

融合多策略的改进黏菌算法及工程应用

李梦真, 莫愿斌

(广西民族大学 人工智能学院, 广西 南宁 530006)

摘要:黏菌优化算法(Slime Mould Algorithm, SMA)是根据黏菌个体振荡捕食行为提出的一种新型元启发式算法,因其原理简单被应用于多种复杂的优化问题中,基本的SMA在处理一些较为复杂的问题时仍然存在收敛速度较慢、精度不足、鲁棒性差等劣势。为克服这些缺点,提升原算法性能,提出一种融合Sine混沌映射、 t 分布以及黄金正弦策略的改进黏菌算法(GTSMA)。首先,引入Sine混沌序列初始化种群,提高算法在初始迭代过程中黏菌种群个体的多样性;其次,在黏菌个体更新位置过程中将自由度参数 t 与基本SMA融合,增加算法跳出局部最优的概率;最后,通过与黄金正弦算法相结合,挑选更优秀的黏菌个体,输出最优解。利用基准测试函数、CEC2021测试集将GTSMA与其他算法进行对比,实验结果表明GTSMA在测试过程中鲁棒性、寻优精度和收敛性能都优于其他算法。将GTSMA应用于工程优化问题,进一步验证了GTSMA在处理实际优化问题上的优越性。

关键词:黏菌算法;Sine混沌映射;自适应 t 分布;黄金正弦算法;工程优化问题

中图分类号:TP18

文献标识码:A

文章编号:1673-629X(2024)02-0214-07

doi:10.3969/j.issn.1673-629X.2024.02.031

Improved Slime Mould Algorithm with Fusion of Multiple Strategies and Engineering Application

LI Meng-zhen, MO Yuan-bin

(School of Artificial Intelligence, Guangxi Minzu University, Nanning 530006, China)

Abstract:Slime mould algorithm (SMA) is a new meta heuristic algorithm based on the oscillatory predatory behavior of slime mould individuals. Because of its simple principle, SMA has been applied to a variety of complex optimization problems. The basic SMA still has disadvantages such as slow rate of convergence, insufficient accuracy, and poor robustness when dealing with some more complex problems. To overcome these shortcomings and improve the performance of the original algorithm, we propose an improved slime mould algorithm (GTSMA) that integrates sine chaotic mapping, t -distribution, and golden sine strategy. Firstly, the Sine chaotic sequence is introduced to initialize the population and improve the diversity of the slime mould population during the initial iteration process of the algorithm. Secondly, in the process of updating the position of slime mould individuals, the degree of freedom parameter t is fused with the basic SMA to increase the probability of the algorithm jumping out of local optima. Finally, by combining with the golden sine algorithm, better slime mould individuals are selected to output the optimal solution. The benchmark test function and CEC2021 test set were used to compare the test results of GTSMA with other algorithms. Experimental results show that GTSMA has better robustness, optimization accuracy and convergence performance than that of other algorithms during the test. Applying GTSMA to engineering optimization problems further validates its superiority in handling practical optimization problems.

Key words:slime mould algorithm; Sine chaotic map; adaptive t distribution; golden sine algorithm; engineering optimization problem

0 引言

近年来,随着科学技术的不断发展,计算复杂性的增强,优化问题在现阶段受到广泛关注,利用元启发式算法求解多种复杂优化问题也是当下研究的热点。常

见的元启发式算法有粒子群算法(Particle Swarm Optimization, PSO)^[1]、布谷鸟算法(Cuckoo Search, CS)^[2]、差分进化算法(Differential Evolution, DE)^[3]、灰狼优化算法(Grey Wolf Optimizer, GWO)^[4]、鲸鱼优

收稿日期:2023-04-20

修回日期:2023-08-22

基金项目:国家自然科学基金资助项目(21466008);广西自然科学基金资助项目(2019GXNSFAA185017);广西民族大学科研项目(2021MDKJ004)

作者简介:李梦真(1999-),女,硕士研究生,研究方向为群智能优化算法、智能信息控制;通讯作者:莫愿斌(1969-),男,博士,教授,研究方向为智能信息控制、群智能优化算法。

化算法(Whale Optimization Algorithm, WOA)^[5]、海鸥优化算法(Seagull Optimization Algorithm, SOA)^[6]、麻雀优化算法(Sparrow Search Algorithm, SSA)^[7]、阿基米德优化算法(Archimedes Optimization Algorithm, AOA)^[8]、斑点鬣狗优化算法(Spotted Hyena Optimization, SHO)^[9]等等。

黏菌算法是2020年Li等人^[10]根据黏菌个体振荡捕食行为提出的,建立了由黏菌行为模式启发的数学模型,实验证明该算法具有良好的寻优能力,并广泛应用于解决优化问题;文献[11]提出了一种基于成败历史存档的融合龙格库塔黏菌算法,改进后算法的求解精度和鲁棒性更具竞争力;文献[12]利用精英反向学习和二次插值提升算法全局寻优性能、收敛精度及局部开发能力,帮助算法跳出局部极值;文献[13]提出了一种融合精英策略的SMA,在有固定频率约束的桁架结构尺寸优化问题中取得较好结果。该文提出的融合多策略的改进黏菌算法包含三个改进策略,能针对原算法存在的缺点进行整合性提升。首先,采用Sine映射初始化种群,提高种群多样性;其次,引入自适应变异 t 分布策略,改进原算法容易陷入局部最优的缺点;最后,引入黄金正弦机制来改进算法在迭代后期收敛速度慢、收敛精度低的问题。通过基准测试函数及CEC2021测试函数对改进后的黏菌算法进行性能测试,测试结果明显优于其他对比算法,然后将其应用于工程应用问题,GTSMA都取得了理想的结果。

1 黏菌算法

黏菌算法生物背景新颖,结构清晰,主要利用黏菌觅食的潜力与特性来解决复杂的优化问题。黏菌觅食的潜力最初体现在路径规划中,该算法主要模拟黏菌靠近食物、包围食物和获得食物三个阶段。首先,黏菌根据空气中的气味浓度去接近食物,黏菌个体会根据公式1的规则进行更新移动:

$$X(t+1) = \begin{cases} \text{rand} \times (U_B - L_B) + L_B, \text{rand} < z \\ X_b(t) + v_b \times (W \times X_A - X_B), r < p \\ V_c \times X(t), r \geq p \end{cases} \quad (1)$$

其中, X 表示黏菌的当前位置, L_B 与 U_B 表示搜索范围的上下边界, X_A 与 X_B 表示从所有黏菌个体中随机挑选的两个个体, V_b 是介于 $[-a, a]$ 的参数,迭代次数越多, V_b 的值越趋于0, V_c 从1-0呈线性变化, t 表示黏菌个体当前的迭代次数,权重参数 W 表示黏菌的质量, $X_b(t)$ 表示第 t 次迭代适应度最优的个体位置, r 表示 $[0,1]$ 区间的随机值,其中控制变量 p 和参数 a 的函数表达式如式2、式3:

$$p = \tanh |S(i) - \text{DF}| \quad (2)$$

$$a = \text{arctanh}(1 - \frac{t}{t_{\max}}) \quad (3)$$

其中, $S(i)$ 代表黏菌个体 X 的适应度,而 DF 表示最佳适应度值, t_{\max} 代表个体更新的最大迭代次数。而 W 的表达式如式4:

$$W(\text{smellIndex}) = \begin{cases} 1 + r \times \log(\frac{\text{bF} - S(i)}{\text{bF} - \text{wF}} + 1), \text{condition} \\ 1 - r \times \log(\frac{\text{bF} - S(i)}{\text{bF} - \text{wF}} + 1), \text{other} \end{cases} \quad (4)$$

其中,condition表示适应度排在前一半的个体,other表示余下的种群个体, r 表示 $[0,1]$ 区间内的随机值,bF和wF分别表示当前种群中的最好和最差适应度值。

2 融合多策略的改进黏菌算法

2.1 Sine 混沌映射

丰富的初始种群,对于算法的收敛速度和寻优精度都十分重要。黏菌在初期的位置具有随机性和不确定性,可能会出现种群分布不均匀,容易陷入局部最优。而混沌序列可以用来解决上述问题^[14]。常见的混沌映射有logistic映射、kent映射等,该文采用Sine混沌映射,选取式5产生的初始变量值利用式6映射到黏菌个体上,产生多样性较好的初始种群。

$$Y_{i+1} = \rho \sin(\pi i) \quad (5)$$

$$X_{i,d} = L_d + (1 + Y_{i,d}) \frac{U_d - L_d}{2} \quad (6)$$

其中, $Y_i \in [-1,1]$ 为混沌序列; $i=1,2,\dots,N$ 表示种群规模; ρ 为控制参数; U_d 和 L_d 分别为黏菌个体在第 d 维的上限和下限。

2.2 自适应 t 分布变异策略

t 分布最早被命名为“Student's distribution”,高斯分布(Gaussian Distribution, GD)和柯西分布(Cauchy Distribution, CD)是 t 分布的两个特殊边界分布,该策略融合了高斯分布和柯西分布的优点,初期参数 t 取较小值,符合柯西分布;迭代进行到中后期时, t 取值就会变大, t 分布无限接近高斯分布。具体的黏菌位置更新方式如式7:

$$X(t+1) = \begin{cases} \text{rand} \times (U_B - L_B) + L_B, \text{rand} < z \\ X_b(t) + t(S) \times v_b \times (W \times X_A - X_B), r < p \\ V_c \times X(t) \times t(S), r \geq p \end{cases} \quad (7)$$

其中, t_s 是以SMA迭代次数为自由度的 t 分布,随着 S 增加, t 也会增加, t 分布由开始的柯西分布逐渐转变为高斯分布。

2.3 黄金正弦策略

黄金正弦算法 (golden Sine Algorithm, goldenSA)^[15] 是引用黄金分割系数加入正弦函数设计来提升寻优性能的一种新型智能算法。算法开始时通过随机生成 N 个个体来更新初始空间, 在个体位置更新过程中, 利用黄金分割系数来缩小个体的搜索空间。其位置更新公式如下:

$$V_i^{t+1} = V_i^t \times |\sin(r_1)| + r_2 \times \sin(r_1) \times |x_1 \times D_i^t| - x_2 \times V_i^t \quad (8)$$

其中, V_i^t 表示第 i 个个体在 t 次的迭代位置, $i = 1, 2, \dots, N$; V_i^{t+1} 表示个体在第 $t+1$ 次的迭代位置, D_i^t 是第 i 个个体在 t 次迭代的最优位置; r_1 是 $[0, 2\pi]$ 内的随机数, 决定了个体 i 的移动距离; r_2 是 $[0, \pi]$ 内的随机数, 决定了个体 i 的移动方向; τ 为黄金分割率 $\tau = (\sqrt{5} - 1)/2$; x_1, x_2 是黄金分割系数, $x_1 = -\pi + (1 -$

$\tau) * 2\pi$, $x_2 = -\pi + \tau * 2\pi$ 。上述参数可以扩大黏菌的搜索空间, 同时平衡“搜索”与“开发”的过程, 通过对搜索区域的缩小, 引导黏菌个体不断趋近最优值。

3 GTSMA 优化算法

3.1 算法流程

针对基本 SMA 的缺点, 该文提出融合多策略的改进黏菌算法 (GTSMA)。首先, 引入 Sine 混沌映射提高迭代初期种群的多样性; 其次, 随着迭代次数的增加, 自适应 t 分布变异策略中自由度参数 t 取值也会随之变大, 可以增加算法跳出局部最优的概率; 最后与黄金正弦算法相结合。在算法迭代后期利用黄金分割系数对整个过程进行优化, 改进算法在迭代后期收敛精度较低的问题。GTSMA 流程如图 1 所示。

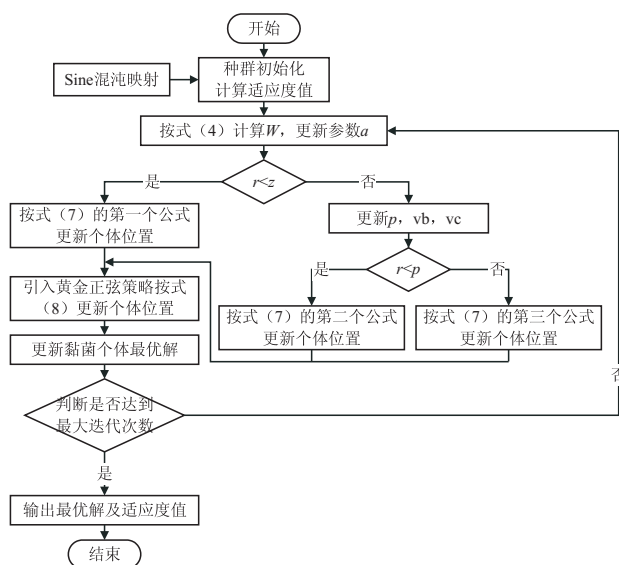


图 1 算法流程

3.2 算法的时间复杂度分析

设黏菌种群规模为 N , 问题维度为 D , 最大迭代次数为 T , 目标函数复杂度为 O_{obj} 。则 GTSMA 在 Sine 混沌映射初始化种群阶段的时间复杂度为 $O(N \times M)$, 适应度值评估和排序的复杂度为 $O(T \times N \times (1 + \log N))$, 权重参数 W 更新时间复杂度为 $O(T)$, 迭代后期融合黄金正弦更新最优位置的时间复杂度为 $O(T \times M)$, 最后阶段位置更新的时间复杂度为 $O(T \times N \times M)$ 。综上所述, GTSMA 的时间复杂度为 $O(T \times N \times (M + O_{obj} + 1 + \log N))$ 。

4 实验仿真与结果分析

4.1 实验环境

所有实验代码均在 Matlab R2018a 上运行, 以保证公平的比较。

4.2 基准测试函数

选取单峰、多峰函数等不同类型的基准测试函数对 GTSMA 进行测试来验证 GTSMA 的性能, 函数的维数、范围和理论最优值如表 1 所示。

4.3 GTSMA 与其他算法对比

论文通过对 WOA, PSO, GWO, CS, SMA, GTSMA 分别在 6 个测试函数独立运行 30 次的结果进行对比来检验文中改进策略是否有效, 运行结果见表 2。

由表 2 数据可知, 针对不同类型的函数, GTSMA 的综合性能最强。对于函数 f_3, f_5, f_6 , GTSMA 的三项指标均优于原始 SMA。由图 2 可以更直观看出 GTSMA 效果更理想。对于函数 f_1, f_2 , 其他比较算法收敛精度较低, GTSMA 相对于原始 SMA 收敛到最优值所需迭代次数更少。对于函数 f_4 , 算法迭代初期该函数的收敛曲线下降明显、收敛速度快。对于函数 f_5 ,

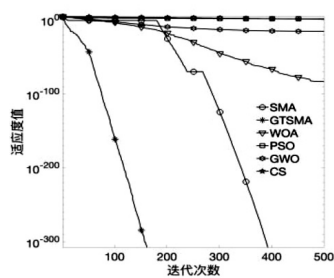
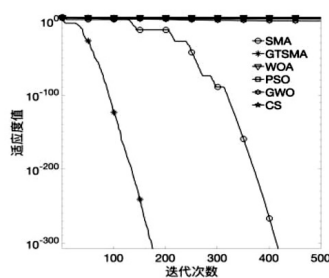
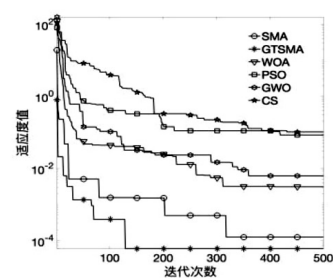
GTSMA 只需五十多次迭代即可获得理论上最优值。说明该算法不仅能够快速收敛,而且还能在后期的迭代对于 f_3, f_6 , 从图中可以看出函数拐点次数明显增多, 代过程中快速逃离局部最优, 提高了算法的勘探能力。

表1 基准测试函数

类型	函数	维度	范围	最优值
f_1	$f_x = \sum_{i=1}^{n-1} x_i^2$	30	$[-100, 100]$	0
f_2	$f_x = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	30	$[-100, 100]$	0
f_3	$f_x = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1)$	30	$[-1.28, 1.28]$	0
f_4	$f_x = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i + 10)]$	30	$[-5.12, 5.12]$	0
f_5	$f_x = -20\exp(-0.2\sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	30	$[-32, 32]$	0
f_6	$f_x = 0.1 \{ \sin^2(3\pi x_i) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] \} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	30	$[-50, 50]$	0

表2 所有比较算法的运行结果

函数	指标	WOA	PSO	GWO	CS	SMA	GTSMA
f_1	最优值	2.463 4e-89	2.900 1	2.332 9e-17	3.462 9	0	0
	平均值	1.590 8e-70	17.627 2	1.113 9e-15	8.172 7	0	0
	标准差	8.712 7e-70	10.055 2	1.597 1e-15	2.474 5	0	0
f_2	最优值	14 805.240 1	149.199 9	0.000 286 41	1 432.968	0	0
	平均值	44 876.930 8	675.050 5	0.091 796	2 256.542 5	0	0
	标准差	13 806.318 5	499.230 6	0.100 91	545.715 9	0	0
f_3	最优值	0.000 107 1	0.007 025	0.002 539 9	0.052 845	6.952 7e-04	4.511 7e-06
	平均值	0.003 336 4	0.037 077	0.006 724 3	0.090 145	0.000 134 94	1.186 1e-04
	标准差	0.003 941	0.013 735	0.003 715 3	0.027 063	0.000 108 61	1.004 7e-04
f_4	最优值	0	62.829 9	1.676 9e-11	76.432 8	0	0
	平均值	1.894 8e-15	112.592 7	7.005 4	106.096 2	0	0
	标准差	1.037 8e-14	28.728 2	7.078 2	16.962	0	0
f_5	最优值	7.751 8e-15	1.174 3	1.625 5e-09	4.094	8.214 6e-16	8.881 8e-16
	平均值	4.440 9e-15	2.484 1	5.278 6e-09	6.889 1	7.593 2e-16	8.881 8e-16
	标准差	2.468 5e-15	0.608 42	3.331 4e-09	2.018 6	4.468 5e-16	0
f_6	最优值	0.127 49	1.360 1	1.051 6	2.297 3	0.016 456	6.749 4e-04
	平均值	0.524 09	2.754	1.458 7	8.602 6	0.005 036 2	0.001 259 3
	标准差	0.281 79	1.037 5	0.229 94	4.115 2	0.006 403 7	0.001 111 6

(a) f_1 比较算法收敛曲线(b) f_2 比较算法收敛曲线(c) f_3 比较算法收敛曲线

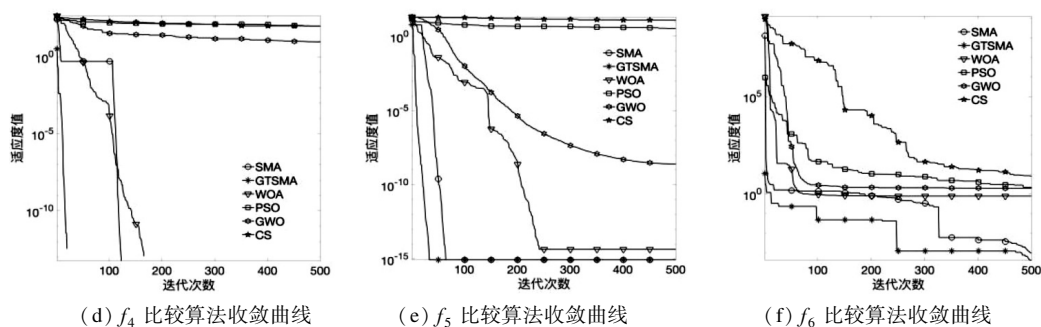


图 2 所有比较算法的收敛曲线

4.4 CEC2021 测试集实验分析

在 CEC2021 测试集上与其他同类型算法进行对比,检验 GTSMA 的综合性能,CEC2021 测试函数见

表 3。

表 3 CEC2021 测试集

函数	特征	范围	f_{\min}	维度
CEC2021-01	UF	$[-100,100]$	100	30
CEC2021-02	MF	$[-100,100]$	1 100	30
CEC2021-03	MF	$[-100,100]$	700	30
CEC2021-04	MF	$[-100,100]$	1 900	30
CEC2021-05	HF	$[-100,100]$	1 700	30
CEC2021-06	HF	$[-100,100]$	1 600	30
CEC2021-07	HF	$[-100,100]$	2 100	30
CEC2021-08	CF	$[-100,100]$	2 200	30
CEC2021-09	CF	$[-100,100]$	2 400	30
CEC2021-10	CF	$[-100,100]$	3 000	30

将 GTSMA 与标准 SMA, SOA, WOA, DE 以及 GWO 进行对比。实验参数取种群规模为 $N=30$, 维度 $d=30$, 最大迭代次数为 500。表 4 展示了算法在运行过程中取得的最优平均值和标准差。由表 4 中算法的

运行结果可知, GTSMA 在 10 个测试函数中平均值和标准偏差均为 0, 根据上述实验结果分析, GTSMA 相比同类算法优势更大。

表 4 所有比较算法 CEC2021 测试集实验结果

函数	指标	GTSMA	SMA	SOA	WOA	DE	GWO
CEC2021-01	Mean	0	0	3.386 8e-32	3.282 6e-38	1.055e-08	1.658 e-28
	Std	0	0	1.481 4e-30	1.036 4e-37	3.339e-08	5.248 7e-28
CEC2021-02	Mean	0	1.616 6e-4	7.530 5e-09	4.357 3	36.914	4.763 1
	Std	0	1.011 4e-3	4.124 6e-08	37.812 5	34.281	7.624 4
CEC2021-03	Mean	0	0	2.993 9e-24	1.054 3e-18	26.372 8	19.099 4
	Std	0	0	1.167 2e-23	4.563 8e-16	5.339 8	19.099 4
CEC2021-04	Mean	0	0	4.886 7e-13	2.506 8e-05	12.247 7	2.202 9
	Std	0	0	2.676 6e-12	7.457 6e-05	7.522 4	2.274 7
CEC2021-05	Mean	0	0.728 47	1.090 5e-17	5.472 6e-13	0.558 76	6.427 8e-10
	Std	0	17.621 3	5.833 5e-17	1.776 8e-12	32.391 9	7.869 5e-09
CEC2021-06	Mean	0	0	0.027 764	1.372 3	20.513 2	3.437 9
	Std	0	0	0.068 51	2.250 6	47.117 7	5.747 5

续表 4

函数	指标	GTSMA	SMA	SOA	WOA	DE	GWO
CEC2021-07	Mean	0	0	5.758 8e-05	2.632	5.135 1	0.621 01
	Std	0	0	0.014 301	4.782 9	33.725 7	13.726 7
CEC2021-08	Mean	0	7.530 5e-09	1.233 6e-9	1.954 7e-10	22.395	5.579 5e-11
	Std	0	4.124 6e-08	6.756 6e-10	8.011 3e-10	66.235 1	2.164 8e-09
CEC2021-09	Mean	0	1.185 9	524.065	1.189 9e-16	2.208 4	2.340 4e-16
	Std	0	3.281 7	309.347 5	1.293 5e-15	5.089 8	1.261 1e-15
CEC2021-10	Mean	0	5.758 8e-05	32.462 9	1.539 e-06	35.133	19.736 7
	Std	0	0.014 301	29.63	0.008 270 4	20.882 5	26.642 9

5 GTSMA 在工程优化问题的应用

拉压弹簧设计问题作为优化工程应用问题中的经典案例,结构模型如图 3 所示,它的目标是在满足一定约束条件下令弹簧质量 $f(x)$ 最小。该问题主要包括弹簧金属丝直径 $D(x_1)$ 、弹簧圈平均直径 $D(x_2)$ 以及弹簧的有效圈数 $N(x_3)$ 三个设计变量以及最小挠度、剪切应力、振荡频率以及外径限制四个不等式约束。其数学模型如式 9 所示。



图 3 压力弹簧模型

约束条件为:

$$\min f(x) = (x_3 + 2)x_2x_1^2$$

$$\begin{aligned} g_1(x) &= 1 - \frac{x_2^3x_3}{71\,785x_1^4} \leq 0 \\ g_2(x) &= \frac{4x_2^2 - x_1x_2}{12\,566(x_2x_1^3 - x_1^4)} + \frac{1}{5\,108x_1^2} - 1 \leq 0 \\ g_3(x) &= 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0 \\ g_4(x) &= \frac{x_1 + x_2}{1.5} - 1 \leq 0 \end{aligned} \tag{9}$$

其中,变量 x_1, x_2, x_3 的取值范围如下:

$$0.05 \leq x_1 \leq 2, 0.25 \leq x_2 \leq 1.3, 2 \leq x_3 \leq 15$$

用 GTSMA 和标准 SMA 求解拉压弹簧设计问题,并与 WOA^[5],SSA^[7],AOA^[8],混沌粒子群算法(Chaos particle swarm optimization algorithm,CPSO)^[16]取得的最优值进行对比。为确保数据的准确性,用于比较算法的数据均取自于对应的参考文献。其中对比结果如表 5 所示。

表 5 拉压弹簧设计问题优化结果

算法	x_1	x_2	x_3	$f(x)$
GTSMA	0.050 0	0.317 0	10.328 1	0.012 66
SMA	0.052 5	0.376 7	14.028 8	0.012 68
WOA ^[5]	0.051 2	0.345 2	12.004 0	0.012 67
SSA ^[7]	0.051 2	0.345 2	12.004 0	0.012 67
AOA ^[8]	0.050 0	0.317 1	14.079 7	0.012 75
CPSO ^[16]	0.051 7	0.357 6	11.244 5	0.012 67

从表 5 的数据可知,在拉压弹簧设计问题寻优结果中,GTSMA 的寻优结果为 0.012 66,结果均优于其他四种优化算法。因此,GTSMA 的寻优能力更强,可以有效地处理拉压弹簧工程问题。

6 结束语

在原始 SMA 的基础上,提出了一种融合多策略的改进黏菌算法,迭代初期在种群初始化过程中引入

Sine 混沌序列提高种群多样性;然后引入自适应 t 分布策略避免黏菌个体陷入局部最优;最后融合黄金正弦算法思想更新个体位置,提高算法的收敛精度及运行速度。在基准测试函数、CEC2021 测试集以及实际工程设计优化问题上均取得了满意效果,表明了 GTSMA 的可操作性和适用性。今后的工作将继续研究改进的优化策略,提高该算法的运行速度并将其应用到更复杂的优化问题中。

参考文献:

- [1] 刘道文,杨拥军. 基于混沌局部搜索的粒子群算法及其应用[J]. 计算机技术与发展,2021,31(4):216-220.
- [2] YANG X S,DEB S. Cuckoo search:recent advances and applications[J]. Neural Computing and Applications,2014,24(1):169-174.
- [3] ABBASS H A,SARKER R. The Pareto differential evolution algorithm[J]. International Journal on Artificial Intelligence Tools,2002,11(4):531-552.
- [4] KAUR S,AWASTHI L K,SANGAL A L, et al. Tunicate swarm algorithm;a newbio-inspired based metaheuristic paradigm for global optimization[J]. Engineering Applications of Artificial Intelligence,2020,90:103541.
- [5] 刘景森,马义想,李煜. 改进鲸鱼算法求解工程设计优化问题[J]. 计算机集成制造系统,2021,27(7):1884-1897.
- [6] 岳文静,孙鹏,陈志. 基于改进海鸥算法的认知无人机网络频谱分配[J]. 计算机技术与发展,2021,31(9):7-12.
- [7] MIRJALILI S,GANDOMI A H,MIRJALILI S Z, et al. Salp swarm algorithm;a bio-inspired optimizer for engineering design problems[J]. Advances in Engineering Software,2017,114:163-191.
- [8] ABUALIGAH L,DIABAT A,MIRJALILI S, et al. The arithmetic optimization algorithm[J]. Computer Methods in Applied Mechanics and Engineering,2021,376:113609.
- [9] DHIMAN G,KUMAR V. Spotted hyena optimizer;a novel bio-inspired based metaheuristic technique for engineering applications[J]. Advances in Engineering Software,2017,114:48-70.
- [10] LI S,CHEN H,WANG M, et al. Slime mould algorithm;a new method for stochastic optimization[J]. Future Generation Computer Systems,2020,111(12):300-323.
- [11] 刘宇淦,刘升. 成败历史存档的融合龙格库塔-黏菌算法[J]. 计算机工程与应用,2022,58(17):61-71.
- [12] 郭雨鑫,刘升,张磊,等. 精英反向与二次插值改进的黏菌算法[J]. 计算机应用研究,2021,38(12):3651-3656.
- [13] KAVEH A, HAMEDANI K B, KAMALINEJAD M. Improved slime mould algorithm with elitist strategy and its application to structural optimization with natural frequency constraints[J]. Computers & Structures,2022,264:106760.
- [14] 回立川,陈雪莲,孟嗣博. 多策略混合的改进麻雀搜索算法[J]. 计算机工程与应用,2022,58(16):71-83.
- [15] 张磊,刘升,高文欣,等. 精英反向黄金正弦海洋捕食者算法[J]. 计算机工程与科学,2023,45(2):355-362.
- [16] 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.