杨海马, 郑和庆, 黄宏欣. 一种混合策略改进的哈里斯鹰算法及应用[J]. 智能计算机与应用,2024,14(6):169-176. DOI:10.20169/j.issn.2095-2163.240624

一种混合策略改进的哈里斯鹰算法及应用

杨海马, 郑和庆, 黄宏欣
(上海理工大学 光电信息与计算机工程学院, 上海 200093)

摘 要: 针对标准的哈里斯鹰优化算法收敛精度不高,易陷入局部最优等缺点,提出一种混合策略改进的哈里斯鹰优化算法DMHHO。利用非线性周期递减的逃逸能量因子来平衡算法的全局探索和局部开发能力;引入多项式变异策略,对全局最优个体进行扰动,增强算法跳出局部极值的能力;采用变种差分策略,对个体的位置进行变异,增加算法的种群多样性,提升了算法的全局寻优能力。通过 12 个测试函数对 DMHHO 算法进行寻优测试,并与其他优化算法对比,实验结果表明改进的哈里斯鹰优化算法的收敛速度和寻优精度都得到了提升。将改进的哈里斯鹰优化算法用在工程优化问题中,进一步验证了DMHHO 算法在实际应用中可行性。

关键词: 哈里斯鹰优化算法; 非线性周期能量递减; 变种差分策略; 多项式变异
中图分类号: TP301.6 **文献标志码**: A **文章编号**: 2095-2163(2024)06-0169-08

A hybrid strategy improved Harris Hawks algorithm and its application

YANG Haima, ZHENG Heqing, HUANG Hongxin
(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: A hybrid strategy improved Harris Hawks Optimization algorithm DMHHO is proposed to improve the convergence accuracy of the standard Harris Hawks Optimization algorithm, which is easy to fall into the local optimum. The ability of global exploration and local development are balanced by the escape energy factor of nonlinear period decline. Polynomial variation strategy is introduced to perturb the global optimal individual and enhance the global exploration ability of the algorithm. The variable difference strategy is used to mutate individual positions, increase the diversity of the algorithm population, and improve the optimization ability of the algorithm. DMHHO algorithm is tested by 12 test functions, and compared with other optimization algorithms. The experimental results show that the convergence speed and optimization accuracy of the improved Harris Hawks Optimization algorithm are improved. The improved Harris Hawks Optimization algorithm is applied to engineering optimization problems, which further verifies the feasibility of DMHHO algorithm in practical application.

Key words: Harris Hawks Optimization; nonlinear periodic energy decreases; variable difference strategy; polynomial mutations

0 引 言

近年来,随着科技的快速发展,人们需要面对越来越复杂的难题,例如工程设计问题,复杂计算问题等。传统的算法在解决这些复杂的问题时,由于问题计算量的庞大,导致计算时间长,不能满足实际的需求。针对这一状况,国内外的专家学者提出了一系列的智能优化算法,并取得了不错的效果。目前,群智能优化算法已经被广泛应用于图像处理、控制优化、电力等等[1-4]领域。主流的群智能优化算法有:遗传算法[5](Genetic Algorithm, GA)、粒子群算法[6](Particle Swarm Optimization, PSO)、差分进化算法[7](Differential Evolution, DE)、灰狼优化算法[8](Grey Wolf Optimizer, GWO)、鲸鱼优化算法[9]

基金项目: 中科院空间主动光电技术重点实验室开放基金(2021ZDKF4); 上海市科委科技创新行动计划(21S31904200, 22S31903700)。
作者简介: 郑和庆(1996-),男,硕士研究生,主要研究方向:智能算法与应用; 黄宏欣(1998-),女,硕士研究生,主要研究方向:光电信息技术,图像检测技术。
通讯作者: 杨海马(1979-),男,博士,副教授,主要研究方向:优化算法,空间目标跟踪控制和三维测量与信息提取,等。Email:snowyhm@sina.com
收稿日期: 2023-04-11

(Whale Optimization Algorithm, WOA)、正余弦优化算法[10](Sine Cosine Algorithm, SCA)和麻雀搜索算法[11](Sparrow Search Algorithm, SSA)等。

哈里斯鹰优化算法[12](Harris Hawks Optimization, HHO)是学者 Heidari 等学者受哈里斯鹰种群捕猎行为启发在 2019 年提出的一种新型群智能优化算法,该算法相对于其他优化算法,具有稳定性强,结构简单等优点,但是 HHO 算法面对复杂优化问题也存在容易陷入局部极值和收敛精度不高等缺点。为了改善 HHO 算法的不足,许多学者提出了不同的改进策略。Zhang 等学者[13]探讨了不同的逃逸能量 E 的更新方式对算法的影响。汤安迪等学者[14]引入精英等级策略来增强种群的多样性,使用非线性逃逸能量因子来平衡算法的全局探索阶段和局部开发阶段,最后通过高斯变异来对最优个体进行变异扰动,帮助算法跳出局部最优。Li 等学者[15]通过引入反向学习和螺旋探索等策略提升算法的全局探索能力。聂春芳[16]在算法的探索阶段引入黄金正余弦算法,来提高算法的探索能力,在开发阶段加入高斯随机游走策略,加强算法的开发能力。李云秋等学者[17]引入变邻域搜索策略,提高算法的局部搜索能力,采用逐维柯西高斯变异策略来提高算法的跳出局部最优能力。孙林等学者[18]在算法初始化阶段引入 sine 映射函数来初始化种群,使用自适应调整算子和动态跳跃距离来改变算法的步长,增强了算法的寻优能力。

这些改进策略都提升了原始算法的寻优能力,但是哈里斯鹰算法还有很大的改进潜力,本文针对哈里斯鹰优化算法存在的缺点,提出一种混合策略改进的哈里斯鹰优化算法 DMHHO(Differential Multinomial HHO)。首先,通过引入非线性周期性递减的逃逸能量 E, 平衡了算法的全局探索和局部开发能力;其次,引入多项式变异策略,对全局最优个体进行扰动,减小算法陷入局部极值的可能性;最后,通过变种差分策略对个体更新后的位置进行变异,增加种群的多样性,提高算法的寻优能力。通过 12 个基准测试函数进行寻优测试,并且与其他优化算法进行对比,实验结果证明了 DMHHO 算法的有效性。研究又将算法应用在三杆桁架设计问题中,实验结果表明,对比其他算法的结果,DMHHO 算法有更强的寻优能力,具有一定的应用价值。

1 哈里斯鹰优化算法

哈里斯鹰优化算法是受到哈里斯鹰狩猎行为启发提出的一种新型元启发式算法。该算法分为 3 个阶段,分别为全局探索阶段、局部开发阶段和转换阶段。

1.1 全局探索阶段

在全局探索阶段,哈里斯鹰会在狩猎区域内,大步长搜索猎物的位置,通过 2 种策略来更新自己的位置,这 2 种方式的机会是相等的,位置更新公式如下:

$$X(t+1)=X_{\text{rand}}(t)-r_1\left|X_{\text{rand}}(t)-2r_2X(t)\right|,\quad q\geqslant 0.5 \tag{1}$$

$$X(t+1)=\left(X_{\text{rabbit}}(t)-X_m(t)\right)-r_3(lb+r_4(ub-lb)),\quad q<0.5 \tag{2}$$

$$X_m(t)=\frac{1}{N}\sum_{i=1}^{N}X_i(t) \tag{3}$$

其中,$X(t)$ 表示第 t 次迭代的位置;$X(t+1)$ 表示第 $t+1$ 次迭代的位置;$X_{\text{rand}}(t)$ 表示在当前种群中随机选择的个体位置;$X_{\text{rabbit}}(t)$ 表示猎物的位置、即全局最优解;ub 表示种群的上界;lb 表示种群的下界;$X_m(t)$ 表示种群的平均位置;q、r_1、r_2、r_3、r_4 为 0~1 之间的随机数。

1.2 转换阶段

在哈里斯鹰优化算法中,控制哈里斯鹰个体执行探索行为或者开发行为的是逃逸能量 E, 可由式(4)、式(5)来描述:

$$E_1=1-\frac{t}{T} \tag{4}$$

$$E=2\times E_0\times E_1 \tag{5}$$

其中,t 表示当前的迭代次数;T 表示最大迭代次数;E_0 表示-1~1 之间的随机数;E_1 表示线性因子。当 $|E|\geqslant 1$ 时,算法执行探索阶段,大步长搜索猎物位置。当 $|E|<1$ 时,算法执行开发阶段,进行局部精细搜索。

1.3 局部开发阶段

在进入开发阶段后,哈里斯鹰已经将猎物包围起来了,此时哈里斯鹰将会对猎物发起进攻,哈里斯鹰有多种进攻策略,分别为软包围、硬包围、俯冲式软包围和俯冲式硬包围。通过逃逸能量 E 和随机数 k 控制选择进攻策略,k 为 0~1 之间的随机数。

1.3.1 软包围

当 $|E|\geqslant 0.5$ 且 $k\geqslant 0.5$ 时,哈里斯鹰采用软包围策略,数学表达如式(6)所示:

$$\boldsymbol{X}(t+1)=\left(\boldsymbol{X}_{\text{rabbit}}(t)-\boldsymbol{X}(t)\right)-E\left|J\times\boldsymbol{X}_{\text{rabbit}}(t)-\boldsymbol{X}(t)\right| \tag{6}$$

其中,J 表示猎物跳跃能量,是一个范围为0~2之间的随机数,E 表示猎物的逃逸能量。

1.3.2　硬包围

当 $|E|<0.5$ 且 $k\geqslant 0.5$ 时，哈里斯鹰采用硬包围策略，此时猎物的能量已经耗尽了，无法进行跳跃，可由式(7)表示为：

$$\boldsymbol{X}(t+1)=\boldsymbol{X}_{\text{rabbit}}(t)-E\,|\,\boldsymbol{X}_{\text{rabbit}}(t)-\boldsymbol{X}(t)\,| \tag{7}$$

1.3.3　俯冲式软包围

当 $|E|\geqslant 0.5$ 且 $k<0.5$ 时，哈里斯鹰采用俯冲软包围策略，此时被包围的猎物还具有较大的逃逸能量 E，猎物的能量没有耗尽，还具有跳跃能力，这时哈里斯鹰会用 2 种方式进攻。如果方式一进攻没有效果，则采用方式二；如果方式二也没有效果，则保持原来的位置不变。这一过程可描述为式(8)～式(11)：

$$\boldsymbol{Y}=\boldsymbol{X}_{\text{rabbit}}(t)-E\,|J\times\boldsymbol{X}_{\text{rabbit}}(t)-\boldsymbol{X}(t)\,| \tag{8}$$

$$\boldsymbol{Z}=\boldsymbol{Y}+\boldsymbol{S}\times LF(D) \tag{9}$$

$$LF(x)=0.01\times\frac{\mu\times\delta}{|v|^{\frac{1}{\beta}}},\sigma=\left(\frac{\tau(1+\beta)\times\sin\left(\frac{\pi\beta}{2}\right)}{\tau\left(\frac{1+\beta}{2}\right)\times\beta\times 2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \tag{10}$$

$$\boldsymbol{X}(t+1)=\begin{cases}\boldsymbol{Y}, & f(\boldsymbol{Y})<f(\boldsymbol{X}(t))\\ \boldsymbol{Z}, & f(\boldsymbol{Z})<f(\boldsymbol{X}(t))\end{cases} \tag{11}$$

其中，$\boldsymbol{S}$ 表示一个 D 维随机向量；$LF(D)$ 表示 Levy 飞行函数；$f(x)$ 表示适应度函数。

1.3.4　俯冲式硬包围

当 $|E|<0.5$ 并且 $k<0.5$ 时，哈里斯鹰采用俯冲硬包围策略，此时被包围的猎物的逃逸能量 E 不足，不具备跳跃能力。这时哈里斯鹰也会用 2 种方式进攻。如果方式一进攻失败，则采用方式二；如果方式二也进攻失败，则保持原来的位置不变。这一过程可描述为式(12)～式(14)：

$$\boldsymbol{Y}=\boldsymbol{X}_{\text{rabbit}}(t)-E\,|J\times\boldsymbol{X}_{\text{rabbit}}(t)-\boldsymbol{X}_m(t)\,| \tag{12}$$

$$\boldsymbol{Z}=\boldsymbol{Y}+\boldsymbol{S}\times LF(D) \tag{13}$$

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

2　改进的哈里斯鹰优化算法

2.1　非线性周期递减的逃逸能量 E

如何平衡全局探索阶段和局部开发阶段对于算法来说非常重要，因为全局探索阶段可以确定最优解的大致范围，局部开发阶段能够提高算法的寻优精度，二者平衡，算法才会有一个较为优秀的寻优能力；哈里斯鹰优化算法的探索阶段和开发阶段受到逃逸能量 E 的控制，在 HHO 算法中逃逸能量 E 是线性递减的，如图 1 所示。当逃逸能量 $|E|\geqslant 1$ 时，算法执行探索阶段，此时算法具有较强的全局搜索能力；当逃逸能量 $|E|<1$ 时，算法执行开发阶段，此时算法具有较强局部搜索能力。在原始 HHO 算法中，当迭代次数 $t\geqslant T/2$ 时，此时逃逸能量 E 始终小于 1，则算法始终执行开发阶段，容易陷入到局部最优，导致算法早熟。为了解决算法中期以后只执行开发阶段这个问题，提出一种融合非线性递减和周期起伏[19]的逃逸能量 E，如图 2 所示，推得的数学公式如下所示：

$$E_2=1-\frac{t}{T}\times\frac{t}{T} \tag{15}$$

$$E=2\times E_0\times E_2\times\cos\left(2k\pi\times\frac{t}{T}\right) \tag{16}$$

其中，t 表示当前迭代次数；T 表示最大迭代次数；E_2 表示非线性因子；k 的取值为 4。可以看出，改进后的能量逃逸 E 总体的趋势是前期下降的速度较慢、注重全局探索，后期快速下降、注重局部开发，最终趋向于 0，加强收敛能力，平衡了全局搜索和局部搜索能力，提升算法对搜索空间的探索能力，使算法在迭代中期以后还有机会执行探索阶段，有利于算法跳出局部最优解。综上，改进后的逃逸能量 E 能更好地平衡全局探索能力和局部开发能力。

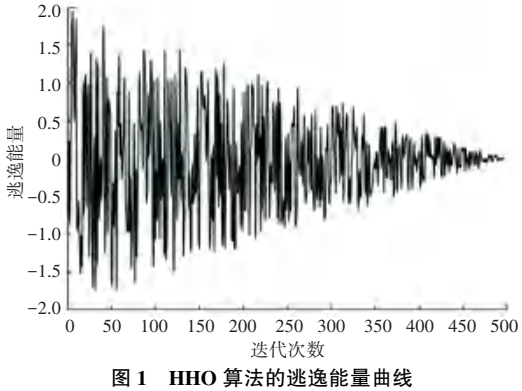

图 1　HHO 算法的逃逸能量曲线

Fig. 1　Escape energy curve of HHO algorithm

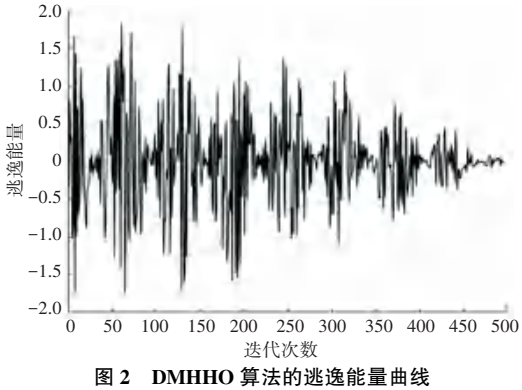

图 2　DMHHO 算法的逃逸能量曲线

Fig. 2　Escape energy curve of DMHHO algorithm

2.2 基于多项式变异的最优解扰动

从标准的哈里斯鹰优化算法的位置更新公式可以看出,大量使用了全局最优个体对其他个体进行引导,因此整个种群都在向最优个体靠近,当算法在处理简单的单峰问题时,可以使算法快速地收敛,取得一个比较好的结果。但是处理复杂的多峰问题时,该问题由于存在多个局部极值,一旦全局最优个体陷入到局部极值时,就会引导种群向局部极值靠拢,造成算法早熟,所以最优个体对整个算法的寻优精度影响很大。为了改善算法容易陷入局部最优的缺点,本文引入多项式变异策略,对全局最优个体进行变异扰动,多项式变异的数学表达式为:

$$\delta_1 = \frac{X_{\text{rabbit}} - lb}{ub - lb} \tag{17}$$

$$\delta_2 = \frac{ub - X_{\text{rabbit}}}{ub - lb} \tag{18}$$

$$\delta = \begin{cases} (2u + (1-2u)(1-\delta_1)^{\eta_m+1})^{\frac{1}{\eta_m+1}} - 1, & \text{if } u \leqslant 0.5 \\ 1 - (2(1-u) + 2(u-0.5)(1-\delta_2)^{\eta_m+1})^{\frac{1}{\eta_m+1}}, & \text{if } u > 0.5 \end{cases} \tag{19}$$

$$X'_{\text{rabbit}} = X_{\text{rabbit}} + \delta(ub - lb) \tag{20}$$

其中,u 表示取值范围为 0~1 的随机变量;η_m 表示非负数的突变因子;X_{rabbit} 表示全局最优位置;X'_{rabbit} 表示扰动后的候选个体位置。当候选个体位置的适应度优于原位置,则为有效的变异,替换原位置。否则,就是无效的变异,直接舍弃。综上通过多项式变异对最优个体扰动,减少了算法早熟的可能性,提高了算法的全局搜索能力。

2.3 变种差分策略

哈里斯鹰优化算法和大多数优化算法一样,在迭代的过程中,如果遇到局部极值的干扰,会导致种群中的个体都会快速地趋向在一起,这样会导致种群的多样性快速减小,使算法的寻优精度降低。针对这个问题,当哈里斯鹰个体位置更新公式正常执行之后,本文引入差分变异策略(差分变异策略是通过随机选中种群内部的个体进行差分和缩放达到变异的目的)的变种 *DE/rand/2*, 对种群中个体位置进行变异,产生一个新的结果,计算适应度,然后采用贪婪策略,对新位置和旧位置的适应度进行比较,保留适应度更好的位置,有效地提高了原算法的种群多样性,此处需用到的数学公式可写为:

$$X = X_{\text{old}} + (X_{r1}(t) - X_{r2}(t) + X_{r3}(t) - X_{r4}(t)) \tag{21}$$

$$X(t+1) = \begin{cases} X, & f(X) < f(X_{\text{old}}) \\ X_{\text{old}}, & f(X_{\text{old}}) < f(X) \end{cases} \tag{22}$$

其中,X_{old} 表示 HHO 算法正常执行后的个体位置,$r1$,$r2$,$r3$,$r4$ 表示种群中随机的个体。通过变种差分策略对个体进行变异,增加了整个种群的多样性。迭代前期,种群的个体之间差异较大,差值较大,导致位置改变的步长较大,有利于算法进行全局搜索,避免算法早熟。到了迭代后期,种群中的个体趋向一致,差值减小,导致位置改变的步长较小,有利于算法进行精细化探索,保证了算法的收敛性能。

2.4 DMHHO 的算法流程

步骤 1 初始化种群基本参数,例如种群数量,上下边界等参数;

步骤 2 进入迭代,计算整个种群的适应度,记录全局最优个体和适应度;

步骤 3 采用式(20)对最优个体进行变异扰动;

步骤 4 根据式(16)计算 DMHHO 算法的逃逸能量 E;

步骤 5 如果 $|E| \geqslant 1$, 则根据式(1)或者式(2)更新位置;

步骤 6 如果 $|E| \geqslant 0.5$ 且 $k \geqslant 0.5$, 则根据式(6)更新个体的位置参数。如果 $|E| \geqslant 0.5$ 且 $k < 0.5$, 则根据式(11)更新个体的位置参数。如果 $|E| < 0.5$ 且 $k \geqslant 0.5$, 则根据式(7)更新个体的位置参数。如果 $|E| < 0.5$ 且 $k < 0.5$, 则根据式(14)更新个体的位置参数;

步骤 7 采用式(22)更新个体的位置参数;

步骤 8 判断是否达到最大迭代次数,是、则输出最佳位置和最佳解,终止算法,否则返回到步骤 2。

3 实验结果和分析

3.1 实验环境和测试函数

本机的实验环境:电脑操作系统为 Win10 64 位,内存 RAM 为 16 G,处理器为 Intel(R)Core(TM) i7-10750H,仿真软件为 Matlab 2018b。

为了验证改进后的哈里斯鹰优化算法的寻优性能,选取 12 个基准测试函数,在这 12 个基准函数中,函数 $F1$ ~ $F4$ 是单峰测试函数,用于测试算法的收敛性能和局部开发能力。$F5$ ~ $F9$ 是多峰测试函

数，多峰测试函数的特点是存在多个局部极值点，容易让算法陷入局部最优，测试算法的全局探索能力和跳出局部最优能力。函数 $F10 \sim F12$ 是固定维函数，测试算法在低维问题上是否能收敛到理论最优值，测试函数的其他详细信息请参照表 1。

另外，选取灰狼优化算法（GWO）、鲸鱼优化算法（WOA）、飞蛾扑火优化算法[20]（Moth－Flame Optimization，MFO），正余弦算法（SCA）以及原始的哈里斯鹰优化算法（HHO）进行综合对比。每种算法设置种群的数量为 30，迭代次数为 500。为了减小算法随机带来的干扰，每种算法在测试函数上独立运行 30 次，对这 30 次的结果取平均值和标准差。其中，平均值用于评价算法的寻优性能，标准差用于评价算法的稳定性。

表 1　测试函数

Table 1　Test functions

函数	类型	维数	范围	最优值
$F_1(x) = \sum_{i=1}^{n} x_i^2$	单峰	30	$[-100,100]$	0
$F_2(x) = \max\{\lvert x_i \rvert, 1 \leqslant i \leqslant n\}$	单峰	30	$[-100,100]$	0
$F_3(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	单峰	30	$[-30,30]$	0
$F_4(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	单峰	30	$[-100,100]$	0
$F_5(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	多峰	30	$[-5.12,5.12]$	0
$F_6(x) = -20\exp\left(-0.2\sqrt{\left(\frac{1}{n}\sum_{i=1}^{n} x_i^2\right)}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	多峰	30	$[-32,32]$	0
$F_7(x) = \frac{1}{4\,000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	多峰	30	$[-600,600]$	0
$F_8(x) = \frac{\pi}{n}\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_{i-1})^2[1 + \sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i,10,100,4)$	多峰	30	$[-50,50]$	0
$F_9(x) = 0.1\{\sin^2(3\pi x_i) + \sum_{i=1}^{n}(x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)] + (x_i - 1)^2[1 + \sin^2(2\pi x_i)]\} + \sum_{i=1}^{n} u(x_i,5,100,4)$	多峰	30	$[-50,50]$	0
$F_{10}(x) = -\sum_{i=1}^{5}[(X - a_i)(X - a_i)^{\mathrm{T}} + c_i]^{-1}$	固定维	4	$[0,10]$	−10.153 2
$F_{11}(x) = -\sum_{i=1}^{7}[(X - a_i)(X - a_i)^{\mathrm{T}} + c_i]^{-1}$	固定维	4	$[0,10]$	−10.402 8
$F_{12}(x) = -\sum_{i=1}^{10}[(X - a_i)(X - a_i)^{\mathrm{T}} + c_i]^{-1}$	固定维	4	$[0,10]$	−10.536 3

3.2　测试函数实验结果及分析

不同算法的测试结果（$dim = 30$）见表 2。由表 2 的结果可以看出，在函数 $F1$ 和 $F2$ 上，DMHHO 算法的寻优结果达到了理论最优值，大幅度领先其他优化算法。由于单峰函数 $F3$ 和 $F4$ 比函数 $F1$ 和 $F2$ 更加地复杂，DMHHO 算法虽然没有达到理论最优值，但是寻优结果仍然领先其他的算法数个量级。从单峰测试函数的表现可以看出，DMHHO 算法具有较强的局部开发能力，DMHHO 比其他算法收敛精度更高，能快速收敛到最优值。在多峰函数 $F5 \sim F7$ 上，DMHHO 算法和 HHO 算法都达到了理论最优值，而多峰函数 $F8$ 和 $F9$ 有多个极值点且较为分

散，对算法全局探索能力和跳出局部最优的能力要求高，其他优化算法在函数 $F8$ 和 $F9$ 上，都过早地陷入局部最优中，导致寻优精度不高，但 DMHHO 算法的平均值都优于其他的算法数个量级，说明 DMHHO 的全局寻优能力和跳出局部最优能力要强于其他的算法，避免了算法早熟的情况。在固定维函数 $F10$ ~ $F12$ 上，只有 DMHHO 的寻优结果能达到理论最优值，领先于其他优化算法，说明算法在简单问题上，表现依旧出色。此外，从表 2 中的标准差可以看出，DMHHO 算法比其他优化算法的稳定性更强。综上，DMHHO 算法相对比其他的优化算法，不仅具有出色的寻优能力，还有更好的稳定性。

表 2 不同算法的测试结果

Table 2 Test results of different algorithms

函数名	指标	GWO	WOA	MFO	SCA	HHO	DMHHO
$F1$	平均值	1.41E-27	2.11E-72	1.33E+03	1.79E+01	3.07E-97	**0.00E+00**
	标准差	3.74E-27	1.14E-70	6.87E+03	3.98E+01	1.65E-96	**0.00E+00**
$F2$	平均值	1.09E-16	8.61E-51	2.98E+01	1.98E-02	1.61E-47	**0.00E+00**
	标准差	1.01E-16	2.72E-50	2.08E+01	3.38E-02	8.86E-47	**0.00E+00**
$F3$	平均值	2.69E+01	2.80E+01	1.07E+04	1.18E+05	3.05E-02	**3.19E-08**
	标准差	5.35E-01	4.24E-01	2.72E+04	2.94E+05	3.46E-02	**1.59E-07**
$F4$	平均值	6.82E-01	3.99E-01	9.73E+03	1.19E+01	3.18E-04	**2.87E-09**
	标准差	3.92E-01	1.93E-01	8.16E+03	9.98E+00	6.65E-04	**9.25E-09**
$F5$	平均值	2.80E+00	**0.00E+00**	3.39E+02	3.37E+01	**0.00E+00**	**0.00E+00**
	标准差	5.11+00	**0.00E+00**	5.48E+01	2.88E+01	**0.00E+00**	**0.00E+00**
$F6$	平均值	1.07E-13	4.44E-15	1.95E+01	1.78E+01	**8.88E-16**	**8.88E-16**
	标准差	1.84E-14	2.64E-15	5.51E-01	6.03E+00	**0.00E+00**	**0.00E+00**
$F7$	平均值	6.00E-03	**0.00E+00**	7.20E+01	9.34E-01	**0.00E+00**	**0.00E+00**
	标准差	1.06E-02	**0.00E+00**	7.57E+01	3.64E-01	**0.00E+00**	**0.00E+00**
$F8$	平均值	4.81E-02	2.17E-02	9.49E+06	2.75E+05	2.81E-06	**1.39E-08**
	标准差	2.21E-02	1.23E-02	4.67E+07	1.27E+06	2.91E-06	**3.41E-08**
$F9$	平均值	6.28E-01	5.61E-01	5.69E+07	2.47E+04	6.87E-05	**4.79E-09**
	标准差	2.15E-01	3.10E-01	1.42E+08	4.87E+04	9.57E-05	**1.83E-08**
$F10$	平均值	-9.42E+00	-7.76E+00	-6.63E+00	-1.94E+00	-5.40E+00	**-1.02E+01**
	标准差	2.25E+00	2.81E+00	3.25E+00	1.95E+00	1.29E+00	**9.81E-08**
$F11$	平均值	**-1.04E+01**	-6.98E+00	-7.66E+00	-3.84E+00	-5.42E+00	**-1.04E+01**
	标准差	9.35E-04	3.10E+00	3.45E+00	1.74E+00	1.30E+00	**2.54E-07**
$F12$	平均值	-1.03E+01	-6.60E+00	-8.47E+00	-3.50E+00	-5.46E+00	**-1.05E+01**
	标准差	1.48E+00	3.60E+00	3.25E+00	1.79E+01	1.30E+00	**7.71E-08**

3.3 收敛性曲线分析

在图 3 中，展示了单峰测试函数 $F3$ 和 $F4$，多峰测试函数 $F8$ 和 $F9$，固定维测试函数 $F11$ 和 $F12$ 的收敛曲线(种群数为 30，最大迭代次数为 500，维数 30)。从单峰测试函数和多峰测试函数的收敛曲线图可知，DMHHO 算法的收敛速度领先其他优化地算法，其他优化算法迭代前期适应度曲线能快速地下降，但是中后期曲线就趋向于平坦，停滞不前，但是 DMHHO 算法的收敛曲线在整个迭代周期都不断地下降，对比 HHO 算法和其他算法，说明 DMHHO 的收敛精度大幅度提升。对于固定维测试函数，DMHHO 算法的收敛曲线在迭代前期比其他算法要更陡峭，能快速地收敛到最优值，说明 DMHHO 算法比其他算法具有更强的收敛性能和寻优能力。综上，从收敛曲线图可以看出，改进后的算法不但加快了收敛速度，而且还大幅度提升了算法的收敛精度。

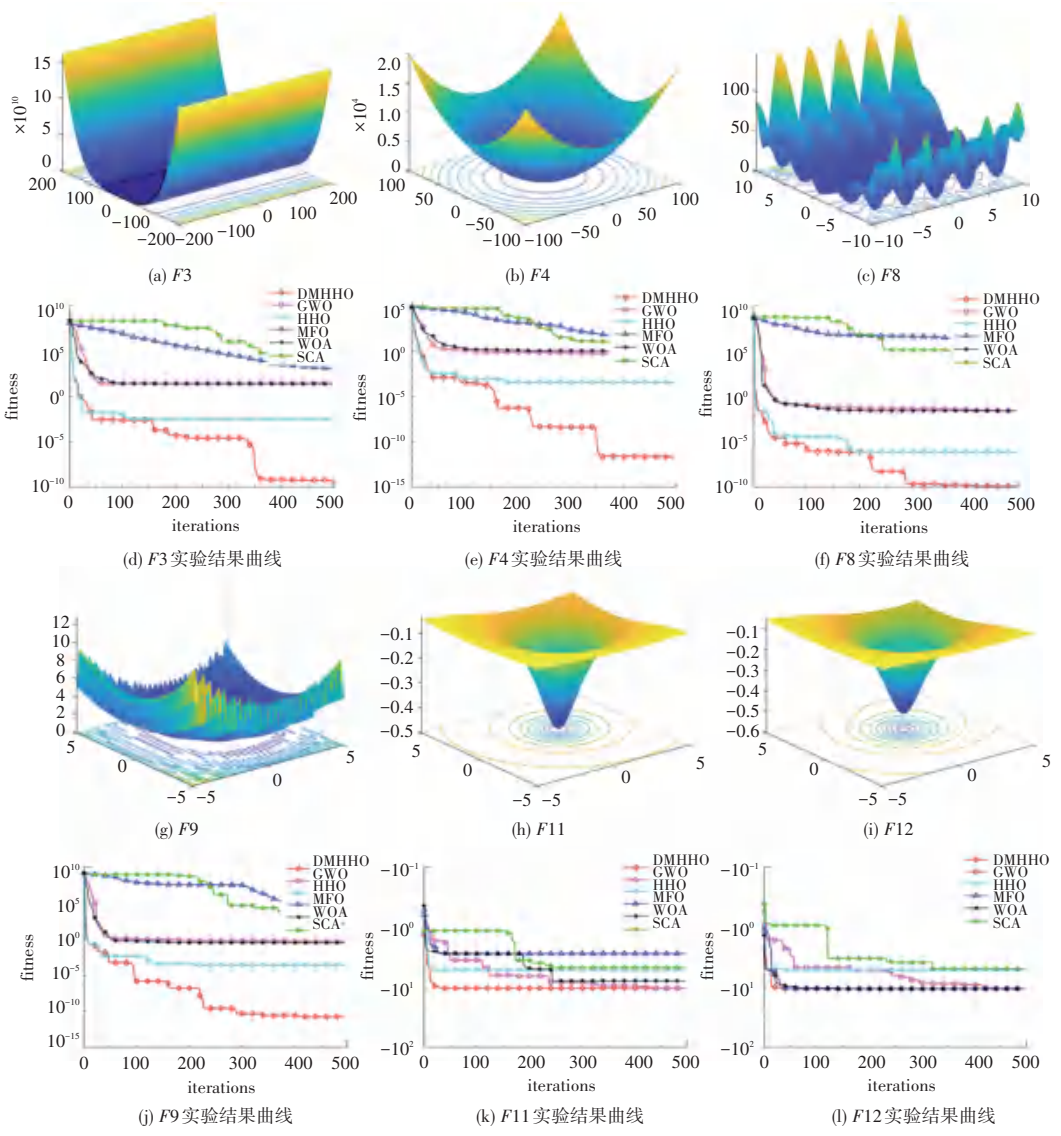

图 3　部分函数收敛曲线

Fig. 3　Partial functions convergence curves

4　改进的哈里斯鹰优化算法在工程问题中的应用

为了验证改进的哈里斯鹰优化算法在工程应用中的效果，选取了经典工程问题中的三杆桁架设计问题，三杆桁架结构如图 4 所示。将改进后的哈里斯鹰优化算法（DMHHO）应用在该问题上，并将结果与其他优化算法对比，验证 DMHHO 算法的有效性。三杆桁架设计问题一直是常见的约束问题，该问题的目标是让三杆桁架体积最小化，该设计问题有 2 个变量，分别是横截面积 $A_1(x_1)$ 和横截面积 $A_2(x_2)$。

目标函数为：

$$\min f(x) = \left(2\sqrt{2}\,x_1 + x_2\right) \times l \tag{23}$$

条件约束：

$$g_1(x) = \frac{\sqrt{2}\,x_1 + x_2}{\sqrt{2}\,x_1^2 + 2x_1x_2}P - \sigma \leqslant 0 \tag{24}$$

$$g_2(x) = \frac{x_2}{\sqrt{2}\,x_1^2 + 2x_1x_2}P - \sigma \leqslant 0 \tag{25}$$

$$g_3(x) = \frac{1}{\sqrt{2}\,x_2 + x_1}P - \sigma \leqslant 0 \tag{26}$$

其中，$l = 100\ \text{cm}$；$P = 2\ \text{kN/cm}^2$；$\sigma = 2\ \text{kN/cm}^2$；$0 \leqslant x_1, x_2 \leqslant 1$。

将 DMHHO 算法，与正余弦优化算法（SCA）、鲸鱼优化算法（WOA）、麻雀搜索算法（SSA）、哈里斯鹰优化算法（HHO）一起对三杆桁架设计问题进行求解。实验结果见表 3。分析表 3 可知，DMHHO 算法

对三杆桁架设计问题的求解效果要优于其他算法,因此 DMHHO 算法在实际的应用问题中具有一定的可行性。

表 3 不同算法求解三杆桁架设计问题实验结果

Table 3 Experimental results of different algorithms for three-bar truss design problems

算法	x_1	x_2	$f(x)$
DMHHO	0.788 87	0.407 69	**263.895 9**
HHO	0.789 44	0.406 09	263.896 3
SCA	0.787 26	0.412 37	263.907 8
WOA	0.792 49	0.397 56	263.906 4
SSA	0.789 50	0.405 92	263.896 4

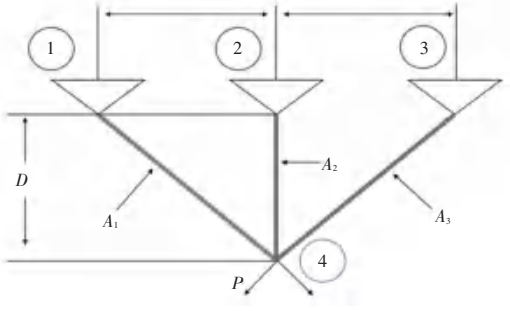

图 4 三杆桁架结构图

Fig. 4 Three-bar truss structure

5 结束语

针对哈里斯鹰优化算法收敛精度不高,容易陷入局部最优的问题,通过改进逃逸能量 E, 使算法在迭代中期以后仍然有机会执行全局探索阶段,平衡了全局搜索和局部开发能力;通过引入多项式变异对全局最优个体进行变异扰动,提高了算法跳出局部最优的能力;最后通过变种差分策略对个体更新后的位置进行变异,计算适应度,与未变异前的位置的适应度进行对比,保留适应度更好的位置,增加了种群的多样性,提高了算法的寻优能力。选取 12 个经典的测试函数,对改进后的哈里斯鹰优化算法进行寻优测试,并与 HHO 在内的 5 个优化算法进行对比,由实验结果可知,改进后的哈里斯鹰优化算法的收敛速度更快、寻优精度更高、稳定性更强。为了进一步验证算法在求解工程问题中的有效性,将改进的算法用于求解三杆桁架设计问题,实验结果表明 DMHHO 算法的改进策略有一定的优越性,接下来的工作继续提升算法的性能,并将其应用到更多的实际优化问题中。

参考文献

[1] CHEN Yi, WANG Mingjing, HEIDARI A A, et al. Multi-threshold image segmentation using a multi-strategy shuffled frog leaping algorithm[J]. Expert Systems with Applications, 2022, 194: 116511.

[2] HARIS M, ZUBAIR S. Mantaray modified multi-objective Harris hawk optimization algorithm expedites optimal load balancing in cloud computing[J]. Journal of King Saud University-Computer and Information Sciences, 2022, 34(10): 9696-9709.

[3] SU Hang, SHOU Yeqi, FU Yujie, et al. A new machine learning model for predicting severity prognosis in patients with pulmonary embolism: Study protocol from Wenzhou, China[J]. Frontiers in Neuroinformatics, 2022, 16: 1052868.

[4] 徐建中, 晏福. 改进鲸鱼优化算法在电力负荷调度中的应用[J]. 运筹与管理, 2020, 29(9): 149-159.

[5] DENNING P J. The science of computing: Genetic algorithms[J]. American Scientist, 1992, 80(1): 12-14.

[6] KENNEDY J, EBERHART R. Particle swarm optimization[C]// Proceedings of ICNN´95 - International Conference on Neural Networks. Perth, Australia :IEEE, 1995, 4: 1942-1948.

[7] STORN R, PRICE K. Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11(4): 341-359.

[8] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.

[9] MIRJALILI S, LEWIS A. The whale optimization algorithm[J]. Advances in Engineering Software, 2016, 95:1-15.

[10] MIRJALILI S. SCA: A sine cosine algorithm for solving optimization problems[J]. Knowledge-Based Systems, 2016, 96: 120-133.

[11] XUE Jiankai, SHEN Bo. A novel swarm intelligence optimization approach: sparrow search algorithm[J]. Systems Science & Control Engineering, 2020, 8(1): 22-34.

[12] HEIDARI A A, MIRJALILI S, FARIS H, et al. Harris hawks optimization: Algorithm and applications[J]. Future Generation Computer Systems, 2019, 97: 849-872.

[13] ZHANG Yang, ZHOU Xizhao, SHIH P C. Modified Harris Hawks optimization algorithm for global optimization problems[J]. Arabian Journal for Science and Engineering, 2020, 45: 10949-10974.

[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.

[15] LI Chenyang, LI Jun, CHEN Huiling, et al. Enhanced Harris hawks optimization with multi-strategy for global optimization tasks[J]. Expert Systems with Applications, 2021, 185: 115499.

[16] 聂春芳. 融合黄金正弦和随机游走的哈里斯鹰优化算法[J]. 智能计算机与应用, 2021, 11(7): 113-119,123.

[17] 李云秋, 熊瑞平, 温记明, 等. 改进哈里斯鹰优化算法求解作业车间调度问题[J]. 组合机床与自动化加工技术, 2022(11): 164-168.

[18] 孙林, 李梦梦, 徐久成. 二进制哈里斯鹰优化及其特征选择算法[J]. 计算机科学, 2023, 50(5): 277-291.

[19] 赵世杰, 高雷阜, 于冬梅, 等. 融合能量周期性递减与牛顿局部增强的改进 HHO 算法[J]. 控制与决策, 2021,36(3): 629-636.

[20] MIRJALILI S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm[J]. Knowledge-Based Systems, 2015, 89: 228-249.