

精英反向黄金正弦鲸鱼算法及其 工程优化研究

肖子雅, 刘 升

(上海工程技术大学管理学院, 上海 201620)

摘 要: 针对鲸鱼优化算法(Whale Optimization Algorithm, WOA)存在的收敛速度慢、寻优稳定性不足等问题, 本文提出了精英反向学习的黄金正弦鲸鱼优化算法(Elite Opposition-Based Golden-Sine Whale Optimization Algorithm, EGolden-SWOA). 利用精英反向学习策略提高种群的多样性和质量可以有效提升算法的收敛速度, 同时引入黄金分割数优化 WOA 的寻优方式, 从而协调算法的全局探索与局部开发能力. 对 20 个单模态和多模态测试函数进行寻优实验, 并与 RLPSO(Reverse-learning and Local-learning Particle Swarm Optimization)、IWOA(Improved Whale Optimization Algorithm based on nonlinear convergence factor)等多个算法进行对比, 实验结果表明 EGolden-SWOA 具有更好的寻优精度和稳定性. 进一步对 EGolden-SWOA 进行求解大规模问题的实验, 实验结果表明 EGolden-SWOA 可以有效解决大规模优化问题. 最后将 EGolden-SWOA 应用于压力容器和蝶形弹簧设计优化问题, 结果表明 EGolden-SWOA 在工程优化方面的性能优于 RCSA(Rough Crow Search Algorithm)、CPSO(Co-evolutionary Particle Swarm Optimization)等改进算法, 可以有效运用于实际工程优化问题.

关键词: 鲸鱼优化算法; 精英反向学习; 黄金正弦算法; 大规模优化问题; 工程优化

中图分类号: TP301; TP391

文献标识码: A

文章编号: 0372-2112 (2019)10-2177-10

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2019.10.020

Study on Elite Opposition-Based Golden-Sine Whale Optimization Algorithm and its Application of Project Optimization

XIAO Zi-ya, LIU Sheng

(School of Management, Shanghai University of Engineering Science, Shanghai 201620, China)

Abstract: In order to improve the slow convergence rate and low stability of WOA, elite opposition-based golden-sine whale optimization algorithm is proposed. Elite opposition-based learning strategy is used to improve the diversity and quality of the population so that the convergence rate can be promoted. At the same time, golden ratio is introduced to improve the optimal method of WOA, so as to coordinate the global exploration and local exploitation. Twenty unimodal and multimodal benchmark functions are tested and compared with that of other algorithms, and the experimental results show that EGolden-SWOA has a better performance in convergence rate and stability. The high dimensional function test shows that EGolden-SWOA perform well in solving large scale optimization problem. Finally, EGolden-SWOA is applied to the optimization design of the pressure vessel and tension/compression spring, the result shows that its performance in project optimization is better than RCSA and CPSO, it can be effectively applied to project optimization.

Key words: whale optimization algorithm (WOA); elite opposition-based learning; golden sine algorithm (Golden-SA); large scale optimization problem; project optimization

1 引言

受自然界中生物、物理现象影响而设计出的各种

智能算法因其概念简单、易于实施等优点, 在工程领域的应用愈发广泛. 鲸鱼优化算法(Whale Optimization Algorithm, WOA)^[1]是 2016 年 Mirjalili 等人的提出的一种

新颖算法,因其寻优性能较好、控制参数较少等优势被广泛关注.鲸鱼优化算法是根据座头鲸的狩猎行为设计的,目前该算法已成功用于电容器优化选址^[2]、CO₂排放预测^[3]、临床医学^[4]、图像分割^[5]等实际问题.在工程、医学、经济领域的良好表现表明该算法具有广阔的应用前景.

为使鲸鱼优化算法具有更加优秀的性能,国内外学者对鲸鱼优化算法进行了相关的改进研究,Oliva D等人利用混沌映射产生的混沌序列进行鲸鱼种群位置的初始化^[6],维持初始种群的多样性以便更好地进行全局搜索;Y Ling等人提出一种基于莱维飞行的鲸鱼优化算法^[7]以提升算法的效果.尽管目前已有的改进算法在函数优化领域取得了一些成果,但大多数改进算法只适用于100维以内的中小规模优化问题,在大规模优化问题上表现欠佳.而在实际的工程应用中大规模优化问题普遍存在,因此本文提出了精英反向学习的黄金正弦鲸鱼算法,通过求解20个函数优化问题验证其有效性,并将其应用于求解大规模优化问题,实验结果表明该算法在求解函数优化问题上表现优秀并能有效处理大规模优化问题.最后将其用于解决实际工程优化问题,结果表明精英反向学习的黄金正弦鲸鱼算法能有效处理实际工程问题.

2 基本鲸鱼优化算法

鲸鱼是世界上最具情感的高度智能动物之一,研究发现鲸鱼大脑的某些区域中有类似人类的梭形细胞,这些细胞使得鲸鱼可以像人一样判断、交流甚至学习.鲸鱼优化算法模仿的座头鲸是最大的须鲸之一,座头鲸的主要狩猎方式为泡网觅食法,它们喜欢深入水下收缩包围猎物而后螺旋上升在靠近海面的地方捕食猎物.从座头鲸这种独特的捕食方式中获取灵感,Mirjalili等人设计了鲸鱼优化算法.

鲸鱼优化算法的基本过程描述如下:假定 s 只鲸鱼分布在 d 维搜索空间中,第 t 代第 i 只鲸鱼在 d 维搜索空间中的位置为 $\mathbf{X}_i^t = (X_{i,1}, X_{i,2}, \dots, X_{i,d})$, ($i = 1, 2, \dots, s$; $t = 1, 2, \dots, t_{\max}$), t_{\max} 为最大迭代次数,全局最优位置用 $\mathbf{X}_p^t = (X_{p,1}, X_{p,2}, \dots, X_{p,d})$ 表示为猎物位置.

由于初始化鲸鱼种群无先验经验,所以假设当前种群中的最优位置为猎物位置,其他鲸鱼个体以不同的机制向猎物位置趋近以更新自己的位置,鲸鱼优化算法中鲸鱼的位置更新方式有以下三种机制.

(1) 包围式狩猎机制

座头鲸能够识别猎物的位置并将猎物包围,该过程的位置更新公式如下:

$$\mathbf{X}_i^{t+1} = \mathbf{X}_p^t - A * \mathbf{D} \quad (1)$$

其中 $\mathbf{D} = |C * \mathbf{X}_p^t - \mathbf{X}_i^t|$ 表示包围步长,参数 A 和 C 定义

为 $A = 2a * r_1 - a$, $C = 2 * r_2$, r_1 和 r_2 为 $[0, 1]$ 之间的随机数, $a = 2 - \frac{2 * t}{t_{\max}}$ 为收敛因子,在迭代过程中随着迭代次数 t 的增加从2线性减小到0, C 是 $[0, 2]$ 之间的随机数, C 控制着猎物位置 \mathbf{X}_p^t 距离个体 \mathbf{X}_i^t 远近程度的影响, $C > 1$ 时,影响程度增强,反之减弱.

(2) 螺旋式狩猎机制

座头鲸向猎物位置移动并通过螺旋式运动进行狩猎,该过程的位置更新公式如下:

$$\mathbf{X}_i^{t+1} = \mathbf{D}' * e^{bl} * \cos(2\pi l) + \mathbf{X}_p^t \quad (2)$$

其中 $\mathbf{D}' = |\mathbf{X}_p^t - \mathbf{X}_i^t|$ 表示个体 \mathbf{X}_i^t 与猎物位置 \mathbf{X}_p^t 的距离,螺旋形状由常数 b 限定,随机数 $l \in [-1, 1]$, l 取值为-1时鲸鱼个体距离猎物位置最近, l 取值为1时鲸鱼个体距离猎物位置最远.

值得注意的是,座头鲸在不断缩小的范围内围绕猎物游动,同时进行螺旋式狩猎,这种同时进行的行为即为鲸鱼的泡网狩猎行为,假设泡网狩猎行为中鲸鱼个体选择包围狩猎机制和螺旋式狩猎机制的概率 p_r 均为50%,则该行为可表示为:

$$\mathbf{X}_i^{t+1} = \begin{cases} \mathbf{X}_p^t - A * \mathbf{D}, & p_r < 0.5 \\ \mathbf{D}' * e^{bl} * \cos(2\pi l) + \mathbf{X}_p^t, & p_r \geq 0.5 \end{cases} \quad (3)$$

(3) 随机狩猎机制

除泡网狩猎行为外,座头鲸还可以根据彼此位置进行随机狩猎,该过程的位置更新公式如下:

$$\mathbf{X}_i^{t+1} = \mathbf{X}_R^t - A * \mathbf{D}'' \quad (4)$$

其中 \mathbf{X}_R^t 为当前群体中随机选取的鲸鱼个体位置, $\mathbf{D}'' = |C * \mathbf{X}_R^t - \mathbf{X}_i^t|$ 表示个体 \mathbf{X}_i^t 与随机位置 \mathbf{X}_R^t 的距离,参数 A 和 C 定义公式及取值范围与包围式狩猎机制中的定义一致, C 控制着随机位置 \mathbf{X}_R^t 距离个体 \mathbf{X}_i^t 远近程度的影响.

当参数 $|A| < 1$ 时,鲸鱼个体远离随机个体 \mathbf{X}_R^t ,包围猎物,趋向于最优个体 \mathbf{X}_p^t ,当参数 $|A| \geq 1$ 时,鲸鱼个体被迫偏离猎物位置,以此搜寻更优的猎物,因此参数 A 的取值决定了鲸鱼个体是采用式(1)还是式(4)进行位置更新,由于参数 A 在很大程度上依赖于收敛因子 a ,因此收敛因子 a 是平衡算法“探索”和“开发”的重要因素.

综上所述,基本鲸鱼优化算法的流程图如图1所示.

3 精英反向学习的黄金正弦鲸鱼优化算法

3.1 精英反向学习策略

反向学习策略 (Opposition-Based Learning, OBL)^[8]是Tizhoosh于2005提出的,Tizhoosh的研究表明反向解更接近全局最优的概率比当前解高出50%.该策略可以有效增加种群的多样性和质量,防止算法

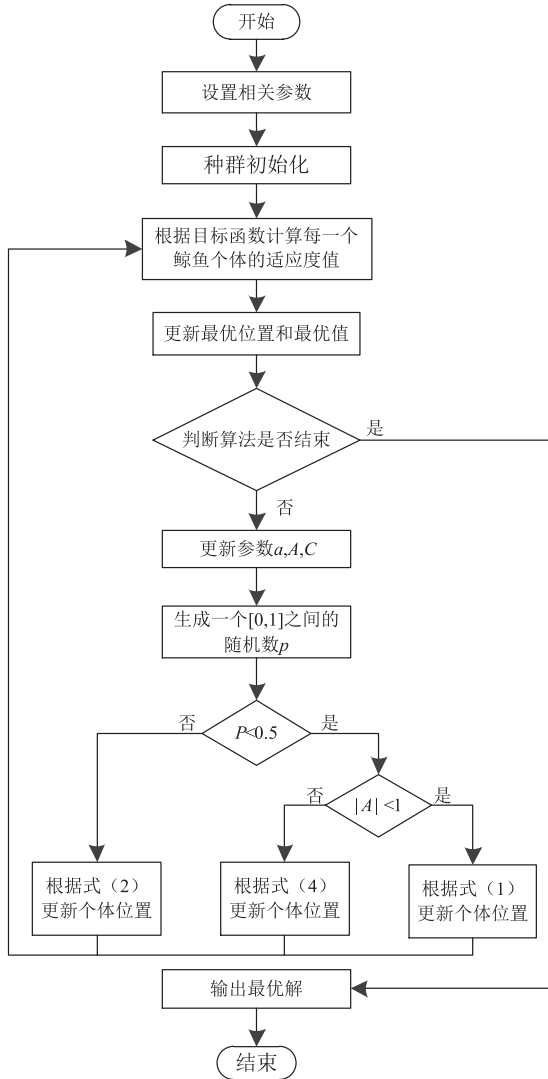


图1 WOA的算法流程图

进入“早熟”状态,目前该策略已被应用于多种算法的改进并取得较好效果,如夏学文等人将反向学习策略引入粒子群算法,有效解决了粒子群算法种群多样性和收敛性之间的矛盾^[9];魏文红等人将反向学习引入差分进化算法,有效提高了算法的多样性和全局收敛速度^[10]。反向学习策略计算并评估当前解的反向解,从当前解及其反向解中选取最优解个体作为新一代个体。

定义1(反向解) 假设当前种群的一个可行解为 $X = (x_1, x_2, \dots, x_d)$ (d 维搜索空间; $x_j \in [a_j, b_j]$), 则其反向解为 $\bar{X} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_d)$, 其中 $\bar{x}_j = \omega(a_j + b_j) - x_j$, ω 为在 $[0, 1]$ 上均匀分布的一般化系数。

精英反向学习 (Elite Opposition-Based Learning, EOBL) 是针对反向学习策略生成的反向解不一定比当前搜索空间更容易搜索到全局最优解这一问题提出的,已成功应用于多种算法的改进研究,如谢承旺、

王志杰等人将动态反向学习策略引入到进化算法提出了一种新型多目标进化算法^[11],谢承旺、许雷等人还将精英反向学习机制以及烟花爆炸算法引入多目标优化领域并取得了较好效果^[12],周新宇等人将精英粒子引入粒子群算法增强了算法性能^[13]。该策略利用精英个体比一般个体包含更多有效信息的这一特点,通过当前种群中的精英个体构造出反向种群以此来增加种群的多样性,并从当前种群和反向种群构成的新种群中选取最优个体作为新一代个体,进入下一次的迭代。

定义2(精英反向解) 假设当前种群中的一般个体对应的自身极值点为精英个体 $X_{i,j}^e = (X_{i,1}^e, X_{i,2}^e, \dots, X_{i,d}^e)$ ($i = 1, 2, 3, \dots, s; j = 1, 2, \dots, d$), 其反向解 $\bar{X}_{i,j}^e = (\bar{X}_{i,1}^e, \bar{X}_{i,2}^e, \dots, \bar{X}_{i,d}^e)$, 可以定义为:

$$\bar{X}_{i,j}^e = K * (\alpha_j + \beta_j) - X_{i,j}^e \quad (5)$$

其中, K 为在 $(0, 1)$ 上的动态系数, $X_{i,j}^e \in [\alpha_j, \beta_j]$, $\alpha_j = \min(X_{i,j}^e)$, $\beta_j = \max(X_{i,j}^e)$, α_j 和 β_j 为动态边界,动态边界克服了固定边界难以保存搜索经验的缺点,使精英反向解可以定位在狭窄的搜索空间中,有利于算法收敛。若动态边界操作使 $\bar{X}_{i,j}^e$ 越过边界成为非可行解,可以利用随机生成的方法重置,重置方式如下:

$$\bar{X}_{i,j}^e = \text{rand}(\alpha_j, \beta_j) \quad (6)$$

3.2 黄金正弦算法

黄金正弦算法 (Golden Sine Algorithm, Golden-SA) 是 Erkan Tanyildizi 于 2017 年提出的新型智能算法^[14],该算法的设计灵感来源于数学中的正弦函数,其优点是收敛迅速、鲁棒性好、易于实现。

Golden-SA 不是模拟自然现象设计的,而是利用数学中的正弦函数进行计算迭代寻优,并在位置更新过程中引入黄金分割数使“搜索”和“开发”达到良好的平衡,因此该算法不需任何假设条件。

Golden-SA 算法的核心过程是其解的更新过程,首先随机产生 s 个个体的位置,假设优化问题的每个解对应搜索空间中对应个体的位置 $X_i^t = (X_{i,1}^t, X_{i,2}^t, \dots, X_{i,d}^t)$, 其中 X_i^t 表示 d 维个体空间中第 t 次迭代中第 i ($i = 1, 2, \dots, s$) 个个体的空间位置, $P_i^t = (P_{i,1}^t, P_{i,2}^t, \dots, P_{i,d}^t)$ 表示第 t 代个体 i 的最优位置,在 $t+1$ 次迭代中,第 i 个个体的位置更新公式如下:

$$X_i^{t+1} = X_i^t * |\sin(R_1)| + R_2 * \sin(R_1) * |x_1 * P_i^t - x_2 * X_i^t| \quad (7)$$

其中, R_1 为 $[0, 2\pi]$ 的随机数,决定下一次迭代中个体的移动距离, R_2 为 $[0, \pi]$ 的随机数,决定下一次迭代第 i 个个体的位置更新方向; x_1 和 x_2 是通过引入黄金分割数得到的系数,这些系数缩小了搜索空间引领个体趋近最优值,黄金分割数是一个定义为 $(\sqrt{5}-1)/2$ 的无理

数,记为 $\tau, x_1 = -\pi + (1 - \tau) * 2\pi, x_2 = -\pi + \tau * 2\pi$.

3.3 精英反向学习的黄金正弦鲸鱼优化算法

针对基本鲸鱼优化算法存在的不足,本文提出了精英反向学习的黄金正弦鲸鱼优化算法(Elite Opposition-Based Golden-Sine Whale Optimization Algorithm, EGolden-SWOA).使用精英反向学习策略提高种群的多样性,采用群体选择机制,将当前鲸鱼群体与其反向群体按适应值排序,从中选择出最优的 s 个个体作为下一代鲸鱼个体以此提高种群的质量.首先,对于初始种群来说,精英反向学习策略的引入可以提升其多样性,为较好的全局搜索奠定基础;其次,对于每一代种群,精英反向学习策略可以通过产生远离局部极值点的反向解引导算法跳出局部最优,增强算法的全局寻优能力.同时,精英反向学习所采用的动态边界的跟踪搜索模式可获得一个逐步缩小的搜索空间,有利于提高全局收敛速度,从而协调算法的全局探索与局部开发能力.

另外,本文在保留基本鲸鱼优化算法的包围式狩猎机制和随机狩猎机制的基础上利用黄金正弦算法对螺旋式狩猎机制进行了改进.在螺旋式狩猎机制中,座头鲸在寻找到猎物后,向猎物位置移动并通过对数螺旋式运动捕获猎物,猎物位置即最优个体的位置 X_p^t 是此螺旋运动过程中的导向坐标,这虽然有助于算法后期收敛速度的提升,但是该寻优方式易使种群个体在解空间内迅速集聚,从而导致群体多样性锐降,增大陷入局部最优的可能性.针对这一问题,本文引入了黄金正弦算法对螺旋式狩猎机制进行了改进. EGolden-SWOA 算法中鲸鱼向猎物位置移动并通过黄金正弦螺旋式运动捕获猎物,每一次迭代中鲸鱼个体都会与最优个体进行信息交流,每个个体都可以充分吸收自身与最优个体的位置差信息,同时利用引入黄金分割数而得到的系数逐步缩小搜索空间,且通过参数 R_1 和 R_2 控制位置更新距离和方向,优化了基本 WOA 算法的寻优方式,引领个体稳步趋近最优值,协调算法的全局探索与局部开发能力,从而提高算法的寻优精度和速度,获得理想的寻优结果.引入黄金正弦算法后该过程的位置更新公式如下:

$$X_i^{t+1} = X_i^t * |\sin(R_1)| + R_2 * \sin(R_1) * |x_1 * X_p^t - x_2 * X_i^t| \quad (8)$$

EGolden-SWOA 具体执行步骤如下:

步骤 1 设置相关参数;

步骤 2 种群初始化,主要包括初始种群个体数 s 、候选解维度 d 、最大迭代次数 t_{\max} ;

步骤 3 根据目标函数计算每一个鲸鱼个体的适

应度值;

步骤 4 反向种群 $OP = \{\}$;

步骤 5 根据 $\alpha_j = \min(X_{i,j}), \beta_j = \max(X_{i,j})$ 计算个体的当前搜索边界;

步骤 6 对种群中的每个个体根据式(5)生成精英反向解并添加到反向种群 OP 中;

步骤 7 从当前种群和反向种群中选取适应度值较好的 s 个优良个体作为下一代种群,并将适应度值最小的个体位置记为猎物位置 X_p^t ;

步骤 8 更新参数 a, A, C ;

步骤 9 若参数 $|A| \geq 1$,根据式(4)进行位置行更新;

步骤 10 若参数 $|A| < 1$,根据式(9)对鲸鱼位置进行更新:

$$X_i^{t+1} = \begin{cases} X_p^t - A \cdot D, & p_r < 0.5 \\ X_i^t * |\sin(R_1)| + R_2 * \sin(R_1) * |x_1 * X_p^t - x_2 * X_i^t|, & p_r \geq 0.5 \end{cases} \quad (9);$$

步骤 11 利用约束条件检查新位置的可行性,若新位置可行则鲸鱼个体更新其位置,否则鲸鱼个体保持其原有位置;

步骤 12 重复步骤 3 ~ 步骤 11,直到达到最大迭代次数 t_{\max} 时终止操作.

4 实验仿真与结果分析

4.1 实验参数设置

本文选取了基本鲸鱼优化算法(WOA)、精英反向学习的黄金正弦鲸鱼算法(EGolden-SWOA)、粒子群算法(PSO)以及黄金正弦(Golden-SA)进行对比,所有算法的种群规模设为 30,迭代次数设为 500,所共有参数保持一致.各算法的参数设置如表 1 所示.

表 1 各算法实验参数设置

算法	参数
WOA	$b = 1$
EGolden-SWOA	$b = 1, x_1 = -\pi + (1 - \tau) * 2\pi, x_2 = -\pi + \tau * 2\pi$
PSO	$w = 1, w_{\text{damp}} = 0.99, c_1 = 1.5, c_2 = 2$
Golden-SA	$a = -\pi, b = \pi$

4.2 测试函数

为了进一步验证 EGolden-SWOA 的优化效果和稳定性优于其他算法,本文对多个不同特点的测试函数进行函数优化的对比测试,测试函数及其具体信息如表 2 所示.

表 2 测试函数及其具体信息

函数	函数名称	维数	范围	最优值
$f_1 = \sum_{i=1}^n x_i^2$	Sphere Model	30	$[-100, 100]$	0
$f_2 = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	Schwefel's problem 2.22	30	$[-10, 10]$	0
$f_3 = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	Schwefel's problem 1.2	30	$[-100, 100]$	0
$f_4 = \max_i \{ x_i , 1 \leq i \leq n \}$	Schwefel's problem 2.21	30	$[-100, 100]$	0
$f_5 = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	Generalized Rosenbrock's Function	30	$[-30, 30]$	0
$f_6 = \sum_{i=1}^n ([x_i + 0.5])^2$	Step function	30	$[-100, 100]$	0
$f_7 = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1]$	Quartic Function	30	$[-1.28, 1.28]$	0
$f_8 = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	Generalized Schwefel's problem 2.26	30	$[-500, 500]$	$-418.9829 * n$
$f_9 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	Generalized Rastrigin's Function	30	$[-5.12, 5.12]$	0
$f_{10} = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	Ackley's Function	30	$[-32, 32]$	0
$f_{11} = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	Generalized Griewank Function	30	$[-600, 600]$	0
$f_{12} = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\}$ $y_i = 1 + \frac{x_i + 1}{4}$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$	Generalized Penalized Function	30	$[-50, 50]$	0
$f_{13} = 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] \right. \\ \left. + (x_i - 1)^2 [1 + \sin^2(2\pi x_i)] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	Generalized Penalized Function	30	$[-50, 50]$	0
$f_{14} = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right)^{-1}$	Shekel's Foxholes Function	2	$[-65, 65]$	1
$f_{15} = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$	Kowalik's Function	4	$[-5, 5]$	0.00030
$f_{16} = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	Six-Hump Camel-Back Function	2	$[-5, 5]$	-1.0316
$f_{17} = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos x_1 + 10$	Branin Function	2	$[-5, 5]$	0.398
$f_{18} = [1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \\ * [30 + (2x_1 - 3x_2)^2 * (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	Goldstein-Price Function	2	$[-2, 2]$	3
$f_{19} = - \sum_{i=1}^4 c_i \exp\left(- \sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2\right)$	Hartman's Function	3	$[1, 3]$	-3.86
$f_{20} = - \sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	Shekel's Family	4	$[0, 10]$	-10.1532

4.3 与其他智能算法的对比分析

为了验证本文 EGolden-SWOA 算法改进的合理性和有效性,本文对算法的性能在多个测试函数上进行了试验,为了避免偶然性带来的结果偏差,算法在每个函数上

独立运行 30 次. 表 3 列出了精英反向学习的黄金正弦算法 (EGolden-SWOA)、鲸鱼优化算法 (WOA)、粒子群算法 (PSO) 以及黄金正弦算法 (Golden-SA) 在多个标准测试函数上经过 30 次独立运行后所得的实验结果.

表 3 测试函数实验结果

统计量	算法	f_1	f_2	f_3	f_4	f_5	f_6	f_7
最优值	WOA	2.76E-82	1.43E-57	9.61E+03	3.83E+00	2.77E+01	2.02E-01	7.53E-04
	EGolden-SWOA	0	1.07E-214	0	1.33E-210	8.25E-12	9.13E-14	4.96E-06
	Golden-SA	0	6.47E-159	2.33E-302	1.14E-164	2.12E-05	9.90E-07	3.76E-05
	PSO	6.05E-13	2.47E-03	1.62E+01	1.01E+00	1.93E+01	3.04E-11	1.73E-02
平均值	WOA	7.61E-75	1.85E-52	4.35E+04	4.56E+01	2.82E+01	4.62E-01	4.92E-03
	EGolden-SWOA	0	6.69E-202	0	3.58E-191	3.75E-09	6.86E-10	3.25E-05
	Golden-SA	1.60E-282	1.59E-135	7.07E-245	4.12E-138	9.68E-03	2.52E-04	1.81E-04
	PSO	2.61E-06	7.94E-02	5.51E+01	2.22E+00	7.65E+01	1.52E-06	3.29E-02
标准差	WOA	2.19E-74	4.14E-52	1.83E+04	2.69E+01	3.39E-01	2.34E-01	4.25E-03
	EGolden-SWOA	0	0	0	0	7.45E-09	1.29E-09	2.55E-05
	Golden-SA	0	4.78E-135	0	1.24E-137	1.08E-02	3.25E-04	1.06E-04
	PSO	6.06E-06	7.96E-02	2.73E+01	1.31E+00	1.15E+02	2.01E-06	9.15E-03

统计量	算法	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}
最优值	WOA	-1.26E+04	0	8.88E-16	0	9.36E-03	1.10E-02	0.998
	EGolden-SWOA	-5.58E+102	0	8.88E-16	0	8.53E-14	2.63E-11	0.998
	Golden-SA	-1.26E+04	0	8.88E-16	0	1.55E-07	3.42E-08	0.998
	PSO	-7.67E+03	3.88E+01	9.31E-01	1.49E-09	4.90E-10	1.03E-11	1.992
平均值	WOA	-1.04E+04	0	4.44E-15	0	2.17E-02	3.01E-01	3.6457
	EGolden-SWOA	-5.58E+101	0	8.88E-16	0	1.56E-10	8.68E-10	0.998
	Golden-SA	-1.26E+04	0	8.88E-16	0	1.45E-05	2.14E-05	0.998
	PSO	-6.28E+03	5.52E+01	1.96E+00	3.02E-02	1.24E-01	1.16E-01	5.5988
标准差	WOA	1.77E+03	0	2.25E-15	0	1.09E-02	1.99E-01	3.65E+00
	EGolden-SWOA	1.67E+102	0	0	0	2.61E-10	6.65E-10	0
	Golden-SA	1.37E-01	0	1.97E-31	0	2.25E-05	3.64E-05	0
	PSO	7.81E+02	1.16E+01	7.47E-01	3.20E-02	1.84E-01	2.20E-01	4.26E+00

统计量	算法	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}
最优值	WOA	0.0003	-1.0316	0.39789	3	-0.3005	-10.1529
	EGolden-SWOA	0.0003	-1.0316	0.39789	3	-1.8997	-10.1532
	Golden-SA	0.0003	-1.0316	0.39789	3.008	-0.3005	-10.1532
	PSO	0.0003	-1.0316	0.39789	3	-0.3005	-10.1532
平均值	WOA	0.0007	-1.0316	0.3979	3.0001	-0.3005	-9.57966
	EGolden-SWOA	0.0003	-1.0316	0.39789	3	-1.8997	-10.1532
	Golden-SA	0.0004	-1.02733	0.39789	14.1388	-0.3005	-10.15169
	PSO	0.0004	-1.0316	0.39789	3	-0.3005	-6.38727
标准差	WOA	5.27E-04	2.22E-16	1.76E-05	7.81E-05	5.55E-17	1.52E+00
	EGolden-SWOA	1.68E-05	2.22E-16	0	0	4.90E-05	0
	Golden-SA	1.13E-04	3.99E-03	0	1.35E+01	5.55E-17	1.31E-03
	PSO	3.10E-04	2.22E-16	0	0	5.55E-17	3.21E+00

$f_1 \sim f_7$ 为单模态函数,一般用于检测算法的开发能力. 根据实验结果可知,对于 f_1 ,EGolden-SWOA 寻优效果最佳可以得到最优值 0,Golden-SA 也有着较好寻优效果,平均寻优精度达到 10^{-128} ,PSO 寻优效果最差;对于 f_2 ,EGolden-SWOA 寻优性能明显优于其他算法,Golden-SA 也表现良好,效果仅次于 EGolden-SWOA,WOA 也能得到较理想的寻优效果,PSO 表现最差;对于

$f_3 \sim f_5$,EGolden-SWOA 的寻优性能最好,Golden-SA 次之,WOA 和 PSO 寻优效果不理想;对于 f_6 ,EGolden-SWOA 的寻优效果最好,PSO 次之,WOA 最差;对于 f_7 ,EGolden-SWOA 寻优效果略优于其他算法,4 种算法对该函数的寻优效果均不理想. 从整体上看,对于单模态函数,EGolden-SWOA 与其他算法相比最优值及平均值均最好,且根据标准差可知,EGolden-SWOA 的寻优稳

定性较好.

$f_8 \sim f_{20}$ 为多模态函数,一般用来评价算法的探索能力.从整体上看,除 f_{19} 外,EGolden-SWOA 均表现出的最佳的寻优效果,其寻优平均值优于其他三种算法,多个函数可以直接搜索到函数最优值.此外,EGolden-SWOA 的标准差均优于其他三种算法,说明 EGolden-SWOA 鲁

棒性较好.以上分析表明 EGolden-SWOA 的整体寻优能力优于 WOA、Golden-SA 和 PSO.

为了进一步验证 EGolden-SWOA 的收敛性,实验选取了测试函数的迭代收敛曲线,限于篇幅限制,这里仅列出 f_1, f_5, f_{12}, f_{15} ,如图 2 所示.从图 2 可以看出,EGolden-SWOA 是优于 WOA 以及 Golden-SA 和 PSO 的.

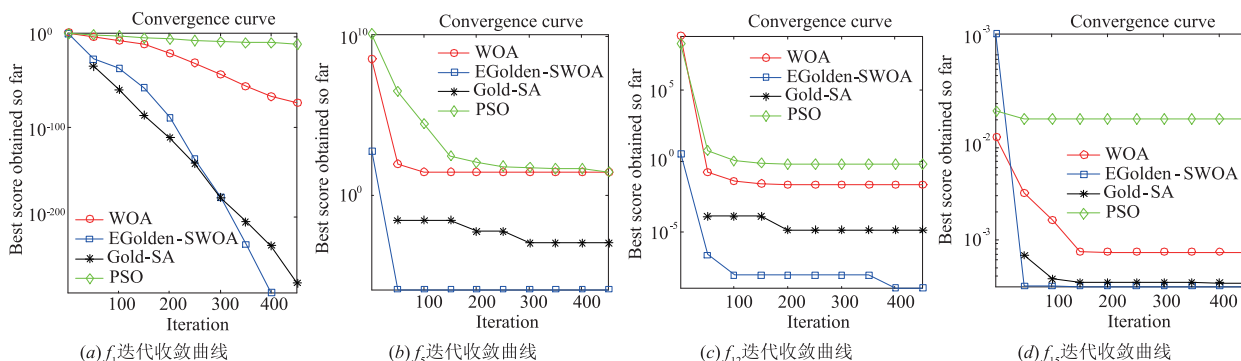


图2 部分测试函数的迭代收敛曲线

最后将 EGolden-SWOA 与具备反向学习和局部学习能力的粒子群算法(RLPSO)进行对比,RLPSO 的相关数据直接来源于文献[9],为保证算法对比的科学性与有效性,本文将种群规模设为 20,迭代次数设为 500,维数设为 30,最大评价次数设为 10^5 次,与文献[9]保持一致,对比结果如表 4 所示.

由表 4 可知 EGolden-SWOA 的收敛精度优于 RLPSO,本文对 WOA 的改进是科学有效的.

表 4 两种算法的对比结果

函数	EGolden-SWOA	RLPSO ^[9]
f_1	0	$1.23\text{E}-13$
f_2	$3.2635\text{e}-199$	$9.02\text{E}-08$
f_5	$1.9783\text{e}-07$	$2.03\text{E}+02$
f_9	0	$1.56\text{E}+00$
f_{11}	0	$1.63\text{E}-02$

基于测试函数的适应度值的比较结果(表 3、表 4)和部分迭代收敛曲线(图 2)的分析,表明了本文提出的精英反向学习的黄金正弦鲸鱼算法 EGolden-SWOA 具有良好的寻优性能,在函数优化方面是有效可靠的.

4.4 EGolden-SWOA 的时间复杂度分析

对 EGolden-SWOA 的算法时间复杂度进行分析,设算法的种群规模为 s ,问题维度为 d ,最大迭代次数为 t_{\max} ,则每个操作的时间复杂度如下:初始化 $o(s * d)$,精英反向学习 $o(s * d)$,适应度计算 $o(s * d)$,选择策略进行鲸鱼个体位置更新 $o(s * d * d)$,整体复杂度为 $o(t_{\max} * s * d * d)$,这与基本 WOA 的时间复杂度相同,没有增加计算负担.为直观地对比两种算法的时间复杂度,本节从 4.2 节选择 5 个测试函数分别用 EGolden-SWOA、WOA 在同一平台上单独进行 30 次实验,计算

30 次实验的平均耗时,实验结果如表 5 所示.

表 5 EGolden-SWOA 与 WOA 的平均消耗时间对比

函数	平均耗时		EGolden-SWOA 与 WOA 平均耗时的百分比
	WOA	EGolden-SWOA	
f_1	2.777	1.739	62.62%
f_3	3.603	3.586	99.51%
f_{12}	4.795	2.414	50.34%
f_{14}	3.1912	2.723	85.33%
f_{15}	1.4452	0.7278	50.36%

从表 5 可以看出,本文对基本 WOA 的改进不仅没有增加算法的时间复杂度,而且通过改进提高了算法的收敛速度,增强了算法的寻优性能.

4.5 与其他改进策略的鲸鱼算法的对比分析

将本文所提出的 EGolden-SWOA 算法对 $f_1 \sim f_{10}$ 函数的求解结果与文献[15]中提出的基于非线性收敛因子的改进鲸鱼优化算法(IWOA)的求解结果进行对比,分析两种算法的寻优性能,进一步验证 EGolden-SWOA 算法的有效性和可行性.

IWOA 算法的实验参数按照文献[15]给出的数据进行设置,其中 $a_{\text{initial}} = 2$, $a_{\text{final}} = 1$, $\mu = 25$, EGolden-SWOA 算法的实验参数设置为 $x_1 = -\pi + (1 - \tau) * 2\pi$, $x_2 = -\pi + \tau * 2\pi$.为保证对比实验的科学性与有效性,本文使两种算法共有的实验参数保持一致,即种群规模统一设为 30,迭代次数设为 500,共有参数 $b = 1$.两种改进算法的对比实验结果如表 6 所示.

从表 6 中实验结果可知,对于这 10 个测试函数来说,除 f_6 和 f_8 外,EGolden-SWOA 的寻优平均值及标准差全部优于 IWOA,本文提出的 EGolden-SWOA 比 IWOA 在寻优精度和寻优稳定性上都有明显优势,因此

本文的改进是可行的且比 IWOA 更加有效.

表 6 EGolden-SWOA 与 IWOA 对比实验结果

函数	IWOA ^[15]		EGolden-SWOA	
	平均精度	标准差	平均精度	标准差
f_1	6.25E-125	6.28E-125	0	0
f_2	2.15E-073	3.64E-073	6.69E-202	0
f_3	1.56E-023	1.81E-023	0	0
f_4	7.06E-007	2.18E-006	3.58E-191	0
f_5	27.27950	0.215438	3.75E-09	7.45E-09
f_6	0	0	6.86E-10	1.29E-09
f_7	2.42E-004	4.41E-004	3.25E-05	2.55E-05
f_8	-11423.3746	112.6361	-5.58E+101	1.67E+102
f_9	0	0	0	0
f_{10}	3.02E-015	1.95E-015	8.88E-16	0

4.6 两种改进策略的有效性分析

本文对基本 WOA 算法做了两个方面的改进,即采用精英反向学习策略以及黄金正弦机制改进策略,为了分析两种改进策略的有效性,本节从 4.2 节选择 5 个测试函数,通过函数优化实验将 EGolden-SWOA 与仅采用精英反向学习策略改进的 EWOA 算法和仅引入黄金正弦机制的 Golden-SWOA 算法进行比较,为了避免偶然性带来的结果偏差,算法在每个函数上独立运行 30 次,参数设置与 4.1 节相同,实验结果如表 7 所示.

表 7 采用不同改进策略的 WOA 算法测试函数实验结果

函数	EWOA		Golden-SWOA		EGolden-SWOA	
	平均精度	标准差	平均精度	标准差	平均精度	标准差
f_1	0	0	5.0E-279	0	0	0
f_3	9.7E-53	2.9E-52	2.8E-247	0	0	0
f_7	1.10E-04	6.4E-05	1.0E-04	5.2E-05	3.3E-05	2.6E-05
f_{12}	9.8E-09	2.1E-08	1.6E-02	7.2E-03	1.6E-10	2.6E-10
f_{13}	6.6E-07	1.3E-06	1.3E-01	7.7E-02	8.7E-10	6.7E-10

从表 7 中实验结果可知,采用两种改进策略的 EGolden-SWOA 的寻优精度和寻优稳定性(标准差)要优于采用单一改进策略的 EWOA 和 Golden-SWOA, EWOA 在大多数函数优化上的表现优于 Golden-SWOA. 单独采用精英反向学习策略和黄金正弦机制改进策略对基本 WOA 算法性能的改进虽然有一定效果但效果较为有限,多样的改进策略可以更加有效地提高算法的寻优性能.

4.7 EGolden-SWOA 求解大规模问题的可行性分析

针对实际工程应用中大规模优化问题普遍存在的情况,本文对 EGolden-SWOA 进行高维函数优化求解实验,以证明其应用于求解大规模优化问题的可行性,实验结果如表 8 所示.

从实验结果可以看出,EGolden-SWOA 在求解大规模函数上表现良好,寻优精度与低维函数相比变化不大,均

能得到较理想的寻优结果, f_1 , f_3 , f_9 和 f_{11} 可以直接得到最优值,未陷入“维灾难”,说明该改进算法的稳定性较好.

表 8 EGolden-SWOA 求解高维函数的实验结果

函数	EGolden-SWOA			
	维度 $d = 500$		维度 $d = 1000$	
	平均精度	标准差	平均精度	标准差
f_1	0	0	0	0
f_2	2.64E-176	0	1.20E-177	0
f_3	0	0	0	0
f_4	1.35E-189	0	2.99E-177	0
f_5	3.61E-06	2.85E-06	6.78E-06	1.18E-05
f_6	2.65E-06	6.36E-06	2.00E-07	3.75E-07
f_7	1.07E-04	1.16E-04	9.38E-05	7.08E-05
f_8	-1.32E+97	3.27E+97	-7.90E+107	2.37E+108
f_9	0	0	0	0
f_{10}	8.88E-16	1.97E-31	8.88E-16	1.97E-31
f_{11}	0	0	0	0
f_{12}	3.11E-11	3.92E-11	1.46E-10	2.38E-10
f_{13}	2.23E-08	1.72E-08	2.91E-08	2.41E-08

5 EGolden-SWOA 算法的工程优化应用研究

5.1 压力容器设计问题

压力容器设计问题的目标是使压力容器制作(配对、成型和焊接)成本最小,压力容器的设计如图 3 所示,压力容器的两端都有盖子封顶,头部一端的封盖为半球状. L 是不考虑头部的圆柱体部分的截面长度, R 是圆柱体部分的内壁直径, T_s 和 T_h 分别表示圆柱体部分壁厚和头部的壁厚, L , R , T_s 和 T_h 即为压力容器设计问题的四个优化变量. 问题的目标函数和四个优化约束表示如下:

$$x = [x_1, x_2, x_3, x_4] = [T_s, T_h, R, L]$$

$$\text{Min}f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

subject to:

$$g_1(x) = -x_1 + 0.0193x_3 \leq 0$$

$$g_2(x) = -x_2 + 0.00954x_3 \leq 0$$

$$g_3(x) = -\pi x_3^2 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0$$

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

$$0 \leq x_i \leq 100, i = 1, 2$$

$$10 \leq x_i \leq 200, i = 3, 4$$

分别利用基本 WOA 算法和 EGolden-SWOA 算法对该问题进行求解,两种算法的相关系数设置相同,求解结果如表 9 所示. 并将两种算法的求解结果与文献[16]中的基于粗糙集的乌鸦搜索算法(RCSA)的求解结果加以比较. 从表 9 可以看出,对于压力容器设计问题,EGolden-SWOA 的求解精度和稳定性都优于基本 WOA, RCSA 的求解稳定性虽优于 WOA 以及 EGolden-

SWOA 但求解精度明显劣于其他两种算法. 再者从图 5 可以看出 EGolden-SWOA 在 50 代左右的时候便已经开始收敛于最优值, 但基本 WOA 接近 100 代时才开始收敛到最优值水平, EGolden-SWOA 收敛速度比基本 WOA 快, 所以 EGolden-SWOA 整体寻优性能较好.

表 9 压力容器设计问题两种算法求解结果

算法	最优值	平均值	标准差
EGolden-SWOA	5942.02	5997.58	32.94
WOA	6039.69	6067.89	67.78
RCSA ^[16]	6059.60	6059.8448	0.058

5.2 蝶形弹簧设计问题

蝶形弹簧的最佳设计必须在满足剪应力、浪涌频率、挠曲度等相关指标标准的约束下达到其质量的最小值, 其设计如图 4 所示, 包括三个设计变量, 分别为弹簧线材直径 d 、弹簧线圈平均直径 D 以及弹簧有效线圈数量 N , 其目标函数和约束条件描述如下:

$$x = [x_1, x_2, x_3] = [d \ D \ N]$$

$$\text{Min}f(x) = (x_3 + 2)x_2x_1^2$$

subject to:

$$g_1(x) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0$$

$$g_2(x) = \frac{x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} \leq 0$$

$$g_3(x) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0$$

$$0.05 \leq x_1 \leq 2.00$$

$$0.25 \leq x_2 \leq 1.3$$

$$2.00 \leq x_3 \leq 15.0$$

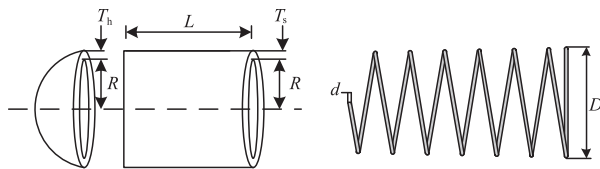


图3 压力容器设计图解

图4 蝶形弹簧设计图解

分别利用基本 WOA 算法和 EGolden-SWOA 算法对该问题进行求解, 两种算法的相关系数设置相同, 求解结果如表 10 所示. 并将两种算法的求解结果与文献 [17] 中的基于共同进化的粒子群算法 (CPSO) 的求解结果加以比较. 从表 10 可以看出, EGolden-SWOA 求解的最优值要领先于 WOA 达到了 0.012667, 总体平均值也优于基本 WOA, 寻优稳定性也略好于基本 WOA; 虽然 CPSO 的最优值好于 EGolden-SWOA, 但从求解平均值和求解稳定性 (标准差) 来看, EGolden-WOA 的求解性能要优于 CPSO. 从图 6 也可以看出 EGolden-SWOA

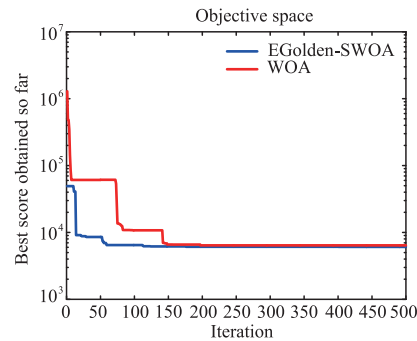


图5 压力容器设计问题两种算法求解结果对比

的收敛速度快于基本 WOA.

表 10 蝶形弹簧设计问题两种算法求解结果

算法	最优值	平均值	标准差
EGolden-SWOA	0.012667	0.012697	3.68E-04
WOA	0.012689	0.012787	3.92E-04
CPSO ^[17]	0.0126474	0.0127300	5.20E-04

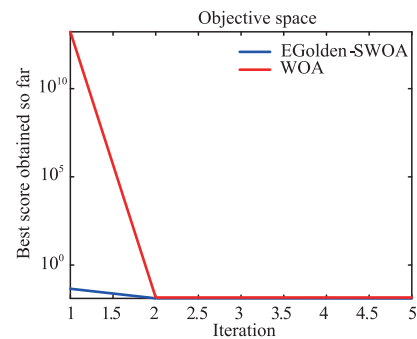


图6 蝶形弹簧设计问题两种算法求解结果对比

6 结束语

鲸鱼优化算法是近年来提出的一种新颖算法, 其寻优性能较好. 为使算法性能更加优秀, 本文提出了一种精英反向学习的黄金正弦鲸鱼算法, 将精英反向学习策略及黄金正弦算法引入基本鲸鱼优化算法, 采用多样的改进策略使得算法性能更加稳定, 并将其应用于函数优化和实际工程设计优化问题, 实验结果表明 EGolden-SWOA 具备一定的优势. 另外, 本文研究表明 EGolden-SWOA 在求解大规模问题中表现良好, 因此下一步的研究重点是将其应用于解决大规模、复杂的多目标优化和实际工程应用等问题.

参考文献

- [1] Mirjalili S, Lewis A. The whale optimization algorithm[J]. Advances in Engineering Software, 2016, 95(05): 51-67.
- [2] Prakash D B, Lakshminarayana C. Optimal siting of capacitors in radial distribution network using whale optimization algorithm[J]. Alexandria Engineering Journal, 2016, 56(04): 499-509.

- [3] Zhao H, Guo S, Zhao H. Energy-Related CO₂ emissions forecasting using an improved LSSVM model optimized by whale optimization algorithm[J]. *Energies*, 2017, 10(07): 874 – 888.
- [4] Saidala R K, Devarakonda N. Improved whale optimization algorithm case study: clinical data of anaemic pregnant woman[A]. *Data Engineering and Intelligent Computing* [C]. Kuala Lumpur, Malaysia; Springer, 2016. 271 – 281.
- [5] Aziz M A E, Ewees A A, Hassanien A E. Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation[J]. *Expert Systems with Applications*, 2017, 83(17): 242 – 256.
- [6] Oliva D, Aziz M A E, Hassanien A E. Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm[J]. *Applied Energy*, 2017, 200(16): 41 – 154.
- [7] Ling Y, Zhou Y, Luo Q. Lévy flight trajectory-based whale optimization algorithm for global optimization[J]. *IEEE Access*, 2017, 5(99): 6168 – 6186.
- [8] Tizhoosh H R. Opposition-based learning: a new scheme for machine intelligence[A]. *International Conference on Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce* [C]. Vienna, Austria; IEEE, 2005. 695 – 701.
- [9] 夏学文, 刘经南, 高柯夫, 李元香, 曾辉. 具备反向学习和局部学习能力的粒子群算法[J]. *计算机学报*, 2015, 38(07): 1397 – 1407.
Xia Xue-wen, Liu Jing-nan, Gao Ke-fu, Li Yuanxiang, Zeng Hui. Particle swarm optimization algorithm with reverse-learning and local-learning ability[J]. *Chinese Journal of Computers*, 2015, 38(07): 1397 – 1407. (in Chinese)
- [10] 魏文红, 周建龙, 陶铭, 袁华强. 一种基于反向学习的约束差分进化算法[J]. *电子学报*, 2016, 44(2): 426 – 436.
Wei Wen-hong, Zhou Jian-long, Tao Ming, Yuan Hua-qiang. Constrained differential evolution using opposition-based learning[J]. *Acta Electronica Sinica*, 2016, 44(2): 426 – 436. (in Chinese)
- [11] 谢承旺, 王志杰, 夏学文. 应用档案精英学习和反向学习的多目标进化算法[J]. *计算机学报*, 2017, 40(3): 757 – 772.
Xie Cheng-wang, Wang Zhi-jie, Xia Xue-wen. Multi-objective evolutionary algorithm based on archive-elite learning and opposition-based learning[J]. *Chinese Journal of Computers*, 2017, 40(3): 757 – 772. (in Chinese)
- [12] 谢承旺, 许雷, 赵怀瑞, 夏学文, 魏波. 应用精英反向学习的多目标烟花爆炸算法[J]. *电子学报*, 2016, 44(5): 1180 – 1188.
Xie Cheng-wang, Xu Lei, Zhao Huai-rui, Xia Xue-wen, Wei Bo. Multi-objective fireworks optimization algorithm using elite opposition-based learning[J]. *Acta Electronica Sinica*, 2016, 44(5): 1180 – 1188. (in Chinese)
- [13] 周新宇, 吴志健, 王晖, 李康顺, 张浩宇. 一种精英反向学习的粒子群优化算法[J]. *电子学报*, 2013, 41(8): 1647 – 1652.
Zhou Xin-yu, Wu Zhi-jian, Wang Hui, Li Kang-shun, Zhang Hao-yu. Elite opposition-based particle swarm optimization[J]. *Acta Electronica Sinica*, 2013, 41(8): 1647 – 1652. (in Chinese)
- [14] Tanyildizi E, Demir G. Golden sine algorithm: a novel math-inspired algorithm[J]. *Advances in Electrical & Computer Engineering*, 2017, 17(2): 71 – 78.
- [15] 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 Wen, Cai Shao-hong, Jiao Jian-jun, et al. Improved whale optimization algorithm for large-scale optimization problems[J]. *Systems Engineering-Theory & Practice*, 2017, 37(11): 2983 – 2994. (in Chinese)
- [16] Hassanien A E, Rizk-Allah R M, Elhoseny M. A hybrid crow search algorithm based on rough searching scheme for solving engineering optimization problems[J]. *Journal of Ambient Intelligence & Humanized Computing*, 2018, 4(2018): 1 – 25.
- [17] He Q, Wang L. An effective co-evolutionary particle swarm optimization for constrained engineering design problems[J]. *Engineering Applications of Artificial Intelligence*, 2007, 20(1): 89 – 99.

作者简介



肖子雅 女, 1994 年 10 月出生于江苏徐州. 2017 进入上海工程技术大学攻读硕士学位, 研究方向为智能算法、项目调度与优化.
E-mail: m18817568572_1@163.com



刘升(通讯作者) 男, 1966 年 01 月出生于江西南昌. 现为上海工程技术大学教授、硕士生导师, 主要研究方向为智能计算, 群智能系统, 进化算法.
E-mail: ls6601@163.com