色于 HHOCM, 但是在 250 次迭代以后, IHHO 迅速下降, 最后收敛精度明显优于 HHOCM, 可见 IHHO 优秀的局部探索能力。 F_5 函数同样在迭代前期就迅速下降, 收敛精度远胜于其他对比算法。对于 F_9 和 F_{12} , IHHO 无论从收敛速度和收敛精度上都优于其他算法。可以看到 IHHO 对于多峰函数有效性。

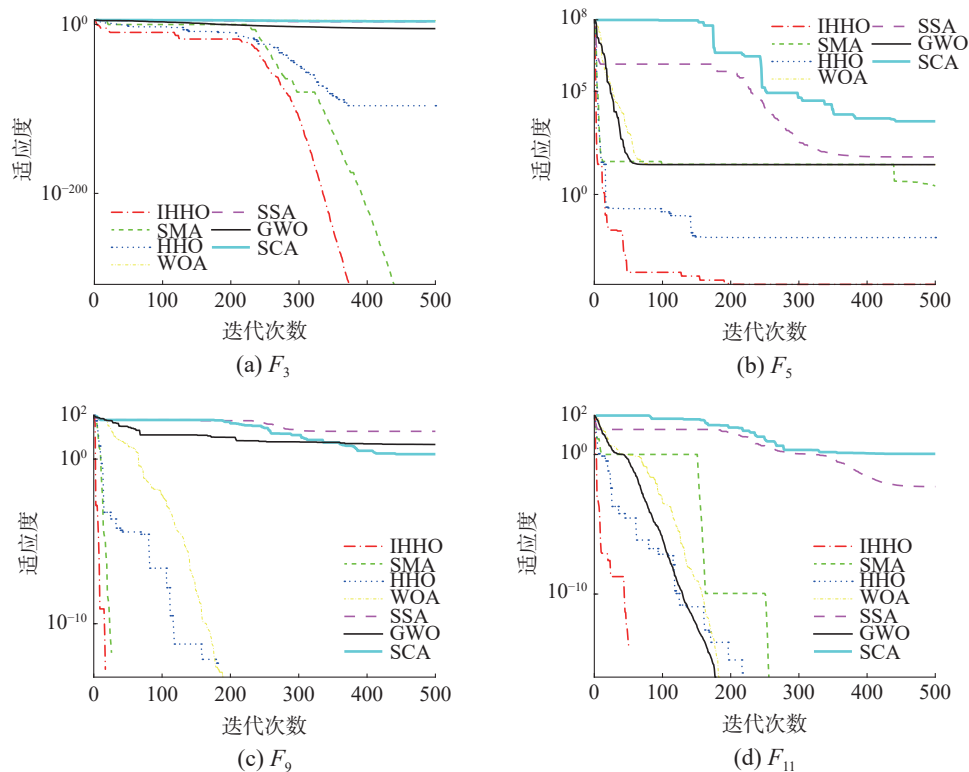


图 5 与经典算法收敛曲线对比

Fig. 5 Convergence curves with classic algorithms

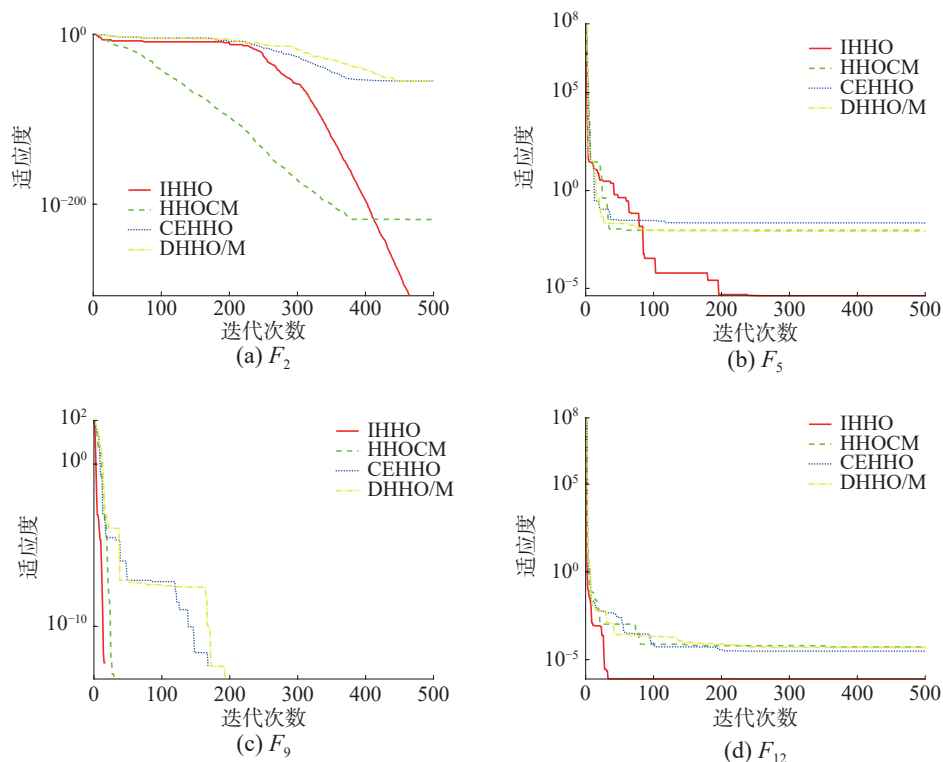


图 6 与其他 HHO 优化算法收敛曲线对比

Fig. 6 Convergence curve with other HHO optimizer

3.4 工程设计优化问题

本节利用压力容器设计优化问题压力容器设计优化问题^[25](如图 7)测试 IHHO 对于实际问题的优化性能,实验参数设置见表 2。该问题是一个重要的研究领域,它旨在提高压力容器的性能、可靠性和成本效益。

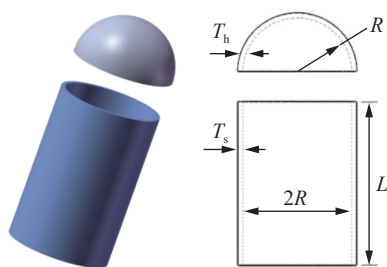


图 7 压力容器设计问题

Fig. 7 Pressure vessel design problem

本章节内容旨在通过 IHHO 算法找到合适的优化容器壁厚 T_s 、顶盖壁厚 T_h 、内径 R 和容器管深长度 L ,使容器在达到一定条件下付出成本最小,问题模型考虑:

$$\mathbf{X} = [x_1 \ x_2 \ x_3 \ x_4] = [T_s \ T_h \ R \ L]$$

使得下面条件最小:

$$f(\mathbf{X}) = 0.622 \ 4x_1x_3x_4 + 1.778 \ 1x_2x_3^2 + 3.166 \ 1x_1^2x_4 + 19.84x_1^2x_3$$

满足:

$$g_1(\mathbf{x}) = -x_1 + 0.019 \ 3x_3 \leq 0$$

$$g_2(\mathbf{x}) = -x_3 + 0.009 \ 54x_3 \leq 0$$

$$g_3(\mathbf{x}) = -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1 \ 296 \ 000 \leq 0$$

$$g_4(\mathbf{x}) = x_4 - 240 \leq 0$$

变量范围为

$$0 \leq x_1 \leq 99$$

$$0 \leq x_2 \leq 99$$

$$10 \leq x_3 \leq 200$$

$$10 \leq x_4 \leq 200$$

表 5 列出了 IHHO 及对比算法求解压力容器设计问题的实验结果。通过数据可以看出, IHHO 收敛精度对比其他算法有不同程度的提升。可以看出, IHHO 具备一定的工程优化能力,可应用于实际工程问题。

表 5 压力容器设计问题优化结果

Table 5 The optimization results of pressure vessel problem

算法	$T_s(x_1)$	$T_h(x_2)$	$R(x_3)$	$L(x_4)$	结果
IHHO	0.8251	0.4066	44.8326	145.5222	5928.67
HHO	0.8815	0.4267	47.2647	121.6444	6030.17

续表 5

算法	$T_s(x_1)$	$T_h(x_2)$	$R(x_3)$	$L(x_4)$	结果
SMA	0.865 6	0.423 6	46.419 1	129.560 5	5 989.18
WOA	0.765 1	0.453 4	41.140 0	188.887 1	5 976.27
SSA	0.903 8	0.440 2	48.315 7	112.296 0	6 078.02
GWO	0.885 9	0.434 5	47.429 1	120.162 5	6 036.72
SCA	0.832 0	0.372 8	43.483 6	161.843 2	6 031.48
HHOCM	0.861 4	0.421 9	46.220 1	131.477 9	5 980.16
CEHHO	0.897 2	0.473 3	49.117 2	105.507 5	6 234.97
DHHO/M	0.818 9	0.425 0	44.902 3	144.735 9	5 969.54

4 结束语

结合传统 HHO 算法的优势和不足, 受变异反向学习启发, 提出了一种新型改进的哈里斯鹰优化算法(IHHO)。IHHO 融合 Sigmoid 非线性收敛因子, 有效平衡了全局搜索和局部探索的转换能力, 应用变异 RQBL 策略能够有效增强种群多样性, 提高算法收敛速度和跳出局部最优的能力。通过 13 个标准函数和工程应用问题的测试结果表明, 不论对于单峰函数、多峰函数还是实际优化问题, 均具有较佳的收敛速度和精度。在后续研究中, 尝试将变异准反射学习策略应用到其他优化算法, 以验证该策略应用的广泛性。

参考文献:

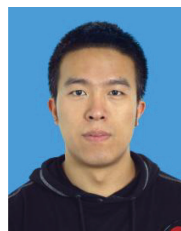
- [1] KENNEDY J, EBERHART R. Particle swarm optimization[C]//Proceedings of ICNN'95 - International Conference on Neural Networks. Perth: IEEE, 2002: 1942–1948.
- [2] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. *Advances in engineering software*, 2014, 69: 46–61.
- [3] MIRJALILI S, LEWIS A. The whale optimization algorithm[J]. *Advances in engineering software*, 2016, 95(C): 51–67.
- [4] MIRJALILI S, GANDOMI A H, MIRJALILI S Z, et al. Salp swarm algorithm[J]. *Advances in engineering software*, 2017, 114(C): 163–191.
- [5] MIRJALILI S. SCA: a sine cosine algorithm for solving optimization problems[J]. *Knowledge based systems*, 2016, 96: 120–133.
- [6] LI Shimin, CHEN Huiling, WANG Mingjing, et al. Slime mould algorithm: a new method for stochastic optimization[J]. *Future generation computer systems*, 2020, 111: 300–323.
- [7] 贾鹤鸣, 李瑶, 孙康健. 基于遗传乌燕鸥算法的同步优化特征选择[J]. *自动化学报*, 2022, 48(6): 1601–1615.
- JIA Heming, LI Yao, SUN Kangjian. Simultaneous feature selection optimization based on hybrid sooty tern optimization algorithm and genetic algorithm[J]. *Acta automatica sinica*, 2022, 48(6): 1601–1615.
- [8] HEIDARI A A, MIRJALILI S, FARIS H, et al. Harris Hawks optimization: algorithm and applications[J]. *Future generation computer systems*, 2019, 97: 849–872.
- [9] JIA Heming, PENG Xiaoxu, KANG Lifei, et al. Pulse coupled neural network based on Harris Hawks optimization algorithm for image segmentation[J]. *Multimedia tools and applications*, 2020, 79(37): 28369–28392.
- [10] FAN Chencheng, ZHOU Yongquan, TANG Zhonghua. Neighborhood centroid opposite-based learning Harris Hawks optimization for training neural networks[J]. *Evolutionary intelligence*, 2021, 14(4): 1847–1867.
- [11] SARAVANAN G, IBRAHIM A, KUMAR D et al. Iot based speed control of BLDC motor with Harris hawks optimization controller[J]. *International journal of grid and distributed computing*, 2020, 13: 1902–1915.
- [12] 马一鸣, 石志东, 赵康, 等. 基于改进哈里斯鹰优化算法的 TDOA 定位[J]. *计算机工程*, 2020, 46(12): 179–184.
- MA Yiming, SHI Zhidong, ZHAO Kang, et al. TDOA localization based on improved Harris hawk optimization algorithm[J]. *Computer engineering*, 2020, 46(12): 179–184.
- [13] HOUSSEIN E H, NEGGAZ N, HOSNEY M E, et al. Enhanced Harris Hawks optimization with genetic operators for selection chemical descriptors and compounds activities[J]. *Neural computing and applications*, 2021, 33(20): 13601–13618.
- [14] 汤安迪, 韩统, 徐登武, 等. 混沌精英哈里斯鹰优化算法[J]. *计算机应用*, 2021, 41(8): 2265–2272.
- TANG Andi, HAN Tong, XU Dengwu, et al. Chaotic elite Harris Hawks optimization algorithm[J]. *Journal of computer applications*, 2021, 41(8): 2265–2272.
- [15] JIA Heming, LANG Chunbo, OLIVA D, et al. Dynamic Harris Hawks optimization with mutation mechanism for satellite image segmentation[J]. *Remote sensing*, 2019, 11(12): 1421.

- [16] MIRJALILI S, GANDOMI A H. Chaotic gravitational constants for the gravitational search algorithm[J]. *Applied soft computing*, 2017, 53(C): 407–419.
- [17] 王宁, 何庆. 融合黄金正弦与 sigmoid 连续化的海鸥优化算法 [J]. *计算机应用研究*, 2022, 39(1): 157–162, 169. WANG Ning, HE Qing. Seagull optimization algorithm combining golden sine and sigmoid continuity[J]. *Application research of computers*, 2022, 39(1): 157–162, 169.
- [18] WANG Shuang, LIU Qingxin, LIU Yuxiang, et al. A hybrid SSA and SMA with mutation opposition-based learning for constrained engineering problems[J]. *Computational intelligence and neuroscience*, 2021, 2021: 6379469.
- [19] TIZHOOSH H R. Opposition-based learning: a new scheme for machine intelligence[C]//International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce. Vienna: IEEE, 2006: 695–701.
- [20] RAHNAMAYAN S, TIZHOOSH H R, SALAMA M M A. Quasi-oppositional differential evolution[C]//2007 IEEE Congress on Evolutionary Computation. Singapore: IEEE, 2007: 2229–2236.
- [21] EWEES A A, ABD ELAZIZ M, HOUSSEIN E H. Improved grasshopper optimization algorithm using opposition-based learning[J]. *Expert systems with applications*, 2018, 112: 156–172.
- [22] FAN Qian, CHEN Zhenjian, XIA Zhanghua. A novel quasi-reflected Harris Hawks optimization algorithm for global optimization problems[J]. *Soft computing*, 2020, 24(19): 14825–14843.
- [23] 李守玉, 何庆, 杜逆索. 分段权重和变异反向学习的蝴蝶优化算法 [J]. *计算机工程与应用*, 2021, 57(22): 92–101. LI Shouyu, HE Qing, DU Nisuo. Piecewise weight and mutation opposition-based learning butterfly optimization algorithm[J]. *Computer engineering and applications*, 2021, 57(22): 92–101.
- [24] 贾鹤鸣, 姜子超, 李瑶. 基于改进秃鹰搜索算法的同步优化特征选择 [J]. *控制与决策*, 2022, 37(2): 445–454. JIA Heming, JIANG Zichao, LI Yao. Simultaneous feature selection optimization based on improved bald eagle search algorithm[J]. *Control and decision*, 2022, 37(2): 445–454.
- [25] SONG Meijia, JIA Heming, ABUALIGAH L, et al. Modified Harris Hawks optimization algorithm with exploration factor and random walk strategy[J]. *Computational intelligence and neuroscience*, 2022, 2022: 4673665.

作者简介:



宋美佳, 助理实验师, 主要研究方向为群体智能优化算法、数据挖掘。
E-mail: 20200125@fjismu.edu.cn。



贾鹤鸣, 教授, 主要研究方向为群体智能优化算法与工程应用。主持福建省自然科学基金等项目 10 余项。发表学术论文 60 余篇。E-mail: jiaheminglucky99@126.com。



林志兴, 高级实验师, 主要研究方向为信息安全。E-mail: 6051751@qq.com。