

融合多策略改进的北方苍鹰算法及其应用^{*}赵 深¹ 韦根原^{1,2} 常耀华¹ 陈 亮¹ 侯彦辰³(1. 华北电力大学自动化系 保定 071003; 2. 保定市综合能源系统状态检测与优化调控重点实验室 保定 071003;
3. 广西大学机械工程学院 南宁 530004)

摘 要: 针对北方苍鹰算法寻优精度低以及容易陷入最优值等问题,提出一种融合减法优化器和 t 分布小波变异的改进北方苍鹰算法。首先,在算法初始阶段利用 Tent 映射-动态反向学习策略,提高初始种群的质量和多样性,加快算法的迭代速度;其次,在勘探阶段融合减法平均优化器和最佳值引导策略更新种群位置;最后,采用自适应 t 分布小波变异策略对种群进行扰动,避免陷入局部最优。通过测试函数仿真实验并将改进后的算法与极限学习机相结合,用于预测光伏发电量的情况,同时应用于两种工程设计问题中,实验结果表明,改进后的算法对比其他改进算法在收敛精确度和鲁棒性方面有显著提升,并且有效提升了解决复杂问题的性能。

关键词: 北方苍鹰算法;混沌映射;动态反向学习;减法平均优化器;自适应 t 分布小波变异;光伏预测

中图分类号: TP301.6;TN2 **文献标识码:** A **国家标准学科分类代码:** 520.1040

The northern goshawk algorithm integrating Multi-strategy improvement and its application

Zhao Shen¹ Wei Genyuan^{1,2} Chang Yaohua¹ Chen Liang¹ Hou Yanchen³(1. Department of Automation, North China Electric Power University, Baoding 071003, China; 2. Baoding Key Laboratory of State Detection and Optimization Regulation for Integrated Energy System, Baoding 071003, China;
3. School of Mechanical Engineering, Guangxi University, Nanning 530004, China)

Abstract: To address the issues of low optimization accuracy and the tendency to fall into local optima in the northern goshawk algorithm, an improved version is proposed that integrates the subtraction optimizer and t -distribution wavelet mutation. In the initial phase of the algorithm, the Tent map combined with the dynamic reverse learning strategy is utilized to improve the quality and diversity of the initial population, thereby accelerating the iteration speed of the algorithm. Secondly, in the exploration stage, the subtractive average optimizer and the best value guidance strategy are introduced to update the population position. Finally, an adaptive t -distribution wavelet mutation strategy is employed to perturb the population, preventing it from falling into local optima. Through simulation experiments using test functions and integrating the improved algorithm with the extreme learning machine, the approach was applied to predict photovoltaic power generation. Additionally, it was implemented in two engineering design applications. The experimental results demonstrate that the improved algorithm significantly outperforms other modified algorithms in terms of convergence accuracy and robustness, and effectively enhances the performance in solving complex problems.

Keywords: northern goshawk algorithm; chaos mapping; dynamic reverse learning; subtractive average optimizer; adaptive t -distribution wavelet variation; PV forecasting

0 引言

随着科学与技术持续进步,实际应用中遭遇到的问题和优化挑战变得更为复杂和庞大,以至于传统的优化方法不再适应或无法满足高精度处理需求,受自然界生物习性

的启发,学者们提出许多群体智能算法,如人工蜂鸟算法(artificial hummingbird algorithm, AHA)^[1], 斑马优化算法(zebra optimization algorithm, ZOA)^[2], 白鲸优化算法(beluga whale optimization, BWO)^[3], 蜣螂优化算法(dung beetle optimizer, DBO)^[4] 和金豺优化算法(golden jackal

optimization, GJO)^[5]等,这些优化算法的为解决复杂问题提供了良好的思路,在工程优化和机器学习等领域得到了广泛应用。

北方苍鹰优化算法(northern goshawk optimization, NGO)是一种新型智能算法,受苍鹰的捕猎行为启发而设计。相比于其他智能算法,北方苍鹰算法具有操控简单,易于实现等优点,已经成功应用到数据预测^[6]、故障识别^[7]、参数优化^[8]等领域,并表现出色,例如,文献[9]提出通过引入 Logistic-tent 混沌映射、Cauchy 变异算子和非线性递增权重 3 种策略对北方苍鹰优化算法进行了改进,结合极限学习机构成组合模型,用于变压器故障诊断,该模型在一定程度上提高了故障诊断精度,但 Logistic-tent 混沌映射存在高维空间中出现遍历性不足的问题,导致种群初始化的多样性受限,影响算法在复杂问题中的全局搜索能力;文献[10]提出采用了一种结合立方混沌优化与透镜成像反向学习的策略来进行种群初始化,从而达到增强种群多样性和提升算法搜索性能的效果。此外,采用了最优最差反向学习策略与透镜成像反向学习方法,以防止算法陷入局部最优,但该策略在引导种群向最优解靠近的过程中,缺乏对解空间的动态适应性,可能导致算法在搜索后期陷入停滞;文献[11]提出利用 tent 映射初始化种群,增强全局搜索能力,使用自适应权重和黄金正弦算法相结合来平衡局部与全局开发的能力,并引入非线性因子进行扰动提升局部优化能力,但自适应权重和黄金正弦算法的结合在实际应用中,权重调整的时机和幅度难以精准把握,可能无法有效平衡局部和全局搜索;文献[12]提出了一种结合混沌优化与莱维飞行中的贪婪策略的方法,以提升算法的搜索能力,并有效避免陷入局部最优解的问题,但在面对高维复杂问题时,寻优效果依然有待提高。

针对上述对于改进北方苍鹰算法后依旧存在的问题,本文提出一种融合多策略改进北方苍鹰算法(STWNGO)。当 NGO 算法在初始阶段时,在 Tent 映射的基础上融合动态反向学习策略,能更灵活地筛选出适应度值小的个体,有效提升种群质量和多样性,加快收敛过程提升了种群的质量;在勘探阶段引入融合减法平均优化器和最佳值引导策略,相较于其他改进策略,两种搜索方式结合,在增强局部搜索能力和提高全局搜索能力的同时,能够直接选定最佳狩猎位置,大大提高收敛速度;在开发阶段后期,为了防止 NGO 算法陷入局部最优解中,采用自适应 t 分布小波变异策略进行扰动,提高了整体的寻优精度。最后应用在光伏预测及两个工程设计问题中,进一步证明本文所改进算法具有明显的优势。

1 北方苍鹰算法

NGO^[13]的执行过程主要包括两个阶段:一是猎物识别阶段(勘探阶段),二是追逐与逃生阶段(开发阶段)。前者在成功识别到猎物后会以极快的速度向猎物发起攻击,后

者则在开发阶段进行追逐和逃生。

1.1 猎物识别阶段

在初始阶段,北方苍鹰会随机选择一个猎物目标,并立即展开攻击,其位置更新公式如下:

$$x_{i,j}^{new,P_1} = \begin{cases} x_{i,j} + r(p_{i,j} - Ix_{i,j}), & F_{p_i} < F_i \\ x_{i,j} + r(x_{i,j} - p_{i,j}), & F_{p_i} \geq F_i \end{cases} \quad (1)$$

$$X_i = \begin{cases} X_i^{new,P_1}, & F_i^{new,P_1} < F_i \\ X_i, & F_i^{new,P_1} \geq F_i \end{cases} \quad (2)$$

$$P_i = Xk, i = 1, 2, \dots, N, k = 1, 2, \dots, i-1, i+1, \dots, N \quad (3)$$

式中: P_i 表示第 i 个北方苍鹰的猎物位置; F_{p_i} 是算法的适应度值; $p_{i,j}$ 为第 i 个北方苍鹰在第 j 维维所处的猎物位置,其中 k 是在 $[1, N]$ 范围内的随机整数; N 为北方苍鹰的总数, X_i^{new,P_1} 表示第 i 个北方苍鹰的新位置; $x_{i,j}^{new,P_1}$ 是第 i 个北方苍鹰在第 j 维的更新位置; F_i^{new,P_1} 表示在第一阶段(P_1)更新后的第 i 个北方苍鹰的目标函数的数值; P_1 代表北方苍鹰的第一阶段, r 为 $[1, 2]$ 范围内的随机数; I 是 1 或 2 的随机整数。

1.2 追逐与逃生阶段

在第二阶段,抓捕猎物时,猎物会逃跑,但北方苍鹰速度很快,能在大多数情况下猎取猎物,该过程的数学表达式如下:

$$X_{i,j}^{new,P_2} = x_{i,j} + R(2r - 1)x_{i,j} \quad (4)$$

$$R = 0.02(1 - \frac{t}{T}) \quad (5)$$

$$X_i = \begin{cases} X_i^{new,P_2}, & F_i^{new,P_2} < F_i \\ X_i, & F_i^{new,P_2} \geq F_i \end{cases} \quad (6)$$

式中: R 代表攻击半径,随着迭代次数的增加而逐渐减小; T 是最大迭代次数; t 表示当前迭代次数; X_i^{new,P_2} 是第 i 个北方苍鹰在第二阶段(P_2)的状态,其第 j 维为 $x_{i,j}^{new,P_2}$; F_i^{new,P_2} 在第二阶段(P_2)完成更新后的目标函数值。

2 多策略改进北方苍鹰算法

2.1 Tent 映射-动态反向学习策略

NGO 初始化种群的质量,一定程度上影响算法的寻优能力,在解决寻优问题时,增加种群的多样性,提高种群的质量,有利于加快算法逼近最优值的迭代速度。因此引入 Tent 映射-动态反向学习策略来改善这一问题。

Tent 映射^[14]作为混沌映射的一种,具有遍历性,使得种群遍布更加均匀,被广泛应用到算法的种群初始化中,如式(7)所示。

$$x_{i+1} = \begin{cases} x_i/\alpha, & x_i \in [0, \alpha) \\ (1-x_i)/(1-\alpha), & x_i \in [\alpha, 1) \end{cases} \quad (7)$$

式中: α 为混沌系数, i 为混沌迭代次数。

为了提高种群的质量,加快向最优值逼近的速度,又引入反向学习策略^[15],由一个初始个体得到一个反向个体,如式(8)所示。

$$X_{i+1} = U_b + L_b - X_i \quad (8)$$

式中: L_b 为边界的最小值, U_b 为边界的最大值, X_i 为 Tent 映射产生的初始个体。但反向学习策略由于产生的解与当前解的距离固定,缺乏一定的随机性,种群的多样性无法得到有效增加,因此在此基础上引入动态因子,如式(9)所示。

$$X_{i+1} = X_i + r_1 \times (r_2 \times (U_b + L_b - X_i) - X_i) \quad (9)$$

式中: r_1 和 r_2 是取值范围在(0,1)之间的随机数, X_{i+1} 则是获得的动态反向个体。综上,Tent 映射-动态反向学习策略的具体流程为:

1)首先利用 Tent 映射生成 N 个初始种群个体,然后采用动态反向学习策略,生成 N 个相应的反向种群个体。

2)然后通过这两种方法得到两组对应种群的适应度值,进行排序,取适应度值最小的前 N 个数据作为新种群,从而获得初始种群。

该策略通过引用 Tent 映射提高种群的遍历性,使得种群分布更加均匀,同时加入动态反向学习策略,在种群中挑选适应度值最小的 N 个个体进行后续算法的迭代寻优,极大地缩短了因种群质量低而导致的寻优时间长的的问题。

2.2 融合减法平均优化器和最佳值引导策略

在 NGO 的勘探阶段,虽然随机的种群的捕食行为可以增加种群勘探能力,但是随机选取捕食位置会造成算法收敛速度过慢同时影响寻优结果的精确度,为了解决该问题引入减法平均优化器和最佳值引导策略。

当 $F_{p_i} < F_i$ 时,种群中随机选取的狩猎位置变为最佳狩猎位置,以达到更快的捕食速度,避免狩猎的盲目性,提高狩猎效率,加大算法的收敛速度,增强寻优结果的精确度,改进后的公式如下:

$$x_{i,j}^{new} = x_{i,j} + r \times (xbest - Ix_{i,j})F_{p_i} < F_i \quad (10)$$

式中: r 为随机数, $xbest$ 为每一代的最优值, I 为 1 或 2 的随机整数。

当 $F_{p_i} \geq F_i$ 时,引入减法优化器算法,使用累积种群位置差值的算术平均值,而不是仅仅使用某一种群的位置的差值来更新,避免因种群位置突变而造成算法陷入局部最优的情况,因此更新公式如下:

$$x_{i,j}^{new} = x_{i,j} + \vec{r} \times \frac{1}{N} \sum_{k=1}^N (X_i - {}_vX_k)F_{p_i} \geq F_i \quad (11)$$

式中: N 为个体总数, v 为一个维度为 m 的向量, X_i 和 ${}_vX_k$ 为北方苍鹰的位置。将最佳引导策略和减法优化器策略相结合得到北方苍鹰算法勘探阶段的位置即式(1)更新为:

$$x_{i,j}^{new,P_1} = \begin{cases} x_{i,j} + r \times (xbest - Ix_{i,j}), F_{p_i} < F_i \\ x_{i,j} + \vec{r} \times \frac{1}{N} \sum_{k=1}^N (X_i - {}_vX_k), F_{p_i} \geq F_i \end{cases} \quad (12)$$

在勘探阶段,采用最佳值引导策略,直接选定最佳狩猎位置代替随机选取的狩猎位置,提高算法的收敛速度和寻优精确度,同时又加入了减法优化器策略,通过对累积种群位置的平均值来更新,避免陷入局部最优,两种方式的结合,丰富了苍鹰的狩猎方式,提高了在勘探阶段北方苍鹰在解空间的遍历性,加快了算法的收敛速度和寻优能力。

2.3 自适应 t 分布小波变异

在开发阶段后期,北方苍鹰容易因捕食相同猎物导致种群范围逐渐收缩,从而陷入局部最优,导致算法的精确程度降低。因此采用自适应 t 分布小波变异来改善这一问题。其中自适应 t 分布^[16]结合了高斯分布与柯西分布各自的特点,既具有较好的局部开发的能力,又具有出色的全局搜索能力,能够很好克服局部最优的问题。而 Morlet 小波变异,产生正数和负数的机率相同,故其更易获得最优解。将 t 分布策略与小波变异策略结合,在 NGO 开发阶段后期,对种群位置进行扰动,可以提高种群多样性,提升收敛速度,避免陷入局部最优。

在开发阶段后期,当 $p > 0.5$ 时选取种群所在最优位置进行自适应 t 分布小波变异,当 $p < 0.5$ 时选取当前位置进行更新公式如下:

$$x_{i,j} = \begin{cases} \gamma \times xbest + xbest \times t(ite), rand < p \\ x_{i,j}, rand > p \end{cases} \quad (13)$$

式中: γ 为小波函数值, $x_{i,j}$ 为更新后的新个体, $xbest$ 为全局最优个体, $t(ite)$ 是将迭代次数 ite 作为参数自由度的 t 分布,小波函数值公式如下:

$$\gamma = \frac{1}{\sqrt{\beta}} \exp\left(-\left(\frac{\phi}{\beta}\right)^2\right) \cos\left(5\left(\frac{\phi}{\beta}\right)\right) \quad (14)$$

式中: ϕ 的可行域为 $[-2.5\beta, 2.5\beta]$, β 的公式为:

$$\beta = \exp(-a \times (1 - \frac{t}{T_{max}})^\eta) + a \quad (15)$$

式中: $a=4$, η 表示形状参数, $\eta=5$ 。

在算法开发后期引入自适应 t 分布小波变异。在迭代前期,t 分布变异近似柯西变异,对单个种群位置具有较强的干扰能力,可以增加种群多样性,避免陷入局部最优;在迭代中期 t 分布变异结合柯西分布和高斯分布的优点,具有平衡全局和局部搜索的能力;在迭代后期,t 分布变异近似于高斯分布,进一步增强了对种群局部开发的能力,在 t 分布变异中加入小波函数,可以提高整体寻优精确度,增加算法的收敛速度。

2.4 STWNGO 实现步骤

STWNGO 实现步骤如下:

1)初始化 STWNGO 的参数时,需要设置种群规模 N 、空间维度 D 和最大迭代次数 T 。

2)根据式(7)利用混沌映射得到种群,通过式(9)对种群进行更新,将更新前后的两个群对应的适应度值进行排列,选取前 n 个最优适应度值的种群,作为初始

种群。

3)根据式(12),更新北方苍鹰勘探阶段的种群位置,并进行边界检验。

4)根据式(13)随机选择自适应 t 分布策略来扰动北方苍鹰的位置,并检验是否超出搜索边界,计算适应度值并对比保存最优解。

5)判断是否满足最大迭代次数,不满足则返回步骤 3),满足则输出最后结果。

STWNGO 流程如图 1 所示。

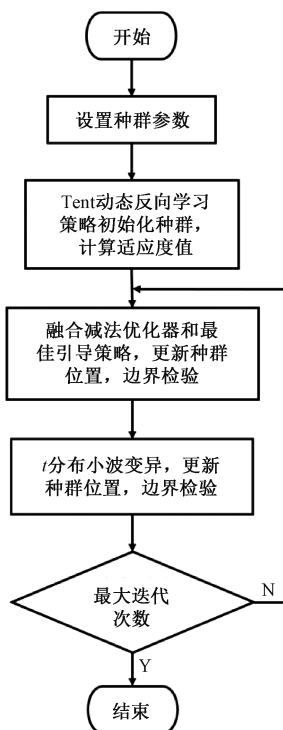


图 1 STWNGO 算法流程

Fig. 1 Flow chart of the STWNGO

2.5 时间复杂度分析

时间复杂度是检验算法运算效率的一个重要指标,设北方苍鹰种群规模为 N ,空间维度为 D ,最大迭代次数为 T ,对于 STWNGO 的时间复杂度如下:

在算法初始化阶段,设定参数初始化阶段的时间为 o_1 ,Tent 映射产生一个种群所需时间为 o_2 ,动态反向学习中生成一个反向个体所需的时间为 o_3 ,设找到当前最佳个体时间为 o_4 ,种群空间维度为 D 的个体目标函数计算时间为 $f(D)$,因此,在 STWNGO 的初始化过程中,其时间复杂度可以用以下公式表示:

$$T_1 = O(o_1 + No_2 + Nf(D) + No_3 + o_4) \quad (16)$$

在算法迭代阶段,设采用融合减法平均优化器和最佳值引导策略所用时间为 o_5 ,采用自适应 t 分布小波变异更新当前解的时间为 o_6 ,则 STWNGO 每次迭代的时间复杂度如式(17)所示。

$$T_2 = O(N(o_5 + o_6 + f(D))) = O(f(D)) \quad (17)$$

所以,STWNGO 的总的时间复杂度如公式如式(18)所示。

$$T(D) = T_1 + T \times T_2 = O(D + f(D)) \quad (18)$$

可以发现,STWNGO 的时间复杂度与 NGO 的时间复杂度的数量级相同,效率并没有降低。

在文献[17]中,多策略协同优化的北方苍鹰优化算法(northern goshawk optimization algorithm for multi-strategy co-optimization,BLNGO)的时间复杂度,设定参数初始化阶段的时间为 o_7 ,最优个体引导策略中产生一个个体的时间为 o_8 ,种群空间维度为 D 的个体目标函数计算时间为 $f(D)$,因此,在 BLNGO 算法的初始化过程中,其时间复杂度可以用式(19)表示。

$$T_3 = O(o_7 + No_8 + Nf(D)) \quad (19)$$

在算法迭代阶段设引入增强莱维飞行策略所用时间为 o_9 ,采用小孔成像策略更新当前解的时间为 o_{10} ,则 BLNGO 算法每次迭代的时间复杂度如式(20)所示。

$$T_4 = O(N(o_9 + o_{10} + f(D))) = O(f(D)) \quad (20)$$

所以,BLNGO 算法的总的时间复杂度如式(21)所示。

$$T(D) = T_3 + T \times T_4 = O(D + f(D)) \quad (21)$$

可以发现,STWNGO 的时间复杂度与 BLNGO 的时间复杂度的数量级相同,效率并没有降低。

3 仿真实验与结果分析

3.1 实验设计及测试函数

为验证 STWNGO 性能的可靠性,共设计了 5 组实验:实验 1 为不同单一策略对于 NGO 改进有效性分析;实验 2 为与 NGO 及其他改进 NGO 进行对比;实验 3 为与其他智能优化算法在不同维度的对比及在低维度收敛性分析;实验 4 为 Wilcoxon 秩和检验;实验 5 为 CEC2017 复杂函数测试。

本文在 MATLAB2022b 软件环境中进行仿真测试。为了验证本文提出的优化算法性能,实验中将种群规模设为 $N=30$,最大迭代次数为 $T=500$,测试仿真采用 12 个基准函数,具体如表 1 所示,其中 F1~F5 是单峰测试函数,F6~F10 属于多峰测试函数,F11~F12 为固定维度测试函数。

3.2 单一策略改进算法的有效性分析

为了验证本文所提出的不同单一改进策略对算法性能的影响程度,将只引入 Tent 映射-动态反向学习策略的 NGO 记为 NGO1,只引入融合减法平均优化器和引入最佳值引导策略的 NGO 记为 NGO2,只引入自适应 t 分布小波变异的 NGO 记为 NGO3,空间维度为 30,迭代次数为 500,函数测试实验结果如表 2 所示。

由表 2 可以看出,NGO1 在测试函数中表现良好,在 F1~F4 单峰测试函数中,算法精确度优于 NGO,在 F7、F8 多峰测试函数中,依然优于 NGO,该结果体现了 Tent 映

表 1 基准测试函数

Table 1 Benchmark functions

函数	函数名	范围	维度	最优值
F1	Sphere	$[-100,100]$	30/100/300	0
F2	Schwefel2.22	$[-10,10]$	30/100/300	0
F3	Schwefel1.2	$[-100,100]$	30/100/300	0
F4	Schwefel2.21	$[-100,100]$	30/100/300	0
F5	Rosenbrock	$[-30,30]$	30/100/300	0
F6	Step	$[-100,100]$	30/100/300	0
F7	Quartic	$[-1.28,1.28]$	30/100/300	0
F8	Schwefel2.26	$[-500,500]$	30/100/300	-418.98D
F9	Rastrigin	$[-5.12,5.12]$	30/100/300	0
F10	Ackley	$[-32.32]$	30/100/300	0
F11	Kowalik	$[-5,5]$	4	0.0003
F12	Hartman's family2	$[0,1]$	6	-3.32

射-动态反向学习策略,有效提高种群的多样性和质量,从而强化算法的局部搜索效果;NGO2 在单峰测试函数中改进效果要好于 NGO1,在多峰测试函数 F8 中,虽然测试效果不如 NGO1,但是精确程度要好于 NGO,此外,在其他多峰测试函数中性能指标好于 NGO1,因此可以看出融合减法平均优化器和引入最佳值引导策略在一定程度上可以扩大搜索空间,提高算法的搜索能力;NGO3 在 F1~F4 测试函数性能指标中比标准 NGO 提高上百个数量级并且在 F7、F8、F10 中仍取得很好的测试结果,证明该改进策略可以有效提高算法的精确度,而 STWNGO 算法的指标在基准函数上基本均优于标准 NGO 和其他单一改进策略,表明在融合所有改进策略之后算法的精确程度得到提升,同时全局搜索能力及跳出局部最优解的能力也得到提升;在固定维度函数 F11、F12 中,改进后的算法各有优势,但数值差距不太明显。综上分析,改进后的 NGO 要好于标准 NGO,同时将各个改进策略融合在一起,整体性能要好于单一改进策略,验证了 STWNGO 算法的优越性。

表 2 单一改进策略的算法测试结果

Table 2 Algorithm test results of a single improvement strategy

函数	算法	std	avg	函数	算法	std	avg	函数	算法	std	avg
F1	NGO	1.50×10^{-87}	7.50×10^{-88}	F5	NGO	4.02×10^{-1}	2.59×10^1	F9	NGO	0	0
	NGO1	1.03×10^{-87}	1.65×10^{-91}		NGO1	4.23×10^{-1}	2.49×10^{-1}		NGO1	9.11×10^{-32}	2.98×10^{-33}
	NGO2	9.79×10^{-94}	2.19×10^{-94}		NGO2	2.71×10^{-1}	2.40×10^{-1}		NGO2	0	0
	NGO3	0	3.35×10^{-229}		NGO3	8.30×10^{-1}	2.80×10^{-1}		NGO3	0	0
	STWNGO	0	0		STWNGO	1.98×10^{-2}	2.37×10^{-2}		STWNGO	0	0
F2	NGO	1.18×10^{-45}	1.34×10^{-45}	F6	NGO	7.84×10^{-4}	2.94×10^{-4}	F10	NGO	2.79×10^{-15}	7.41×10^{-15}
	NGO1	1.03×10^{-45}	8.05×10^{-46}		NGO1	4.61×10^{-2}	8.55×10^{-3}		NGO1	1.87×10^{-15}	6.01×10^{-15}
	NGO2	3.99×10^{-50}	4.08×10^{-50}		NGO2	4.91×10^{-6}	2.45×10^{-6}		NGO2	0	4.00×10^{-15}
	NGO3	2.12×10^{-117}	1.24×10^{-117}		NGO3	5.73×10^{-1}	2.63×10^0		NGO3	1.70×10^{-15}	2.81×10^{-15}
	STWNGO	0	0		STWNGO	8.15×10^{-20}	5.66×10^{-18}		STWNGO	0	4.44×10^{-16}
F3	NGO	4.12×10^{-21}	7.68×10^{-22}	F7	NGO	3.68×10^{-4}	5.80×10^{-4}	F11	NGO	0	7.81×10^{-4}
	NGO1	5.06×10^{-24}	1.81×10^{-24}		NGO1	2.71×10^{-4}	3.10×10^{-4}		NGO1	0	3.63×10^{-4}
	NGO2	1.32×10^{-25}	2.53×10^{-25}		NGO2	2.31×10^{-4}	6.90×10^{-5}		NGO2	0	3.70×10^{-4}
	NGO3	0	1.01×10^{-208}		NGO3	6.19×10^{-5}	1.16×10^{-5}		NGO3	0	3.94×10^{-4}
	STWNGO	0	0		STWNGO	4.48×10^{-5}	4.28×10^{-6}		STWNGO	0	3.18×10^{-4}
F4	NGO	3.45×10^{-37}	2.24×10^{-37}	F8	NGO	4.36×10^{-2}	-7.51×10^3	F12	NGO	4.12×10^{-2}	-2.15×10^0
	NGO1	7.75×10^{-38}	8.28×10^{-38}		NGO1	3.67×10^{-2}	-7.80×10^3		NGO1	3.32×10^{-2}	-2.31×10^0
	NGO2	5.17×10^{-39}	5.07×10^{-39}		NGO2	3.72×10^{-2}	-7.67×10^3		NGO2	1.13×10^{-2}	-2.91×10^0
	NGO3	3.60×10^{-112}	1.14×10^{-112}		NGO3	1.52×10^{-2}	-7.93×10^3		NGO3	2.16×10^{-2}	-3.31×10^0
	STWNGO	0	0		STWNGO	3.29×10^{-3}	-1.25×10^4		STWNGO	1.12×10^{-2}	-3.32×10^0

3.3 与不同改进 NGO 算法整体性分析

为了验证 STWNGO 相较于其他文献中对 NGO 的改进方法的优越性,本文选取了多个改进 NGO 算法进行比较。具体而言,将文献[11]中的改进 NGO 方法定义为 LNGO,文献[17]的改进方法为 BLONGO,文献[18]的改进定义为 ENGO,而文献[19]中的改进 NGO 方法定义为 MENG0。在空间维度为 30 的情况下,本文对上述算法进

行了性能对比实验,并采用 30 次独立运行所得的最佳值作为实验结果,具体数据如表 3 所示。

由表 3 可以看出,本文改进后的算法对比其他改进算法具有明显的优越性,在测试函数 F1~F6、F9、F10 能够收敛到最优值,反映了该算法在求解质量和稳定性上的平衡,在测试函数 F7 中虽然在标准差中次于 ENGO,但在平均值中要优于所对比的智能算法,整体展现出较好的寻优稳定性。

在固定维度的测试函数 F11 和 F12 上,STWNGO 的优化能力依然优于其他对比算法,保持最佳表现。因此,对比

各群智能算法在所测试的基准函数中,STWNGO 具有更好的寻优精度,同时在寻优过程中表现出更好的稳定性。

表 3 STWNGO 与其他改进方法对比结果

Table 3 Comparison results of STWNGO with other improved methods

函数	算法	std	avg	函数	算法	std	avg	函数	算法	std	avg
F1	NGO	1.19×10^{-86}	3.17×10^{-87}	F5	NGO	3.84×10^{-1}	2.57×10^1	F9	NGO	0	0
	ENG0	4.61×10^{-311}	1.26×10^{-310}		ENG0	4.13×10^{-1}	2.58×10^1		ENG0	0	0
	LNG0	6.73×10^{-241}	5.36×10^{-255}		LNG0	4.38×10^{-1}	2.83×10^1		LNG0	0	0
	MENG0	5.73×10^{-364}	8.36×10^{-342}		MENG0	4.90×10^{-2}	3.48×10^{-2}		MENG0	0	0
	BLONG0	0	0		BLONG0	3.81×10^{-2}	2.58×10^{-2}		BLONG0	0	0
	STWNG0	0	0		STWNG0	2.80×10^{-2}	2.48×10^{-2}		STWNG0	0	0
F2	NGO	1.80×10^{-45}	1.51×10^{-45}	F6	NGO	2.50×10^{-3}	9.79×10^{-4}	F10	NGO	1.77×10^{-15}	6.13×10^{-15}
	ENG0	0	8.68×10^{-88}		ENG0	3.50×10^{-4}	3.66×10^{-4}		ENG0	6.49×10^{-16}	3.88×10^{-15}
	LNG0	0	6.66×10^{-70}		LNG0	5.53×10^{-1}	3.80×10^0		LNG0	0	4.44×10^{-16}
	MENG0	1.24×10^{-112}	6.66×10^{-98}		MENG0	1.46×10^{-10}	1.64×10^{-7}		MENG0	1.03×10^{-16}	4.44×10^{-16}
	BLONG0	4.24×10^{-89}	3.66×10^{-90}		BLONG0	2.43×10^{-20}	1.83×10^{-16}		BLONG0	3.27×10^{-16}	4.84×10^{-16}
	STWNG0	0	0		STWNG0	1.48×10^{-21}	1.53×10^{-16}		STWNG0	0	3.44×10^{-16}
F3	NGO	2.76×10^{-22}	1.17×10^{-22}	F7	NGO	2.96×10^{-4}	5.20×10^{-4}	F11	NGO	3.18×10^{-4}	3.18×10^{-4}
	ENG0	0	0		ENG0	3.12×10^{-6}	4.36×10^{-5}		ENG0	6.05×10^{-4}	3.97×10^{-4}
	LNG0	1.62×10^{-315}	7.83×10^{-312}		LNG0	6.43×10^{-5}	7.89×10^{-5}		LNG0	3.57×10^{-4}	3.41×10^{-4}
	MENG0	0	0		MENG0	1.36×10^{-5}	6.38×10^{-5}		MENG0	3.26×10^{-4}	3.78×10^{-4}
	BLONG0	2.14×10^{-325}	2.14×10^{-315}		BLONG0	2.65×10^{-5}	3.45×10^{-5}		BLONG0	3.16×10^{-4}	3.58×10^{-4}
	STWNG0	0	0		STWNG0	3.78×10^{-6}	3.16×10^{-5}		STWNG0	3.08×10^{-4}	3.12×10^{-4}
F4	NGO	1.52×10^{-37}	1.67×10^{-37}	F8	NGO	5.72×10^{-2}	-7.66×10^3	F12	NGO	3.12×10^{-2}	-3.31×10^0
	ENG0	2.03×10^{-95}	4.61×10^{-99}		ENG0	3.59×10^{-2}	-7.88×10^3		ENG0	3.29×10^{-2}	-2.46×10^{-2}
	LNG0	8.14×10^{-84}	9.16×10^{-90}		LNG0	8.58×10^{-2}	-5.14×10^3		LNG0	3.02×10^{-2}	-3.21×10^0
	MENG0	0	0		MENG0	4.56×10^{-2}	-8.08×10^3		MENG0	3.02×10^{-2}	-3.21×10^0
	BLONG0	2.23×10^{-102}	4.61×10^{-102}		BLONG0	8.58×10^{-2}	-5.06×10^3		BLONG0	3.10×10^{-2}	-3.26×10^0
	STWNG0	0	0		STWNG0	5.23×10^{-2}	-1.24×10^4		STWNG0	3.01×10^{-2}	-3.32×10^0

3.4 不同智能算法的寻优性分析

为了验证所改算法 STWNGO 在不同维度下的求解效果,将其与粒子群算法 (particle swarm optimization, PSO)^[20]、NGO、GJO、改进粒子速度和位置更新公式的粒子群优化算法 (particle swarm optimization algorithm with improved particle velocity and position update formula, IPSOVP)^[21]、混合策略改进的金豺优化算法 (improved golden jackal optimization, IGJO)^[22]、基于蜂鸟飞行模式的改进粒子群优化算法 (modified particle swarm optimization algorithm based on hummingbird flight patterns, PSOHBF)^[23] 做性能对比。将各算法独立运行 30 次,取 30 次中效果最好的一组标准差、平均值作为评价指标,相关参数按照对应文献设置,实验结果如表 4、5 所示。

由表 4、5 可以看出,STWNGO 对比其他算法,它的整体寻优精度和稳定性要更加出色。当维度相同,比较不同算法时,在维度 30 的情况下,在单峰测试函数,只有测试函数 F5,IGJO 的性能指标好于 STWNG,其他测试函数中

STWNGO 均好于对比函数,甚至大多都达到理想最优,同样在多峰测试函数 F8 中,STWNGO 的平均值略差于 IGJO,但稳定性要好于 IGJO,在测试函数 F6、F7、F9、F10 中,STWNGO 的寻优精度好于其他算法,表明本文改进后的算法能够很好的逼近理想值,能够有效克服局部最优问题。当考虑算法相同,观察不同维度时,可以看出,大多数算法随着维度的增加,其寻优性能发生程度不同地下降,但 STWNGO 的下降浮动最小,在 F1、F2、F4、F9 中依然能够保持收敛值达到全局最优的状态。300 维度时,10 个测试函数中,PSOHBF 的平均值指标只有在测试函数 F8 中优于对比算法,而在其他测试函数中,STWNGO 都强势达到最优效果,表明该算法对于维度变化适应力强,在高维度中依然具有很好的寻优效果。在固定维度函数中,STWNGO 各项指标依然好于其他算法。总体来看,STWNGO 不管是在低维度测试函数中,还是高维度测试函数中,其寻优性能都好于其对比算法,改进后的算法展现了更优的性能。

表 4 不同算法在不同维度测试函数对比结果

Table 4 Comparison results of different algorithms in different dimensions

函数	算法	30D		100D		300D	
		std	avg	std	avg	std	avg
F1	PSO	4.43×10^{-10}	8.35×10^{-10}	5.19×10^{-1}	8.10×10^{-1}	4.84×10^2	8.83×10^2
	NGO	1.12×10^{-87}	8.17×10^{-88}	1.61×10^{-82}	7.94×10^{-83}	3.06×10^{-81}	2.17×10^{-81}
	GJO	1.34×10^{-55}	7.08×10^{-56}	6.16×10^{-27}	6.04×10^{-27}	5.57×10^{-13}	4.21×10^{-13}
	IPSOVP	0	0	0	0	0	0
	IGJO	0	1.00×10^{-159}	0	2.55×10^{-248}	0	1.99×10^{-158}
	PSOHBFB	8.65×10^{-202}	8.35×10^{-103}	8.65×10^{-152}	6.35×10^{-103}	0	0
	STWNGO	0	0	0	0	0	0
F2	PSO	6.50×10^{-8}	9.19×10^{-8}	7.68×10^{-1}	3.01×10^{-1}	1.49×10^{10}	2.91×10^8
	NGO	3.65×10^{-45}	1.90×10^{-45}	2.96×10^{-43}	1.98×10^{-43}	1.26×10^{-42}	1.80×10^{-42}
	GJO	1.38×10^{-33}	9.36×10^{-34}	1.83×10^{-18}	2.88×10^{-18}	2.60×10^{-11}	4.56×10^{-11}
	IPSOVP	0	7.72×10^{-187}	0	6.74×10^{-201}	0	1.11×10^{-213}
	IGJO	7.04×10^{-135}	1.30×10^{-135}	5.78×10^{-130}	1.11×10^{-130}	8.50×10^{-62}	1.60×10^{-132}
	PSOHBFB	0	0	0	0	0	0
	STWNGO	0	0	0	0	0	0
F3	PSO	2.56×10^{-1}	4.76×10^{-1}	2.44×10^1	4.55×10^1	1.05×10^4	7.91×10^5
	NGO	1.36×10^{-22}	5.42×10^{-23}	1.11×10^{-8}	2.11×10^{-9}	8.72×10^{-5}	1.86×10^{-5}
	GJO	2.80×10^{-5}	6.00×10^{-06}	1.84×10^3	1.87×10^3	4.52×10^4	1.76×10^5
	IPSOVP	0	0	0	0	6.36×10^{-105}	0
	IGJO	0	4.81×10^{-94}	0	1.44×10^{-93}	0	4.60×10^{-111}
	PSOHBFB	1.43×10^{-215}	6.52×10^{-263}	6.42×10^{-205}	3.58×10^{-183}	8.53×10^{-124}	5.42×10^{-103}
	STWNGO	0	0	0	0	5.45×10^{-312}	5.45×10^{-301}
F4	PSO	2.69×10^{-8}	4.92×10^{-8}	1.70×10^0	6.10×10^0	5.55×10^0	1.01×10^1
	NGO	2.22×10^{-37}	1.36×10^{-37}	2.71×10^{-34}	4.06×10^{-34}	4.28×10^{-32}	3.01×10^{-32}
	GJO	3.32×10^{-10}	9.50×10^{-11}	1.21×10^1	2.80×10^1	4.35×10^0	7.69×10^1
	IPSOVP	0	6.06×10^{-169}	0	6.18×10^{-191}	0	3.81×10^{-206}
	IGJO	2.70×10^{-126}	4.94×10^{-127}	1.30×10^{-119}	3.16×10^{-120}	7.63×10^{-120}	1.39×10^{-120}
	PSOHBFB	0	0	2.30×10^{-122}	3.15×10^{-119}	0	0
	STWNGO	0	0	0	0	0	0
F5	PSO	2.21×10^{-1}	2.58×10^1	7.22×10^{-1}	9.72×10^1	3.97×10^6	5.98×10^6
	NGO	4.22×10^{-1}	2.60×10^1	6.48×10^{-1}	9.75×10^1	3.64×10^{-1}	2.98×10^2
	GJO	6.53×10^{-1}	2.74×10^1	5.64×10^{-1}	9.79×10^1	9.54×10^{-2}	2.98×10^2
	IPSOVP	2.03×10^{-3}	8.71×10^{-4}	1.29×10^{-1}	5.57×10^{-2}	7.62×10^{-1}	3.76×10^{-1}
	IGJO	7.11×10^{-7}	2.46×10^{-4}	2.95×10^{-4}	7.87×10^{-1}	6.76×10^{-4}	5.27×10^{-2}
	PSOHBFB	1.12×10^1	9.47×10^0	1.70×10^{-2}	1.49×10^{-1}	4.75×10^{-1}	2.71×10^{-1}
	STWNGO	1.44×10^{-2}	2.34×10^{-2}	1.72×10^{-5}	3.21×10^{-2}	8.56×10^{-1}	4.28×10^{-2}
F6	PSO	1.32×10^{-3}	7.01×10^{-4}	8.18×10^{-1}	4.60×10^0	4.40×10^2	4.46×10^3
	NGO	1.20×10^{-3}	4.16×10^{-4}	8.20×10^{-1}	7.09×10^0	1.22×10^0	4.89×10^1
	GJO	4.18×10^{-1}	1.40×10^0	1.02×10^0	1.24×10^1	1.26×10^0	5.52×10^1
	IPSOVP	5.03×10^{-19}	2.30×10^{-19}	2.59×10^{-4}	1.52×10^{-4}	5.59×10^{-3}	2.48×10^{-3}
	IGJO	8.94×10^{-15}	2.96×10^{-15}	4.11×10^{-7}	9.02×10^{-4}	6.48×10^{-5}	1.52×10^{-5}
	PSOHBFB	2.08×10^{-18}	2.64×10^{-20}	7.95×10^{-2}	1.62×10^{-1}	4.26×10^{-1}	1.82×10^0
	STWNGO	5.72×10^{-20}	1.05×10^{-20}	1.31×10^{-7}	7.32×10^{-10}	6.15×10^{-9}	5.42×10^{-9}

表 4(续)
Table 4 (continued)

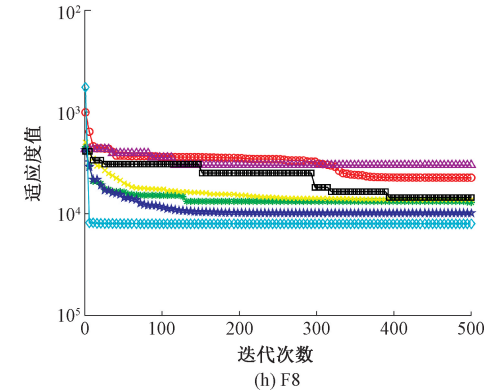
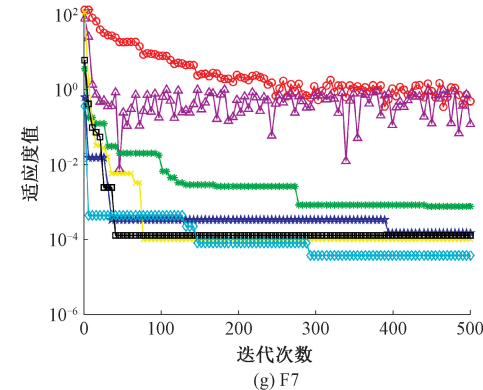
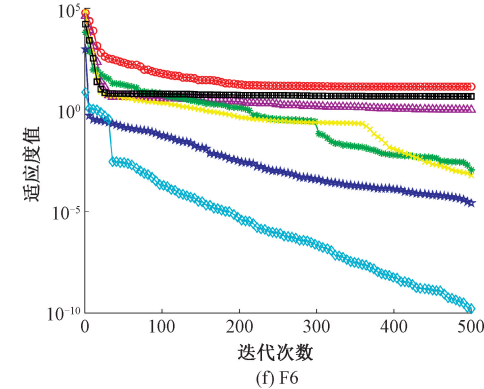
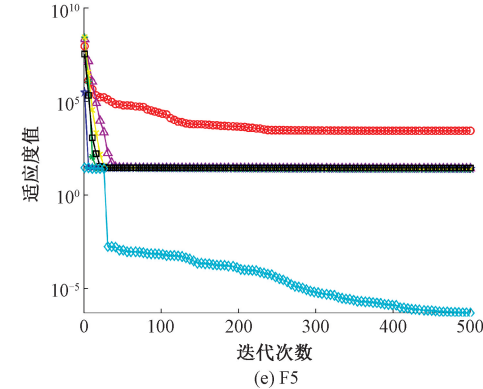
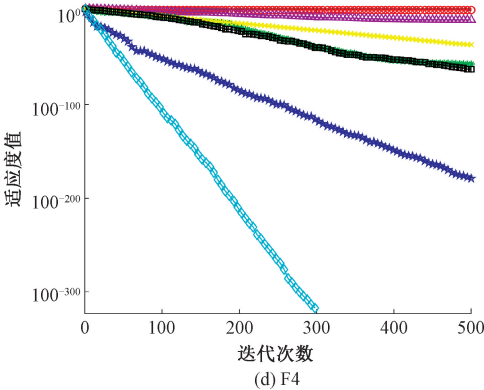
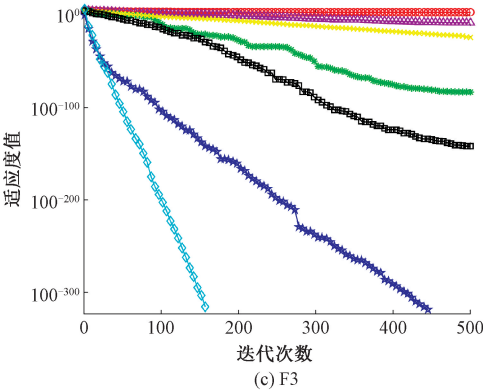
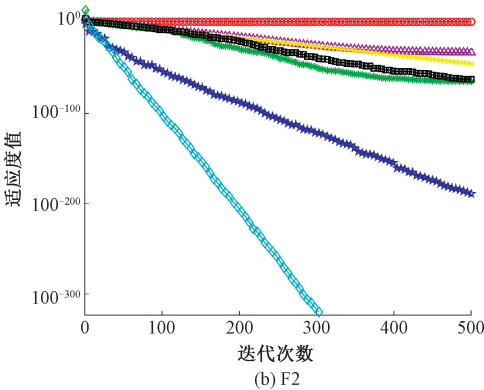
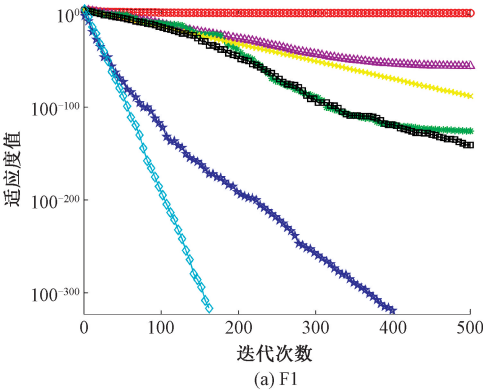
函数	算法	30D		100D		300D	
		std	avg	std	avg	std	avg
F7	PSO	1.18×10^{-1}	5.48×10^{-1}	1.11×10^{-1}	5.26×10^{-1}	8.91×10^{-4}	5.24×10^2
	NGO	2.52×10^{-4}	5.71×10^{-4}	3.10×10^{-4}	7.65×10^{-4}	4.71×10^{-4}	9.07×10^{-4}
	GJO	2.96×10^{-1}	5.74×10^{-1}	2.80×10^{-1}	4.59×10^{-1}	2.69×10^{-1}	5.11×10^{-1}
	IPSOVP	4.14×10^{-4}	3.56×10^{-4}	2.64×10^{-4}	2.33×10^{-4}	2.18×10^{-4}	2.35×10^{-4}
	IGJO	5.01×10^{-4}	6.46×10^{-4}	6.27×10^{-4}	6.77×10^{-4}	9.06×10^{-4}	9.21×10^{-4}
	PSOHBFB	1.40×10^{-4}	1.58×10^{-4}	2.03×10^{-4}	2.04×10^{-4}	1.51×10^{-4}	1.40×10^{-4}
	STWNGO	4.51×10^{-5}	4.80×10^{-5}	5.29×10^{-5}	5.81×10^{-5}	4.58×10^{-5}	5.23×10^{-5}
F8	PSO	1.65×10^{-3}	-5.35×10^3	6.51×10^{-2}	-2.41×10^4	2.20×10^{-1}	-7.50×10^4
	NGO	4.68×10^{-2}	-7.39×10^3	8.09×10^{-2}	-1.79×10^4	1.91×10^{-1}	-3.45×10^4
	GJO	1.52×10^{-3}	-4.30×10^3	4.73×10^{-2}	-9.87×10^3	1.15×10^{-1}	-2.05×10^4
	IPSOVP	4.85×10^{-2}	-1.15×10^4	9.43×10^{-2}	-2.90×10^4	3.54×10^{-2}	-5.50×10^4
	IGJO	1.60×10^{-3}	-6.55×10^4	4.96×10^{-2}	-2.78×10^4	2.32×10^{-1}	-6.13×10^4
	PSOHBFB	2.51×10^{-2}	-1.56×10^4	3.11×10^{-2}	-3.02×10^4	7.54×10^{-1}	-1.51×10^5
	STWNGO	1.19×10^{-4}	-1.24×10^4	4.23×10^{-2}	-4.18×10^4	1.97×10^{-2}	-1.25×10^5
F9	PSO	2.08×10^0	7.33×10^0	9.96×10^{-1}	1.82×10^{-1}	2.61×10^2	4.25×10^3
	NGO	0	0	0	0	0	0
	GJO	1.74×10^1	3.18×10^0	1.80×10^0	3.29×10^{-1}	2.26×10^{-1}	4.13×10^{-2}
	IPSOVP	0	0	0	0	0	0
	IGJO	0	0	0	0	0	0
	PSOHBFB	0	0	0	0	0	0
	STWNGO	0	0	0	0	0	0
F10	PSO	3.01×10^{-1}	8.37×10^{-3}	3.67×10^{-1}	4.44×10^{-6}	2.04×10^{-1}	5.63×10^6
	NGO	1.80×10^{-15}	5.89×10^{-15}	1.08×10^{-15}	7.19×10^{-15}	1.85×10^{-15}	7.31×10^{-15}
	GJO	1.79×10^{-15}	6.01×10^{-15}	1.44×10^{-14}	5.08×10^{-14}	1.03×10^{-8}	1.36×10^{-8}
	IPSOVP	0	4.44×10^{-16}	0	4.44×10^{-16}	0	4.44×10^{-16}
	IGJO	0	4.44×10^{-16}	0	4.44×10^{-16}	0	4.44×10^{-16}
	PSOHBFB	0	4.44×10^{-16}	0	4.44×10^{-16}	0	4.44×10^{-16}
	STWNGO	0	4.44×10^{-16}	0	4.44×10^{-16}	0	4.44×10^{-16}

表 5 F11、F12 的测试函数对比结果
Table 5 Comparison results of F11 and F12 test functions

函数	算法	PSO	NGO	GJO	IPSOVP	IGJO	PSOHBFB	STWNGO
F11	std	3.2261×10^{-4}	3.0644×10^{-4}	5.0566×10^{-3}	4.2668×10^{-4}	3.2909×10^{-4}	2.8675×10^{-4}	2.8358×10^{-4}
	avg	9.3438×10^{-4}	8.0758×10^{-4}	8.7747×10^{-3}	7.2700×10^{-4}	4.8351×10^{-4}	5.2049×10^{-4}	3.8089×10^{-4}
F12	std	1.3048×10^{-1}	1.8805×10^{-9}	1.1867×10^{-1}	6.0328×10^{-2}	5.8273×10^{-2}	3.3770×10^{-2}	1.5139×10^{-15}
	avg	-3.1923×10^0	-3.1570×10^0	-3.1376×10^0	-3.2665×10^0	-3.1245×10^0	-3.2158×10^0	-3.3220×10^0

为直观体现算法在迭代过程中的表现,在维度为 30 的条件下,其迭代收敛曲线如图 2 所示。由图 2 可以看出,STWNGO 具有很好的收敛迭代速度,从图 2(a)~(d)中可以看出 STWNGO 相较其他对比算法能够很快地达到理想最优值,从图 2(e)、(g)、(h)中,可以看出当其他算法因陷入局部最优,而导致寻优值停止不变,而

STWNGO 能够局部最优的状态,随着迭代次数的增加,不断趋于最优值,在图 2(i)~(l)中,可以看出 STWNGO 具有很好的收敛速度,可以用很少的迭代次数,向最优值靠近。通过收敛曲线图对比可以看出,STWNGO 既展现出了优异的收敛速度,也体现了较强的跳出局部最优解的能力。



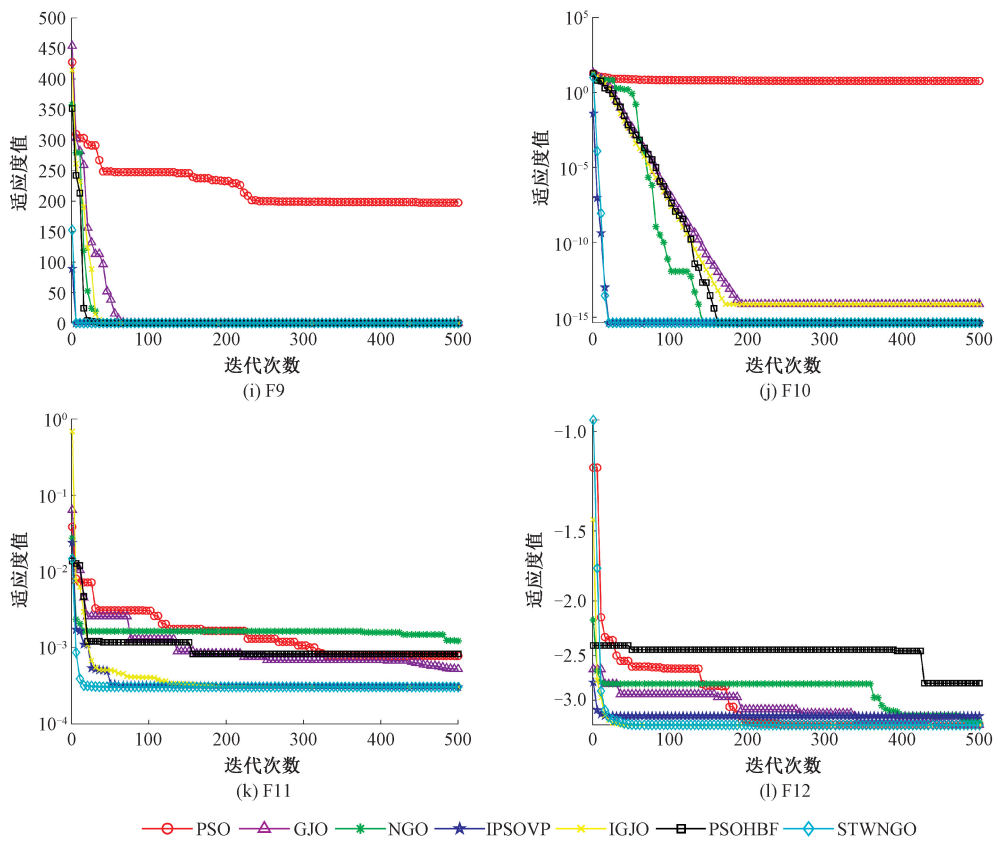


图 2 不同算法的收敛曲线对比图

Fig. 2 Comparison of convergence curves of different algorithms

3.5 Wilcoxon 秩和检验

为评估 STWNGO 的每次结果与其他算法在统计学上是否存在显著差异,使用 Wilcoxon 秩和检验^[24],并独立运行 30 次实验,来验证算法的差异,设定显著水平 $P=5\%$,当 P 值的计算结果低于 0.05 时,认为这两个算法存在明显差异,

当 P 值的计算结果超过 0.05 时,说明两种算法的运行结果并无明显差异,认为它们在寻优性能上是相似的。

从表 6 可以看出,几乎所有的 P 值都没有超过 5%,表明 STWNGO 算法与其他算法之间存在显著差异,同时具有更好的寻优性能。

表 6 Wilcoxon 秩和检验结果

Table 6 Wilcoxon rank-sum test results

函数	PSO	NGO	GJO	IPSOVP	IGJO	PSOHBF	ENGO	LNGO
F1	3.66×10^{-11}	2.16×10^{-11}	1.21×10^{-12}	3.17×10^{-11}	1.48×10^{-11}	1.21×10^{-12}	1.31×10^{-12}	2.21×10^{-12}
F2	6.54×10^{-11}	2.13×10^{-11}	3.02×10^{-11}	2.43×10^{-11}	3.64×10^{-11}	3.02×10^{-11}	3.12×10^{-11}	1.26×10^{-12}
F3	2.17×10^{-11}	4.43×10^{-11}	1.21×10^{-12}	1.43×10^{-11}	3.73×10^{-11}	1.21×10^{-12}	1.22×10^{-12}	1.23×10^{-12}
F4	3.44×10^{-11}	3.18×10^{-11}	4.16×10^{-11}	3.47×10^{-11}	1.47×10^{-11}	3.02×10^{-11}	1.65×10^{-11}	2.13×10^{-12}
F5	3.73×10^{-11}	2.08×10^{-11}	6.43×10^{-11}	3.63×10^{-11}	3.02×10^{-11}	3.26×10^{-11}	3.85×10^{-11}	3.66×10^{-11}
F6	1.36×10^{-11}	2.75×10^{-3}	4.64×10^{-11}	3.75×10^{-11}	6.72×10^{-10}	3.82×10^{-9}	3.75×10^{-11}	3.14×10^{-11}
F7	6.49×10^{-11}	3.02×10^{-11}	1.50×10^{-11}	3.02×10^{-11}	3.02×10^{-11}	3.02×10^{-11}	1.09×10^{-11}	2.57×10^{-11}
F8	3.81×10^{-7}	9.51×10^{-6}	2.13×10^{-11}	4.05×10^{-11}	1.12×10^{-2}	2.61×10^{-2}	6.53×10^{-8}	6.72×10^{-11}
F9	1.21×10^{-12}	1.72×10^{-12}	1.34×10^{-12}	1.35×10^{-12}	1.62×10^{-12}	4.21×10^{-12}	1.25×10^{-12}	5.24×10^{-12}
F10	1.14×10^{-11}	1.46×10^{-11}	1.21×10^{-12}	1.28×10^{-12}	1.85×10^{-12}	1.63×10^{-12}	1.72×10^{-12}	1.21×10^{-12}
F15	3.02×10^{-11}	1.02×10^{-5}	1.95×10^{-3}	1.11×10^{-6}	9.47×10^{-3}	3.02×10^{-11}	3.73×10^{-11}	6.36×10^{-5}
F20	1.94×10^{-11}	6.05×10^{-7}	7.21×10^{-5}	4.29×10^{-6}	1.43×10^{-6}	6.63×10^{-1}	1.72×10^{-12}	4.64×10^{-1}

3.6 CEC2017 复杂函数分析

为了进一步验证改进算法在复杂函数中的寻优能力,本文选择 CEC2017 来评估算法的搜索精确度,采用 4 种类型的测试函数,分别为一个单峰函数(UN),3 个多峰函数(MN),3 个混合函数(HF),3 个复合函数(CF),如表 7 所示。将其与 NGO、PSO、GJO、IPSOVP、IGJO、PSOHBF 进行比较,独立运行 30 次实验,选择最优一组的平均值和标准差作为评估依据,结果如表 8 所示。

由表 8 可知,在单峰、多峰测试函数中,STWNGO 在寻优精度和稳定性方面优于其他对比算法;在混合函数中,该算法仍然有优异的表现,表明该算法具有良好的稳定性;在复合函数中,虽然 CEC20 上,IGJO 的标准差好于其他算法,

但是 STWNGO 的平均值在所有对比算法中最好。综上可知,相较于其他算法,本文提出的方法在处理复杂测试函数时依旧表现出色,这也验证了其较强的鲁棒性。

表 7 CEC2017E 测试函数
Table 7 CEC2017E test functions

函数	特征	范围	最优	函数	特征	范围	最优
CEC01	UN	$[-100,100]$	100	CEC12	HF	$[-100,100]$	1200
CEC03	MN	$[-100,100]$	300	CEC18	HF	$[-100,100]$	1 800
CEC04	MN	$[-100,100]$	400	CEC20	CF	$[-100,100]$	2 000
CEC09	MN	$[-100,100]$	900	CEC25	CF	$[-100,100]$	2 500
CEC11	HF	$[-100,100]$	1100	CEC29	CF	$[-100,100]$	2 900

表 8 CEC2017 函数测试结果
Table 8 CEC2017 function test results

函数	指标	PSO	NGO	GJO	IPSOVP	IGJO	PSOHBF	STWNGO
CEC01	std	$2.093\ 7\times10^9$	$2.042\ 5\times10^7$	$3.109\ 8\times10^9$	$4.831\ 4\times10^7$	$5.731\ 0\times10^7$	$6.715\ 4\times10^9$	$2.956\ 7\times10^6$
	avg	$3.396\ 7\times10^8$	$1.559\ 1\times10^7$	$7.278\ 8\times10^9$	$9.635\ 1\times10^7$	$7.949\ 0\times10^7$	$3.938\ 3\times10^8$	$2.551\ 8\times10^6$
CEC03	std	$2.432\ 4\times10^4$	$9.090\ 1\times10^3$	$8.400\ 5\times10^3$	$8.752\ 0\times10^3$	$8.685\ 2\times10^3$	$8.869\ 6\times10^3$	$8.111\ 5\times10^3$
	avg	$9.142\ 2\times10^4$	$6.361\ 9\times10^4$	$5.077\ 3\times10^4$	$5.054\ 8\times10^4$	$4.619\ 8\times10^4$	$4.959\ 8\times10^4$	$4.185\ 3\times10^4$
CEC04	std	$1.838\ 8\times10^2$	$1.905\ 3\times10^1$	$5.596\ 2\times10^2$	$2.436\ 2\times10^1$	$2.772\ 2\times10^1$	$1.162\ 5\times10^3$	$1.084\ 2\times10^1$
	avg	$6.892\ 6\times10^2$	$8.317\ 4\times10^2$	$9.951\ 7\times10^2$	$5.868\ 5\times10^2$	$6.069\ 5\times10^2$	$5.964\ 7\times10^3$	$5.013\ 0\times10^2$
CEC09	std	$2.156\ 9\times10^3$	$8.481\ 2\times10^2$	$1.638\ 1\times10^3$	$1.029\ 3\times10^3$	$1.453\ 4\times10^3$	$1.205\ 0\times10^3$	$2.600\ 5\times10^2$
	avg	$7.351\ 7\times10^3$	$7.816\ 6\times10^3$	$9.969\ 2\times10^3$	$6.173\ 8\times10^3$	$8.950\ 6\times10^3$	$1.138\ 2\times10^4$	$5.584\ 8\times10^3$
CEC11	std	$1.319\ 4\times10^3$	$7.447\ 2\times10^1$	$4.558\ 9\times10^2$	$8.840\ 1\times10^1$	$8.297\ 3\times10^1$	$2.113\ 9\times10^3$	$6.600\ 1\times10^1$
	avg	$2.070\ 3\times10^3$	$2.260\ 0\times10^3$	$1.703\ 3\times10^3$	$1.363\ 9\times10^3$	$1.366\ 4\times10^3$	$1.364\ 0\times10^3$	$1.306\ 4\times10^3$
CEC12	std	$1.009\ 2\times10^8$	$1.062\ 7\times10^6$	$5.238\ 5\times10^8$	$2.570\ 2\times10^6$	$1.508\ 4\times10^6$	$1.819\ 2\times10^6$	$1.024\ 0\times10^6$
	avg	$7.836\ 5\times10^7$	$1.514\ 3\times10^6$	$4.545\ 9\times10^8$	$2.714\ 7\times10^6$	$1.955\ 9\times10^6$	$1.742\ 8\times10^6$	$1.235\ 0\times10^6$
CEC18	std	$7.195\ 8\times10^6$	$3.864\ 6\times10^5$	$1.994\ 1\times10^6$	$2.719\ 1\times10^6$	$9.214\ 5\times10^5$	$4.684\ 3\times10^5$	$1.038\ 1\times10^5$
	avg	$4.383\ 1\times10^6$	$2.936\ 9\times10^5$	$2.243\ 7\times10^6$	$2.158\ 5\times10^6$	$9.076\ 0\times10^5$	$6.032\ 2\times10^5$	$1.775\ 8\times10^5$
CEC20	std	$2.287\ 8\times10^2$	$6.855\ 8\times10^1$	$2.809\ 8\times10^2$	$2.181\ 8\times10^2$	$6.189\ 7\times10^1$	$1.273\ 5\times10^2$	$1.526\ 6\times10^2$
	avg	$2.743\ 5\times10^3$	$2.454\ 6\times10^3$	$2.684\ 3\times10^3$	$2.571\ 3\times10^3$	$2.759\ 9\times10^3$	$2.272\ 7\times10^3$	$2.180\ 4\times10^3$
CEC25	std	$6.814\ 6\times10^1$	$2.544\ 7\times10^1$	$1.159\ 2\times10^2$	$1.753\ 6\times10^1$	$1.631\ 0\times10^1$	$4.429\ 3\times10^2$	$1.281\ 4\times10^1$
	avg	$2.995\ 1\times10^3$	$2.948\ 6\times10^3$	$3.124\ 0\times10^3$	$2.897\ 5\times10^3$	$2.992\ 0\times10^3$	$3.056\ 0\times10^3$	$2.854\ 6\times10^3$
CEC29	std	$3.291\ 5\times10^2$	$3.658\ 7\times10^2$	$3.028\ 9\times10^2$	$2.740\ 8\times10^2$	$2.977\ 1\times10^2$	$3.894\ 0\times10^2$	$2.672\ 3\times10^2$
	avg	$4.864\ 5\times10^3$	$4.113\ 2\times10^3$	$4.169\ 7\times10^3$	$4.224\ 8\times10^3$	$4.180\ 4\times10^3$	$4.396\ 5\times10^3$	$4.094\ 2\times10^3$

4 光伏发电量预测

4.1 极限学习机原理

极限学习机(extreme learning machine,ELM)是一种利用单隐层前馈神经网络的机器学习策略^[25]。其输出层的权重是通过输入层的权重与隐藏层的偏置共同计算出来的,并采用 Moore-Penrose 广义逆矩阵理论来完成求解。设光伏电站的发电量训练样本为 $[x_i,p_i],x_i=[x_{i1},x_{i1},\cdots,x_{in}]^T\in R^N$ 表示影响光伏发电量的主要因素, $p_i\in R$ 为训练样本的输出值,因此,ELM 模型输出表达式为:

$$p_i = \sum_{i=1}^h \beta_i g(w_i \cdot x_i + b_i)$$
 (22)

式中: $g(x)$ 为隐含层神经元的激活函数, w_i 为输入与隐含层神经元之间的权值, b_i 为隐含层神经元的阈值。

ELM 网络的训练目标是将输出误差最小化,即:

$$\sum_{i=1}^N \|p_i - y_i\| = 0$$
 (23)

$$y_i = \sum_{i=1}^h \beta_i g(w_i \cdot x_i + b_i)$$
 (24)

式(24)可以写成矩阵形式 $T=H\beta$,隐含层输出矩阵 H 为:

$$\mathbf{H} = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & g(w_2 \cdot x_1 + b_2) & \cdots & g(w_h \cdot x_1 + b_h) \\ g(w_1 \cdot x_2 + b_1) & g(w_2 \cdot x_2 + b_2) & \cdots & g(w_h \cdot x_2 + b_h) \\ \vdots & \vdots & \cdots & \vdots \\ g(w_1 \cdot x_N + b_1) & g(w_2 \cdot x_N + b_2) & \cdots & g(w_h \cdot x_N + b_h) \end{bmatrix} \quad (25)$$

计算隐含层与输出层的连接权值 β :

$$\min \| H\beta - T' \| \quad (26)$$

式(26)可以表示为 $\beta = \mathbf{H}^+ T'$, 其中 \mathbf{H}^+ 是 \mathbf{H} 的 Moore-Penrose 广义逆矩阵, T' 为网络输出的转置。

4.2 基于改进北方苍鹰算法的 ELM 预测模型

由于传统 ELM 模型中输入层权重和隐层偏置具有随机性,导致训练精度降低和泛化能力不足等问题,采用 STWNGO 优化 ELM 初始权重和偏差,提高泛化能力。具体步骤如下:

1)确定影响光伏发电量变化的相关因素,即极限学习机的网络输入量。

2)首先,初始化 STWNGO 算法的相关参数,并构建极限学习机模型。同时,设定激活函数、隐层神经元数量,以及输入层权重和隐层偏置的范围。

3)将样本数据分为两部分:训练样本和测试样本,并对数据进行标准化处理。

4)依据 ELM 模型的结构,对北方苍鹰个体的位置进行编码。将均方误差函数作为适应度函数值,公式为:

$$fit = \frac{1}{N} \sum_{i=1}^N (y_i - y'_i)^2 \quad (27)$$

式中: N 为样本数, y_i 为实际发电量, y'_i 为预测发电量。

5)根据改进 NGO 的迭代寻优机制,得到优化后的权重和隐含层权重偏差,通过极限学习机训练,得到光伏发电量的预测值。

4.3 误差指标

本文采用了均方根误差 (root mean squard error, RMSE) 和决定系数 (r-square, R^2) 作为评估指标来分析预测效果。 R^2 用来衡量回归模型的拟合程度,其值介于 $[0, 1]$ 之间,数值越大,拟合效果越理想;RMSE 则通过计算均方误差的平方根来反映预测值与实际值之间的差异。

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (P_i - \hat{P}_i)^2} \quad (28)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (\hat{P}_i - P_i)^2}{\sum_{i=1}^N (\bar{P}_i - P_i)^2} \quad (29)$$

式中: p_i 代表光伏发电的实际值, \hat{P}_i 表示其预测值, \bar{P}_i 表示 P_i 的平均值,而 N 则表示样本的总数。

4.4 实验分析

以某光伏电站 2019 年 1 月 1 日到 2020 年 1 月 1 日的发电情况和天气因素作为研究样本,设置每 15 min 为 1 个样本点。这里选择秋季的发电情况进行预测,共 9 000

组样本数据,将板温、光照强度、转化效率、电流、平均功率作为输入量;将发电量作为预测值。从秋季中随机选出 30 天,其中前 29 天作为训练集,最后一天作为测试集。改进 ELM 预测建模过程参照 4.2 节,7 种改进 ELM 预测模型分别记为 PSO-ELM、NGO-ELM、GJO-ELM、IPSOVP-ELM、IGJO-ELM、PSOHBF-ELM 和 STWNGO-ELM。光伏发电量预测结果如图 3、表 9 所示。

由图 3 可知,STWNGO-ELM 模型的预测与实际数值曲线拟合度最高,其次是其他改进的 ELM 预测模型,最差的是 ELM 预测模型,证明了改进算法可以提高 ELM 预测模型的精确程度,其中 STWNGO-ELM 预测模型在众多改进模型种曲线拟合最好。

由表 9 可以看出,与他改进模型和原始 ELM 相比,STWNGO-ELM 预测模型的 RMSE 在原始 ELM 预测模型基础上降低 63.26%,同时也超过其他改进 ELM 预测模型;决定系数 R^2 是最高的,表明 STWNGO-ELM 在数值拟合度是最好的,能够最准确地预测出数据模型。

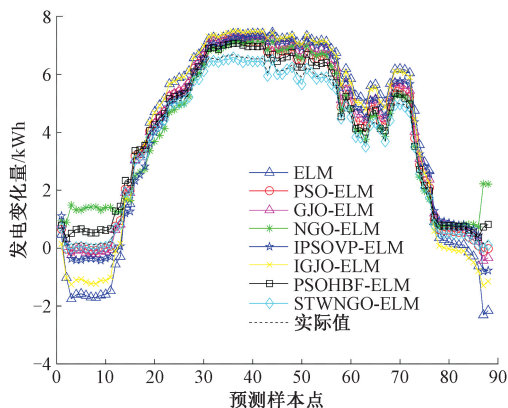


图 3 光伏发电量预测对比图

Fig. 3 Comparison of photovoltaic power generation forecasts

表 9 预测模型误差统计表

Table 9 Error statistics of the prediction model

指标	RMSE	R^2
ELM	0.946 8	0.901 2
PSO-ELM	0.512 7	0.948 2
NGO-ELM	0.593 1	0.935 6
GJO-ELM	0.540 8	0.939 2
IPSOVP-ELM	0.639 5	0.947 7
IGJO-ELM	0.750 2	0.928 1
PSOHBF-ELM	0.479 2	0.972 9
STWNGO-ELM	0.347 9	0.986 6

5 工程设计案例及分析

为了进一步验证 STWNGO 在面对复杂问题时的性能效果,本文选择两个经典的工程设计问题,对改算法进行测试,并与其他算法进行对比。

5.1 拉压弹簧设计

将拉压弹簧结构为研究对象,所求目标为在满足约束条件的前提下使弹簧的质量最小。该弹簧的结构变量为:导线直径 d ,平均线圈直径 D ,有效线圈数量 N ,分别记为 x_1, x_2, x_3 。数学模型如下:

$$\min f(x) = (x_3 + 2)x_2x_1^2 \tag{30}$$

约束条件为:

$$\begin{cases} g_1(x) = 1 - \frac{x_2^3x_3}{71\,785x_1^4} \leq 0 \\ g_2(x) = \frac{4x_2^2 - x_1x_2}{12\,566(x_2x_1^3 - x_1^4)} + \frac{1}{5\,108x_1^2} - 1 \leq 0 \\ g_3(x) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0 \\ g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0 \end{cases} \tag{31}$$

其中, $0.05 \leq x_1 \leq 2, 0.25 \leq x_2 \leq 1.3, 2 \leq x_3 \leq 15$ 。将其与 3.4 节中的 7 种算法进行对比,对比结果如表 10 所示。由表 10 可以看出在求解拉压弹簧质量问题时,STWNGO 的效果要比其他算法的效果好,其最优值所对应的解为 $x = \{0.0516, 0.3539, 11.4568\}$,对应的目标函数值 $f = 0.012\,652$ 。

表 10 拉压弹簧设计问题测试结果

Table 10 Test results for tension and compression spring design problems

算法	min	std	avg
PSO	0.0137 15	2.34×10^{-2}	0.013 915
NGO	0.013 714	3.16×10^{-3}	0.013 824
GJO	0.014 738	3.38×10^{-3}	0.014 818
IPSOVP	0.012 703	4.65×10^{-4}	0.013 003
IGJO	0.013 031	4.79×10^{-4}	0.013 233
PSOHBFB	0.012 669	3.34×10^{-4}	0.012 679
STWNGO	0.012 652	1.34×10^{-4}	0.012 658

5.2 压力容器设计

在该设计问题中,目标函数为压力容器的总成本,包括材料、成形和焊接成本。容器的两端有盖子封顶,头部一端的封盖为半球状。该问题包括 4 个决策变量:容器壁厚度 T_s 、半球封盖厚度 T_h 、半径 R 和圆柱截面的长度 L 。分别将 4 个决策变量记为 x_1, x_2, x_3, x_4 。数学模型如下:

$$\min f(x) = 0.622\,4x_1x_3x_4 + 1.778\,1x_2x_3^2 + 3.166\,1x_1^2x_4 + 19.84x_1^2x_3 \tag{32}$$

约束条件为:

$$\begin{cases} g_1(x) = -x_1 + 0.019\,3x_3 \leq 0 \\ g_2(x) = -x_2 + 0.009\,54x_3 \leq 0 \\ g_3(x) = -\pi x_3^2x_4 - \frac{4\pi x_3^3}{3} + 1\,296\,000 \\ g_4(x) = x_4 - 240 \leq 0 \end{cases} \tag{33}$$

其中, $0 \leq x_i \leq 100, i = 1, 2; 10 \leq x_i \leq 200, i = 3, 4$ 。将其与 3.4 节中的 7 种算法进行对比进行对比,对比结果如表 11 所示。由表 11 可以看出在求解压力容器设计问题时,STWNGO 的效果要比其他算法的效果好,其最优值的解为 $x = \{0.778\,2, 0.384\,6, 40.319\,6, 200\}$,应的目标函数值 $f = 5\,872.559\,2$ 。

表 11 压力容器设计问题测试结果

Table 11 Test results for pressure vessel design problems

算法	min	std	avg
PSO	7 324.370 1	5.34×10^2	8 412.173 1
NGO	7 310.370 1	3.16×10^2	8 330.362 1
GJO	7 476.986 1	4.38×10^3	9 571.643 0
IPSOVP	5 891.990 0	2.15×10^2	6 901.120 3
IGJO	5 970.745 0	4.79×10^2	7 012.182 1
PSOHBFB	5 890.908 1	3.04×10^2	6 953.143 1
STWNGO	5 872.559 2	1.21×10^1	5 989.638 8

综上所述,本文所改进后的 STWNGO 算法在工程问题中取得了比其他算法更好的效果,验证了改进算法在面对复杂工程问题时具有更好的寻优精确度。

6 结 论

本文提出一种融合减法优化器和 t 分布小波变异的改进北方苍鹰算法,该算法采用 Tent 映射结合动态反向学习策略来初始化种群,从而增强种群在初始阶段的质量和多样性,在勘探阶段融合减法平均优化器和引入最佳值引导策略更新种群位置;在开发的后期阶段,通过小波变异策略结合自适应 t 分布,对种群的位置进行扰动和优化调整,在迭代初期扩大了全局的搜索空间,此外,该算法在迭代的后期进一步提升了局部搜索效率,通过数据仿真实验可验证本文提出的算法改进效果。数据仿真实验表明,本文所提出的改进算法对比其他改进算法得到的实验结果更优异,证明其具有更强的竞争力。最后,光伏发电量预测模型的结果,再次验证了 STWNGO 的可靠性;在工程设计应用中,取得最好的寻优计算结果。综上所述,该方法展现出较强的应用潜力,未来可尝试将该算法应用到模式识别、图像处理等方面。

参考文献

[1] ZHAO W, WANG L, MIRJALILI S. Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications [J]. Computer Methods in Applied Mechanics and Engineering, 2022, 388: 114194.
[2] 黄子介,代永强. 多策略融合的斑马优化算法[J]. 国外电子测量技术, 2024, 43(9): 59-68.
HUANG Z J, DAI Y Q. Multi-strategy combined zebra

- optimization algorithm [J]. Foreign Electronic Measurement Technology, 2024, 43(9): 59-68.
- [3] 柴岩, 常晓萌, 任生. 融合多策略改进的白鲸优化算法[J]. 计算机工程与应用, 2025, 61(5): 76-93.
- CHAI Y, CH X M, REN SH. Beluga whale optimization with improved multi-strategy integration problem[J]. Computer Engineering and Applications, 2025, 61(5): 76-93.
- [4] JIAN K X, BO S. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization [J]. The Journal of Supercomputing, 2022, 79(7): 7305-7336.
- [5] 杜森, 蔡建峰. 基于动态尺度反向学习的改进金豺优化算法[J/OL]. 系统工程理论与实践, 1-19 [2025-03-20]. <http://kns.cnki.net/kcms/detail/11.2267.n.20240905.1345.012.html>.
- DU M, CAI J F. An improved golden jackal optimization algorithm using dynamic-scale opposition-based learning[J/OL]. Systems Engineering-Theory & Practice, 1-19 [2025-03-20]. <http://kns.cnki.net/kcms/detail/11.2267.n.20240905.1345.012.html>.
- [6] 付文龙, 章轩瑞, 张海荣, 等. 基于 INGO-SWGMN 混合模型的超短期风速预测研究[J]. 太阳能学报, 2024, 45(5): 133-143.
- FU W L, ZHANG X R, ZHANG H R, et al. Research on ultra-short-term wind speed prediction based on INGO-SWGMN hybrid model [J]. Acta Energiæ Solaris Sinica, 2024, 45(5): 133-143.
- [7] 林伟, 曹登学, 顾雨. 一种 IMNGO-VMD 小样本数据下的轴承故障识别方法[J]. 制造技术与机床, 2023(9): 17-24.
- LIN W, CAO D X, GU Y. A bearing fault identification method under IMNGO-VMD small sample data [J]. Manufacturing Technology & Machine Tool, 2023(9): 17-24.
- [8] 章涛, 陈勇旗, 廖紫洋, 等. 一种北方苍鹰参数优化的 VMD-MRE 轴承故障诊断方法[J/OL]. 机械科学与技术, 1-8 [2023-11-15]. <https://doi.org/10.13433/j.cnki.1003-8728.20230226>.
- ZHANG T, CHEN Y Q, LIAO Z Y, et al. A fault diagnosis method for VMD-MRE bearings based on parameter optimization of northern goshawk[J/OL]. Mechanical Science and Technology for Aerospace Engineering, 1-8 [2023-11-15]. <https://doi.org/10.13433/j.cnki.1003-8728.20230226>.
- [9] 王士彬, 李多, 赵娜, 等. 基于改进北方苍鹰算法优化混合核极限学习机的变压器故障诊断方法[J]. 湖南电力, 2023, 43(4): 125-132.
- WANG SH B, LI D, ZHAO N, et al. Transformer fault diagnosis method for hybrid extreme learning machine based on improved northern goshawk algorithm[J]. Hunan Electric Power, 2023, 43(4): 125-132.
- [10] 付雪, 朱良宽, 黄建平, 等. 基于改进北方苍鹰优化算法的多阈值图像分割[J]. 计算机工程, 2023, 49(7): 232-241.
- FU X, ZHU L K, HUANG J P, et al. Multi-threshold image segmentation based on improved northern goshawk optimization algorithm [J]. Computer Engineering, 2023, 49(7): 232-241.
- [11] 任志玲, 张景智. 多策略融合 LSSVM-NGO 的滑动电接触失效诊断[J]. 电子测量与仪器学报, 2023, 37(12): 37-47.
- REN ZH L, ZHANG J ZH. Sliding electrical contact failure diagnosis based on multi-strategy fusion LSSVM-NGO[J]. Journal of Electronic Measurement and Instrumentation, 2023, 37(12): 37-47.
- [12] 邹红波, 李奇隆. 基于改进北方苍鹰优化算法的激光焊缝图像分类研究[J]. 吉林大学学报(工学版), 1-10 [2023-11-15]. <https://doi.org/10.13229/j.cnki.jdxbgxb.20230709>.
- ZOU H B, LI Q L. Research on laser weld image classification based on improved northern goshawk optimization algorithm [J]. Journal of Jilin University (Engineering and Technology Edition), 1-10 [2023-11-15]. <https://doi.org/10.13229/j.cnki.jdxbgxb.20230709>.
- [13] LIU H, XIAO J, YAO Y, et al. A multi-strategy improved northern goshawk optimization algorithm for optimizing engineering problems [J]. Biomimetics (Basel, Switzerland), 2024, 9(9): 561.
- [14] GAMBHIR G, MANDAL K J. Shared memory implementation and performance analysis of LSB steganography based on chaotic tent map[J]. Innovations in Systems and Software Engineering, 2021, 17(4): 1-10.
- [15] 李丹丹, 朱石磊, 李仲康, 等. 多策略融合的改进金豺优化算法[J]. 传感器与微系统, 2025, 44(1): 127-130.
- LI D D, ZHU SH L, LI ZH K, et al. Multi-strategy fusion improved golden jackal optimization algorithm [J]. Transducer and Microsystem Technologies, 2025, 44(1): 127-130.
- [16] 何静, 高见, 张昌凡. 集成自适应变异混沌松鼠搜索和 LSTM 算法的 RUL 预测方法及应用[J]. 电子测量与仪器学报, 2023, 37(5): 88-97.
- HE J, GAO J, ZHANG CH F. Prediction of RUL and application of the integrated squirrel search algorithm with adaptive mutation chaos for LSTM[J]. Journal of Electronic Measurement and Instrumentation, 2023, 37

- (5):88-97.
- [17] 雷文静,何庆.多策略协同优化的北方苍鹰优化算法及应用[J].智能计算机与应用,2024,14(3):28-36.
LEI W J, HE Q. Northern goshawk optimization algorithm for multi-strategy co-optimization and its application[J]. Intelligent Computer and Applications, 2024,14(3):28-36.
- [18] 刘阳,张璐,吴德强,等.基于 INGO-SVM 的输电铁塔地脚螺栓螺母缺失无损检测方法[J].高压电器,2025,61(2):130-140.
LIU Y, ZHANG L, WU D Q, et al. Nondestructive testing method for missing anchor bolts and nuts of transmission towers based on INGO-SVM[J]. High Voltage Apparatus, 2025,61(2):130-140.
- [19] LI K, HUANG H, FU S, et al. A multi-strategy enhanced northern goshawk optimization algorithm for global optimization and engineering design problems[J]. Computer Methods in Applied Mechanics and Engineering, 2023, 415:116199.
- [20] 尹旭东,王恒,行鸿彦,等.基于改进粒子群算法的多波束卫星带宽功率联合分配[J].电子测量技术,2023,46(24):41-46.
YIN X D, WANG H, XING H Y, et al. Joint bandwidth and power allocation of multi-beam satellite based on improved particle swarm optimization algorithm[J]. Electronic Measurement Technology, 2023,46(24):41-46.
- [21] 李二超,高振磊.改进粒子速度和位置更新公式的粒子群优化算法[J].南京师大学报(自然科学版),2022,45(1):118-126.
LI ER CH, GAO ZH L. Particle swarm optimization algorithm with improved particle velocity and position update formula [J]. Journal of Nanjing Normal University (NaturalScience Edition), 2022, 45 (1): 118-126.
- [22] 朱兴淋,汪廷华,赖志勇.混合策略改进的金豺优化算法[J].计算机工程与应用,2024,60(4):99-112.
ZHU X L, WANG T H, LAI ZH Y. Hybrid-strategy improved golden jackal optimization [J]. Computer Engineering and Applications, 2024,60(4):99-112.
- [23] LAYEB A. Tangent search algorithm for solving optimization problems [J]. Neural Computing and Applications, 2022,34(11):8853-8884.
- [24] 徐明,王风富,龙文.多策略改进的徒步优化算法及其应用[J].电子测量技术,2025,48(3):60-73.
XU M, WANG F F, LONG W. Multiple strategies improved hiking optimization algorithm and its application[J]. Electronic Measurement Technology, 2025,48(3):60-73.
- [25] 孙世政,武宇峰,何江,等.基于 WOA-ELM 的空间分层结构 FBG 三维振动加速度传感器非线性解耦[J].仪器仪表学报,2024,45(7):139-147.
SUN SH ZH, WU Y F, HE J, et al. Nonlinear decoupling of spatially hierarchically structured FBG 3D vibration acceleration sensor based on WOA-ELM[J]. Chinese Journal of Scientific Instrument, 2024, 45 (7): 139-147.

作者简介

赵深, 硕士研究生, 主要研究方向为先进控制理论及应用、火电机组建模与控制。

E-mail:2635706340@qq.com

韦根原(通信作者), 硕士, 副教授, 主要研究方向为发电过程建模、仿真与控制、仪表可靠性及检测信号的获取。

E-mail:2090282033@qq.com

常耀华, 硕士研究生, 主要研究方向为先进控制理论及应用、火电机组建模与控制。

E-mail:1643893520@qq.com

陈亮, 硕士研究生, 主要研究方向为先进控制理论及应用、火电机组建模与控制。

E-mail:1619005405@qq.com