



改进平衡优化器算法在约束优化问题中的应用

李守玉, 何庆⁺, 陈俊

贵州大学 大数据与信息工程学院, 贵阳 550025

+ 通信作者 E-mail: qhe@gzu.edu.cn

摘要:针对平衡优化器算法存在种群勘探与开发难以平衡、粒子进化信息不足、容易出现早熟现象等问题, 提出改进的平衡优化器算法。首先, 根据算法优化进行的迭代阶段采用正弦池策略动态地平衡勘探与开发能力, 迭代前期通过固定角频率的正弦递减进行大范围的全局勘探, 扩大算法探索搜索空间中未知区域, 增强发现潜藏优质粒子的能力; 迭代后期通过变化角频率的正弦递增进行局部开发使勘探与开发自适应平衡, 提高算法优化精度。其次, 自适应优先引力策略引入当前最优粒子信息克服粒子进化信息匮乏的问题, 然后通过融入均匀分布和贝塔分布共同作用丰富种群粒子进化信息, 提高粒子之间的信息交换速率, 增强粒子逃离局部最优的能力, 达到引导种群向全局最优方向快速收敛目的。最后, 使用16个基准测试函数、CEC2017函数集、Friedman检验、Wilcoxon秩和检验以及2个现实中的工程约束优化问题测试所提算法的寻优能力。实验结果表明, 相比其他新提出的智能算法, 所提算法具有更高的优化精度和更快的收敛速度。

关键词:平衡优化器算法; 勘探与开发; 约束工程优化问题

文献标志码:A **中图分类号:**TP301.6; TP399

Application of Improved Equilibrium Optimizer Algorithm to Constrained Optimization Problems

LI Shouyu, HE Qing⁺, CHEN Jun

College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, China

Abstract: Aiming at the problems of the equilibrium optimizer algorithm, such as difficult balance between population exploration and exploitation, insufficient information of particle evolution and prematurity, an improved equilibrium optimizer algorithm is proposed. Firstly, in the iterative stage optimized by the algorithm, the sinusoidal pool strategy is used to balance the exploration and development capabilities dynamically. In the early stage of iteration, a large range of global exploration is carried out through the sinusoidal decrease of fixed angular frequency to expand the algorithm to explore unknown areas in the search space and enhance the ability of discovering potential high-quality particles. At the end of iteration, local exploitation is carried out by sinusoidal increase of changing angular frequency to balance exploration and exploitation adaptively and improve the optimization accuracy of the algorithm. Secondly, the adaptive priority gravity strategy introduces the current optimal particle information to overcome the lack of evolution information, enriches the evolution information of the population particles by incorporating the

基金项目:贵州省科技计划项目重大专项(黔科合重大专项字[2018]3002); 贵州省公共大数据重点实验室开放课题(2017BDKFJJ004); 贵州大学培育项目(黔科合平台人才[2017]5788); 贵州省科学技术厅项目(黔科合基础-ZK[2021]一般335)。

This work was supported by the Major Special Project of Science and Technology Planning Project of Guizhou Province ([2018]3002), the Open Project of Guizhou Key Laboratory of Public Big Data (2017BDKFJJ004), the Cultivation Project of Guizhou University ([2017]5788), and the Science and Technology Foundation of Guizhou Province (ZK[2021]General335).

收稿日期:2021-08-04 **修回日期:**2021-09-24

uniform distribution and beta distribution together, improves the information exchange rate between particles, enhances the escape of the particles from the local area, and achieves the goal of guiding the population to converge rapidly towards the global optimum. Finally, 16 benchmark functions, CEC2017 functions, Friedman test, Wilcoxon rank sum test and two real-world engineering constraint optimization problems are used to test the optimization ability of the proposed algorithm. Experimental results show that the proposed algorithm has higher optimization accuracy and faster convergence speed compared with other new proposed intelligent algorithms.

Key words: equilibrium optimizer algorithm; exploration and exploitation; constrained engineering optimization problem

在实际工程应用领域的优化目标包括从连续到离散、从单目标到多目标、从有约束到无约束,面对这些复杂的优化问题,传统优化算法仅能求取局部最优值且优化结果严重依赖初始值^[1]。研究者们从生物和自然物理现象得到启发,提出的元启发式算法成为了一种有效且实用的优化方案,并成功应用于NP-Hard^[2-3]、图像分割^[4]、PID参数控制^[5]等问题。

平衡优化器算法(equilibrium optimizer, EO)是由Faramarzi等人于2020年提出的基于物理的元启发式算法^[6],其原理是将每一个粒子(解)及其浓度(位置)看作独立个体,然后根据均衡候选解的浓度随机更新个体,最终达到平衡状态(最优解)。它具备参数少、耗时少及易于实现等优点,并且寻优能力优于粒子群算法^[7]、遗传算法^[8]、差分进化算法^[9]、萤火虫算法^[10]。但标准EO算法与其他智能算法一样,存在算法容易陷入局部最优、收敛较慢等问题,其寻优能力仍有待提高。众多学者对其深入研究,提出了一些有效的改进方法。Fan等人^[11]通过反向学习和新的浓度更新公式,提高算法寻优精度。Sayed等人^[12]通过引入混沌映射构建稳定的搜索机制,提高算法的特征选择效率。Dinkar等人^[13]通过拉普拉斯分布的随机游动更新候选解浓度,然后利用反向学习加速开发,使算法快速收敛。Ahmed等人^[14]利用自动学习机寻找合适的参数值,便于解决高维的特征选择问题。Kardani等人^[15]利用突变机制提高搜索能力,避免陷入局部最优,并与极限学习机和人工神经网络结合对致密碳酸盐岩渗透率进行预测。Shankar等人^[16]通过对立学习的更新机制生成最优解来寻找最优空间。虽然上述方法取得不错的效果,但EO的优化精度仍有待提高。

针对以上问题,提出改进的平衡器优化算法,利用正弦变化规律自适应平衡勘探与开发;引入当前最优粒子信息到粒子种群,促进种群粒子信息交流以及吸引其他粒子快速收敛到最优解,进一步提高

和平衡勘探与开发的能力,增强算法逃离局部最优的能力,避免早熟现象。

1 平衡优化器算法

1.1 种群初始化

标准EO在解空间中随机初始化粒子的位置。种群初始化公式如下:

$$C_{i,k} = lb + r_k * (ub - lb) \quad (1)$$

其中, $C_{i,k}$ 是第 k 个粒子的初始位置, r_k 是 $[0, 1]$ 的随机向量, ub 和 lb 分别是搜索空间的上下界。 k 的变化范围为 $[1, N]$, N 为种群粒子数量。

1.2 候选解和均衡池

平衡状态是算法的最终收敛状态。迭代前期种群没有达到平衡状态的依据,只有通过候选解为种群提供依据。种群初始化后,计算每个粒子适应度值,并根据适应度值的大小获得四个候选解。另外,通过它们得到平均候选解,进而构建均衡池。四个候选解有助于开发,而平均候选解有助于勘探。

$$C_p = \{C_{eq1} \sim C_{eq4}, C_{ave}\} \quad (2)$$

$$C_{eq} = Rand(C_p) \quad (3)$$

每个粒子都以相同的概率在候选粒子中进行随机选择来更新其浓度。在优化过程结束之前,每个粒子都将经历浓度更新过程。

1.3 浓度更新

标准EO为了进一步平衡勘探与开发,利用指数项 F 进行调整,数学描述如下:

$$F = a_1 \cdot \text{sign}(r - 0.5) (e^{-\lambda m} - 1) \quad (4)$$

$$m = \left(1 - \frac{t}{T}\right)^{a_2 \frac{t}{T}} \quad (5)$$

其中, a_1 和 a_2 均为常数, λ 是 $[0, 1]$ 之间的随机向量, $\text{sign}(r - 0.5)$ 用来控制勘探与开发的方向, m 是一个随着迭代增加而降低的变量。 t 为当前迭代次数, T 为最大迭代次数。另外,利用生成率 (G) 改进开发阶

段进而提高提供解的精度。数学表达如下:

$$\mathbf{G} = \mathbf{G}_0 e^{-\lambda(m-m_0)} \quad (6)$$

$$m_0 = \frac{1}{\lambda} \ln[-a_1 \text{sign}(r-0.5)(1-e^{-\lambda m})] + m \quad (7)$$

$$\mathbf{G}_0 = \mathbf{GCP}(\mathbf{C}_{eq} - \lambda \mathbf{C}) \quad (8)$$

$$\mathbf{GCP} = \begin{cases} 0.5R_1, & R_2 \geq GP \\ 0, & R_2 < GP \end{cases} \quad (9)$$

其中, \mathbf{GCP} 为 \mathbf{G} 的控制参数, 主要控制粒子是否使用 \mathbf{GCP} 来更新状态。此外, \mathbf{GCP} 表达形式又由生成概率(GP)决定。 m_0 是为降低搜索速度的同时提高算法的勘探与开发能力。 \mathbf{C}_{eq} 是从均衡池随机选出的一个解, R_1 和 R_2 是 $[0, 1]$ 的随机数。标准EO浓度更新定义如下, 其中 V 表示单位体积。算法1为标准EO算法的伪代码。

$$\mathbf{C} = \mathbf{C}_{eq} + (\mathbf{C} - \mathbf{C}_{eq})\mathbf{F} + \frac{\mathbf{C}}{\lambda V}(1 - \mathbf{F}) \quad (10)$$

算法1 平衡优化器算法

1. Initialization the position of particle's populations
2. Define initial parameter $a_1 = 2, a_2 = 1, GP = 0.5, V = 1$
3. Set four equilibrium candidates' fitness a large number
4. While $t < T$
5. For $k = 1:N$
6. calculate the fitness of each particle
7. find best-so-far particles $\mathbf{C}_{eq1} \sim \mathbf{C}_{eq4}$
8. end for
9. calculate average particle \mathbf{C}_{ave} and construct equilibrium pool \mathbf{C}_p
10. implement memory saving
11. For $k = 1:N$
12. select randomly candidate from equilibrium pool
13. generate randomly two vectors λ and r
14. compute \mathbf{F} by Eq.(4) and Eq.(5)
15. compute \mathbf{GCP} by Eq.(9)
16. compute \mathbf{G}_0 by Eq.(8)
17. update \mathbf{C} by Eq.(10)
18. end for
19. $t++$
20. end while

2 改进平衡优化器算法

2.1 正弦池策略

标准EO指出, 当粒子之间距离较远时, 平均候选解有助于迭代初期发现未知的解空间。因此, 增

强平均候选解的勘探与开发能力将提高种群粒子的寻优能力。

元启发式算法的核心是迭代初期对解空间内的大部分区域进行勘探, 然后勘探阶段逐渐转化为开发阶段的寻优过程。因此, 为了提高粒子的勘探与开发能力, 本文提出正弦池策略。该策略主要根据迭代阶段采用不同寻优模式动态协调粒子的全局勘探与局部开发。迭代前期, 固定角频率正弦递减模式中固定角频率和正弦递减特性有助于粒子搜索空间进行大范围快速搜索, 寻找潜在优质解, 同时逐渐向开发阶段过渡; 迭代后期, 变化角频率正弦递增模式主要借助柯西分布的“长跳”特性使角频率发生变化和正弦递增性质, 使粒子的移动步长多变, 有助于粒子持续寻优并且在优质解附近空间进行精细的深度开发。正弦池策略数学公式如下:

$$S = \begin{cases} \frac{1}{2} \left(\sin(2\pi\omega_1 t) \cdot \frac{T-t}{T} + 1 \right), & 0 \leq t \leq \frac{T}{2} \\ \frac{1}{2} \left(\sin(2\pi\omega_2 t) \cdot \frac{t}{T} + 1 \right), & \frac{T}{2} \leq t \leq T \end{cases} \quad (11)$$

$$\omega_2 = \int f_C(x; \varphi, \gamma) dx \quad (12)$$

$$f_C(x; \varphi, \gamma) = \frac{1}{\pi} \cdot \frac{\gamma}{(x - \varphi)^2 + \gamma^2} \quad (13)$$

其中, ω_1 为固定角频率设为 0.5, ω_2 是包含种群信息的柯西分布, γ 为搜索维度 D , φ 为种群大小 N 。式(13)为柯西分布的概率密度函数。另外, 式(14)~(16)主要利用正弦池策略对平均候选解作用, 得到新的平均候选解 N_{ave} 。然后将其与 $\mathbf{C}_{eq1} \sim \mathbf{C}_{eq4}$ 构成新的均衡池 N_p , N_p 从中随机取出一个候选解得到 \mathbf{C}_{eq} 。

$$N_{ave} = S\mathbf{C}_{ave} \quad (14)$$

$$N_p = \{\mathbf{C}_{eq1} \sim \mathbf{C}_{eq4}, N_{ave}\} \quad (15)$$

$$\mathbf{C}_{eq} = \text{Rand}(N_p) \quad (16)$$

2.2 自适应优先引力策略

标准EO寻找平衡状态的过程中, 当前粒子仅根据均衡池中随机选择的候选解更新浓度, 虽然能为当前粒子提供部分信息, 但因粒子进化信息不足且有用信息量少, 其他粒子无法快速收敛到当前最优粒子和全局最优粒子。本文受万有引力启发, 提出自适应优先引力策略。简单来说, 自适应优先引力策略是先找到当前最优粒子, 将它看成吸引力最大的粒子, 它能够向其他粒子传递多且有用的信息吸引其向自身靠近, 引导种群进化。另外, 当前最优粒子可能处于局部最优, 易导致算法早熟。为避免此

种情况发生,采用均匀分布 $U(a,b)$ 对当前最优粒子进行扰动避免陷入局部最优,再用 $Beta(\alpha,\beta)$ 分布促进其他粒子与当前最优粒子之间的信息流动,增大彼此之间的引力作用。均匀分布和 Beta 分布的概率密度函数定义如下:

$$f_U(x; a, b) = \begin{cases} \frac{1}{b-a}, & a < x < b \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

$$f_B(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} \quad (18)$$

其中, $a, b \in R$, $\alpha, \beta > 0$ 。由概率密度函数得到两者的分布函数:

$$U = \int f_U(x; a, b) dx \quad (19)$$

$$B = \int f_B(x; \alpha, \beta) dx \quad (20)$$

为了充分传递粒子之间信息,采用标准均匀分布 $U(0,1)$, B 为 Beta 分布, α 设为 1, β 为式(21)。自适应优先引力策略数学描述, P_b 为当前最优粒子的位置:

$$P_b = U(0,1)P_o + B(1,\beta)C \quad (21)$$

$$\beta = e^{\frac{10t}{T}} \quad (22)$$

综上所述,首先通过正弦池策略提高平均候选解的勘探与开发能力,进一步增强种群粒子寻优能力。然后,自适应优先引力策略添加当前最优粒子的位置信息到种群中丰富其他粒子的进化信息,加强粒子间信息交流,吸引其他粒子向当前最优粒子收敛,进而快速收敛到全局最优解。本文所提的 MDSGEO (multi-distribution sinusoidal gravitation equalization optimizer) 执行步骤如算法 2 所示。

算法 2 改进平衡优化器算法

1. Initialization the position of particle's populations
2. Define initial parameter $a_1 = 2, a_2 = 1, GP = 0.5, V = 1$
3. Set four equilibrium candidates' fitness a large number
4. While $t < T$
5. for $k = 1:N$
6. calculate the fitness of each particle
7. find $C_{eq1} \sim C_{eq4}$
8. end for
9. calculate C_{ave}
10. implement memory saving
11. for $k = 1:N$
12. generate randomly two vectors λ and r
13. compute F by Eq.(4) and Eq.(5)
14. compute G by Eq.(6) to Eq.(9)

15. execute sinusoidal pool strategy by Eq.(11) to Eq.(13)
16. gain C_{eq} by Eq.(14) and Eq.(15)
17. execute prioritized gravity strategy by Eq.(16) to Eq.(18)
18. update C using Eq.(10)
19. end for
20. $t++$
21. end while

2.3 时间复杂度分析

假设种群数量为 N , 目标问题维度为 D , 最大迭代次数为 T , 评估目标问题适应度为 C 。其中“ O ”常用来表示复杂度。

标准 EO 的时间复杂度: 初始参数定义 $O(1)$, 种群初始化 $O(ND)$, 计算目标问题适应度 $O(TCN)$, 记忆存储 $O(TN)$, 浓度更新 $O(TND)$ 。因此, 根据复杂度计算规则, 标准 EO 总时间复杂度为:

$$O(\text{EO}) \cong O(T(CN + ND)) \quad (23)$$

由标准 EO 的复杂度可知, MDSGEO 仅添加正弦池策略 $O(1 + TND)$, 自适应优先引力策略 $O(ND + TND)$ 。因此, MDSGEO 总时间复杂度为:

$$O(\text{MDSGEO}) \cong O(T(ND + CN + N)) \quad (24)$$

综上分析, MDSGEO 复杂度与标准 EO 的复杂度同一级别, 并未增加额外的时间复杂度, 说明 MDSGEO 并未牺牲空间来提升算法性能。

3 仿真实验与分析

3.1 实验相关设置

实验环境为 Windows 7, 64 位操作系统, CPU 为 Intel Core i5-6500H, 主频 3.2 GHz, 内存 8 GB, 算法基于 MATLAB2020a 编写。

对 16 个基准测试函数进行仿真实验, 表 1 中测试函数不仅包含测试算法局部开发能力的单峰函数, 还包括验证算法全局勘探能力的多峰函数。具体有单峰可分(US)、单峰不可分(UN)、多峰可分(MS)、多峰不可分(MN)和固定维度(FD)等多类型函数。

同时, 元启发式算法的参数设置对于算法寻优将产生重要影响。因此, 各算法的参数设置如表 2 所示。

3.2 实验结果与分析

为了充分验证 MDSGEO 寻找平衡状态的有效性及其鲁棒性, 将 MDSGEO 与加入正弦池策略(记为 SEO)、加入自适应优先引力策略(记为 GEO)、灰狼

表1 基准测试函数

Table 1 Benchmark functions

F	Name	Range	Type	F_{min}
F_1	Sphere	[-100,100]	US	0
F_2	Schwefel 2.22	[-10,10]	UN	0
F_3	Schwefel 1.2	[-100,100]	UN	0
F_4	Schwefel 2.21	[-100,100]	UN	0
F_5	Matyas	[-10,10]	UN	0
F_6	Schaffer	[-100,100]	MN	0
F_7	Quartic	[-1.28,1.28]	US	0
F_8	Schwefel 2.26	[-500,500]	MN	-418.982 6*D
F_9	Rastrigin	[-5.12,5.12]	MS	0
F_{10}	Ackley	[-32,32]	MN	0
F_{11}	Griewank	[-600,600]	MN	0
F_{12}	Apline	[-10,10]	MN	0
F_{13}	Salomon	[-100,100]	MN	0
F_{14}	Six-Hump	[-5,5]	FD	-1.031 628 5
F_{15}	Branin	[-5,10]×[0,15]	FD	0.398
F_{16}	Goldstein-Price	[-2,2]	FD	3

表2 主要参数

Table 2 Main parameters

Algorithm	Parameters
MPA	$FADs = 0.2, P = 0.5$
SCA	$a = 2$
TSA	—
GWOCS	—
EO	$a_1 = 2, a_2 = 1, GP = 0.5$
SEO	$a_1 = 2, a_2 = 1, GP = 0.5, \omega_1 = 0.5$
GEO	$a_1 = 2, a_2 = 1, GP = 0.5$
MDSGEO	$a_1 = 2, a_2 = 1, GP = 0.5, \omega_1 = 0.5$

算法与布谷鸟算法混合(记为GWOCS)、SCA(sine cosine algorithm)^[17]、MPA(marine predators algorithm)^[18]和TSA(tunicate swarm algorithm)^[19]进行对比。同时,为了保证对比的公平性,种群粒子数量 N 设为30,最大迭代次数设为500,总评估次数15 000。此外,所有算法在16个基准测试函数上独立运行30次,并取30次的平均值和标准差作为最终的评估指标。具体实验数据如表3~表5所示,其中Ave表示平

表3 30维实验结果对比

Table 3 Comparison of experimental results in 30 dimensions

F	Index	GWOCS	TSA	SCA	MPA	EO	SEO	GEO	MDSGEO
F_1	Ave	3.21E-28	1.82E-21	1.88E+01	3.70E-23	1.66E-40	4.71E-173	0.00E+00	0.00E+00
	Std	7.61E-28	2.95E-21	3.30E+01	5.35E-23	6.06E-40	0.00E+00	0.00E+00	0.00E+00
F_2	Ave	8.69E-16	1.08E-12	9.88E-02	2.09E-12	8.36E-23	8.69E-89	2.54E-159	1.18E-180
	Std	1.40E-15	1.31E-12	1.50E-01	1.43E-12	1.14E-22	3.37E-88	6.39E-159	0.00E+00
F_3	Ave	1.03E-05	2.61E-04	7.75E+03	1.04E-04	1.36E-08	9.84E-121	2.22E-249	6.58E-306
	Std	4.74E-05	5.69E-04	5.49E+03	1.58E-04	4.06E-08	5.38E-120	0.00E+00	0.00E+00
F_4	Ave	3.98E-07	3.09E-01	3.44E+01	3.57E-09	3.10E-10	1.37E-79	1.38E-151	3.19E-176
	Std	5.00E-07	3.19E-01	1.41E+01	1.60E-09	3.82E-10	5.46E-79	3.90E-151	0.00E+00
F_5	Ave	9.44E-101	1.05E-74	4.38E-51	6.89E-50	1.34E-130	5.54E-240	0.00E+00	0.00E+00
	Std	3.59E-100	5.44E-74	2.40E-50	3.78E-49	7.31E-130	0.00E+00	0.00E+00	0.00E+00
F_6	Ave	5.83E-03	9.39E-03	1.93E-14	2.43E-10	2.59E-03	0.00E+00	0.00E+00	0.00E+00
	Std	4.84E-03	1.77E-03	1.06E-13	1.33E-09	4.37E-03	0.00E+00	0.00E+00	0.00E+00
F_7	Ave	1.97E-03	1.16E-02	1.46E-01	1.40E-03	1.28E-03	3.10E-04	4.88E-04	2.57E-04
	Std	1.35E-03	4.74E-03	2.52E-01	8.78E-04	5.87E-04	2.59E-04	3.76E-04	1.69E-04
F_8	Ave	-1.10E+04	-6.14E+03	-3.69E+03	-8.83E+03	-8.83E+03	-8.71E+03	-8.74E+03	-8.88E+03
	Std	1.57E+03	5.35E+02	2.40E+02	4.29E+02	5.60E+02	7.38E+02	7.50E+02	6.69E+02
F_9	Ave	1.85E+00	1.82E+02	4.00E+01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	4.19E+00	3.41E+01	4.23E+01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{10}	Ave	9.09E-14	1.67E+00	1.47E+01	1.45E-12	8.23E-15	8.88E-16	8.88E-16	8.88E-16
	Std	1.65E-14	1.61E+00	8.00E+00	9.71E-13	1.60E-15	0.00E+00	0.00E+00	0.00E+00
F_{11}	Ave	2.40E-03	1.08E-02	8.76E-01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	5.96E-03	1.01E-02	3.63E-01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{12}	Ave	4.67E-04	2.87E+01	2.33E+00	5.90E-14	1.59E-08	3.63E-91	6.82E-162	1.30E-184
	Std	6.68E-04	7.11E+00	4.29E+00	7.10E-14	8.72E-08	1.19E-90	2.16E-161	0.00E+00
F_{13}	Ave	4.37E-05	1.80E-04	1.21E-02	3.75E-18	8.35E-06	3.08E-05	3.68E-16	9.76E-33
	Std	7.82E-05	2.61E-04	1.95E-02	2.04E-17	2.75E-05	5.49E-05	9.62E-16	5.12E-32
Avg.Rank		5.62	6.81	7.23	4.83	4.10	3.08	2.56	1.79
Rank		6	7	8	5	4	3	2	1

表4 100维实验结果对比

Table 4 Comparison of experimental results in 100 dimensions

F	Index	GWOCS	TSA	SCA	MPA	EO	SEO	GEO	MDSGEO
F_1	Ave	2.20E-13	1.75E-10	1.24E+04	2.00E-19	2.86E-29	1.48E-127	1.21E-307	0.00E+00
	Std	1.56E-13	1.56E-10	8.23E+03	2.40E-19	3.41E-29	5.31E-127	0.00E+00	0.00E+00
F_2	Ave	1.83E-07	1.25E-06	4.35E+01	1.76E-10	2.70E-16	3.85E-65	3.22E-156	1.90E-166
	Std	7.17E-08	1.24E-06	3.28E+01	1.93E-10	3.55E-16	9.49E-65	7.75E-156	0.00E+00
F_3	Ave	2.72E+02	1.42E+04	2.42E+05	8.77E+00	2.25E+01	8.99E-78	8.92E-233	2.66E-286
	Std	5.37E+02	7.56E+03	5.30E+04	1.12E+01	1.10E+02	4.18E-77	0.00E+00	0.00E+00
F_4	Ave	3.87E-02	5.72E+01	8.95E+01	2.15E-07	2.75E-01	1.66E-55	6.50E-143	2.93E-163
	Std	4.20E-02	1.39E+01	2.81E+00	7.98E-08	1.47E+00	5.93E-55	1.73E-142	0.00E+00
F_5	Ave	3.80E-101	1.40E-75	3.72E-54	1.13E-48	4.82E-133	6.92E-205	0.00E+00	0.00E+00
	Std	2.08E-100	4.31E-75	1.98E-53	6.16E-48	2.60E-132	0.00E+00	0.00E+00	0.00E+00
F_6	Ave	4.53E-03	9.39E-03	3.24E-04	1.45E-12	3.56E-03	0.00E+00	0.00E+00	0.00E+00
	Std	4.93E-03	1.77E-03	1.77E-03	7.81E-12	4.76E-03	0.00E+00	0.00E+00	0.00E+00
F_7	Ave	6.42E-03	5.10E-02	1.17E+02	1.79E-03	2.81E-03	3.00E-04	4.68E-04	2.71E-04
	Std	2.75E-03	1.94E-02	6.38E+01	1.05E-03	1.15E-03	2.97E-04	3.70E-04	2.33E-04
F_8	Ave	-3.85E+04	-1.33E+04	-6.79E+03	-2.47E+04	-2.62E+04	-2.34E+04	-2.53E+04	-2.36E+04
	Std	3.38E+03	1.09E+03	6.42E+02	9.62E+02	1.67E+03	1.36E+03	1.48E+03	1.29E+03
F_9	Ave	6.95E+00	9.89E+02	2.71E+02	0.00E+00	3.32E-02	0.00E+00	0.00E+00	0.00E+00
	Std	7.88E+00	1.08E+02	1.45E+02	0.00E+00	1.82E-01	0.00E+00	0.00E+00	0.00E+00
F_{10}	Ave	6.24E-08	1.97E-01	1.92E+01	4.57E-11	3.35E-14	4.44E-15	8.88E-16	8.88E-16
	Std	2.57E-08	7.52E-01	3.84E+00	2.87E-11	4.28E-15	0.00E+00	0.00E+00	0.00E+00
F_{11}	Ave	4.58E-03	1.16E-02	1.28E+02	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	9.75E-03	1.66E-02	7.83E+01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{12}	Ave	2.56E-03	1.57E+02	3.24E+01	3.27E-12	3.26E-18	3.50E-67	1.29E-157	4.42E-177
	Std	2.34E-03	2.19E+01	1.73E+01	2.77E-12	3.38E-18	9.47E-67	3.73E-157	0.00E+00
F_{13}	Ave	7.21E-05	1.23E-03	3.58E-02	4.22E-15	5.26E-05	6.87E-05	1.89E-15	1.87E-33
	Std	1.35E-04	2.50E-03	3.81E-02	1.30E-14	1.10E-04	1.23E-04	6.90E-15	1.01E-32
Avg.Rank		5.85	6.90	7.29	4.27	4.54	3.08	2.31	1.77
Rank		6	7	8	4	5	3	2	1

表5 固定维度实验结果对比

Table 5 Comparison of experimental results in fixed dimensions

F	Index	GWOCS	TSA	SCA	MPA	EO	SEO	GEO	MDSGEO
F_{14}	Ave	-1.025 302 0	-1.025 302 0	-1.031 569 2	-1.031 628 5	-1.031 628 5	-1.031 628 5	-1.031 628 5	-1.031 628 5
	Std	1.29E-02	1.29E-02	5.10E-05	4.48E-16	5.90E-16	5.98E-16	5.83E-16	5.76E-16
F_{15}	Ave	0.398	0.398	0.400	0.398	0.398	0.398	0.398	0.398
	Std	7.14E-05	7.14E-05	1.42E-03	2.56E-14	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{16}	Ave	7	7	3	3	3	3	3	3
	Std	1.54E+01	1.54E+01	8.26E-05	1.58E-15	1.69E-15	1.64E-15	1.32E-15	1.39E-15
Avg.Rank		5.33	7.17	6.83	3.33	3.75	3.75	2.92	2.92
Rank		4	6	5	2	3	3	1	1

均最优适应值, Std表示标准差, 最好的结果已加粗表示。

由表3~表5中数据可知: 在 $D=30$, $D=100$ 和固定维度的条件下, 由于MDSGEO同时具有GEO和SEO的优点, 不仅可以更好地平衡勘探与开发关系, 还增强种群进化信息加快算法收敛。因此, MDSGEO在 F_1 、 F_5 、 F_6 、 F_9 、 F_{11} 上全部收敛到理论最优解, 剩

余函数上收敛精度最高。在 $F_1 \sim F_6$ 、 F_9 、 F_{11} 、 F_{12} 上, GEO求解精度远远高于标准EO, 其余函数上也具有明显优势, 充分说明利用自适应优先引力策略丰富粒子进化信息增加粒子的信息交流, 帮助粒子逃离局部最优快速收敛到全局最优的方法效果显著。在 $F_1 \sim F_6$ 上, SEO相比EO求解精度呈指数级增长, 在 F_7 、 F_8 、 F_{10} 所得最优值精度更高, 甚至在 F_6 、 F_9 、

F_{11} 上求得理论最优值,这证明根据迭代进行阶段采用融合增减机制和柯西分布的正弦池策略能够实现种群粒子勘探与开发的动态平衡,将有助于粒子向着全局最优方向移动,提高寻得最优解的概率。此外,MDSGEO可以最大限度减弱维度增加带来的负面影响,在 $F_1 \sim F_{13}$ 上能取得更高质量的解。然而,TSA、GWOCS、SCA和MPA因其搜索模式中种群容易停滞更新出现早熟现象,陷入局部最优仍是它们急需解决的问题。尤其当 $D=100$ 时,SCA在 F_3 上与理论最优值相差 1×10^5 的精度。固定维度上,MDSGEO求得的平均适应度值与其他算法相似,但其标准差反映出MDSGEO在 $F_{14} \sim F_{15}$ 上比其他算法更稳定,GEO在 F_{16} 上最稳定,其次是MDSGEO。

为进一步评估MDSGEO的综合性能,采用非参数估计Friedman检验分析不同维度下各算法的寻优性能,实验结果如表3~表5所示。其中,Avg.Rank表示Friedman检验所得的秩均值,Rank表示排名。秩均值越小说明算法性能越好,排名越靠前。在 $D=30$, $D=100$ 及固定维度的条件下,MDSGEO的秩均值最小且排名第一,从统计学角度证明了MDSGEO的性能优于其他算法。同时,GEO和SEO的秩均值都比EO小,并且排名也在EO之前,再一次证明自适应优先引力策略和正弦池策略的有效性和鲁棒性。

新改进算法需要进行显著性分析,文中利用Wilcoxon秩和检验来判断MDSGEO在统计上的显

著性^[20]。MDSGEO与对比算法0.05的显著性水平下进行比较,假设MDSGEO为最佳算法,在MDSGEO vs GWOCS、MDSGEO vs TSA等之间进行两两成对比较,结果如表6所示。其中,p-value<0.05认为拒绝零假设,表明两对比算法之间存在显著差异,符号“+”“-”和“=”分别表示MDSGEO的性能优于、劣于和相当于对比算法,NA表示算法获得相同结果。从表6最后一行结果可以得出,MDSGEO与GWOCS、TSA、SCA、MPA、EO、SEO和GEO存在显著差异,并且MDSGEO的性能在多个函数上都要优于对比算法。

3.3 收敛性分析

图1(a)~(h)为在 $D=30$,迭代次数为500,评估次数为15 000的条件下得到的函数平均收敛曲线图。为了方便观察算法收敛情况,将纵坐标取以10为底的对数。由图1(a)~(h)能够看出,随着迭代进行,与标准EO相比,MDSGEO和GEO的收敛曲线下下降快且收敛到较高精度的解,这说明通过自适应优先引力策略加入当前最优粒子,丰富种群粒子的进化信息,能够有效帮助粒子增强逃离局部最优的能力,快速收敛到全局最优位置;同时,SEO的收敛曲线下下降速度仅次于GEO和MDSGEO,这证明通过正弦池策略中增减机制和柯西“长跳”特性,能推动粒子由勘探主导的寻优模式向开发模式过渡,动态平衡勘探与开发的关系,进一步帮助粒子向更好的方

表6 基准函数Wilcoxon秩和检验的实验结果对比

Table 6 Comparison of experimental results of Wilcoxon rank sum test of reference functions

F	GWOCS		TSA		SCA		MPA		EO		SEO		GEO	
	p-value	h	p-value	h	p-value	h	p-value	h	p-value	h	p-value	h	p-value	h
F_1	1.21E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	NA	=
F_2	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+
F_3	1.72E-12	+	1.72E-12	+	1.72E-12	+	1.72E-12	+	1.72E-12	+	1.72E-12	+	1.72E-12	+
F_4	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+
F_5	1.21E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	NA	=
F_6	4.79E-08	+	1.66E-11	+	2.16E-02	+	2.16E-02	+	5.56E-03	+	NA	=	NA	=
F_7	8.99E-11	+	3.02E-11	+	3.02E-11	+	4.62E-10	+	1.78E-10	+	4.06E-02	+	1.00E-03	+
F_8	2.57E-07	+	4.50E-11	+	3.02E-11	+	1.70E-02	+	1.17E-03	+	1.41E-01	-	3.50E-03	+
F_9	1.61E-11	+	1.21E-12	+	1.21E-12	+	NA	=	NA	=	NA	=	NA	=
F_{10}	1.10E-12	+	1.21E-12	+	1.21E-12	+	1.21E-12	+	8.64E-14	+	NA	=	NA	=
F_{11}	1.46E-04	+	8.87E-07	+	1.21E-12	+	NA	=	NA	=	NA	=	NA	=
F_{12}	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+
F_{13}	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+	3.02E-11	+
F_{14}	1.01E-11	+	1.01E-11	+	1.01E-11	+	1.87E-08	+	7.84E-01	-	8.41E-02	-	2.68E-01	-
F_{15}	1.21E-12	+	1.21E-12	+	1.21E-12	+	6.59E-04	+	NA	=	NA	=	NA	=
F_{16}	2.72E-11	+	2.72E-11	+	2.72E-11	+	2.90E-02	+	1.85E-02	+	1.73E-01	-	4.76E-02	+
+ / = / -	16/0/0		16/0/0		16/0/0		14/2/0		12/3/1		8/5/3		8/7/1	

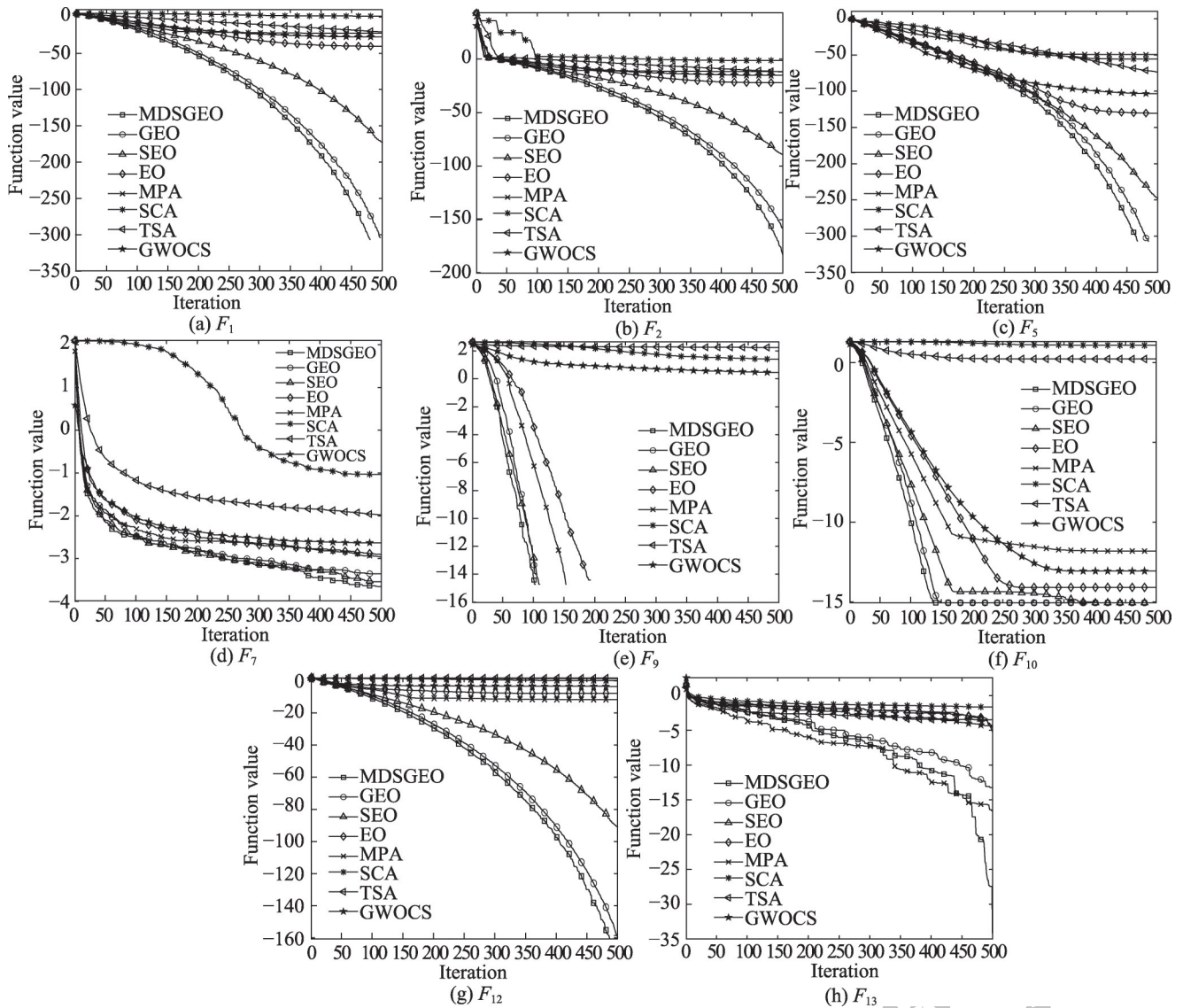


图1 平均收敛曲线

Fig.1 Average convergence curve

向收敛。另外, MDSGEO 进入迭代后期仍能持续寻优, 未出现停滞现象, 收敛速度最快且寻到的解精度远高于其他算法, 这离不开正弦池策略动态调节勘探与开发关系以及自适应优先引力策略丰富种群进化信息的帮助。MDSGEO 和 GEO 在 F_1 、 F_5 、 F_9 上都寻到最优解, 但 MDSGEO 用到评估次数更少, 又一次说明融合正弦池策略和自适应优先引力策略的 MDSGEO 可以寻到最好的解。然而, 标准 EO 和其他对比算法在迭代前期和后期收敛曲线平缓甚至出现不同程度的停滞状况, 进而导致出现早熟现象和求解精度低等问题。

结合表 3~表 5 和图 1 可以得出, 无论是单峰可分、单峰不可分、多峰可分、多峰不可分、高维及固定

维度, MDSGEO 在复杂函数上具备强劲的求解能力以及更快的收敛速度, 验证了所提策略的有效性和鲁棒性。

3.4 与 9 种改进算法对比

为了展现 MDSGEO 的竞争性, 将其与改进的差分算法 JADE^[21]、jDE^[22]、SaDE^[23]、改进的人工蜂群算法 MPGABC^[24]、GABC^[25]、MABC^[26]、ABVCVSS^[27]、DFSABC_elite^[28]、改进的樽海鞘算法 MSNSSA^[29] 进行比较。实验选择 3.1 节中的 13 个典型测试函数, 函数维度设置 $D=30$, 各算法参数设置参照原文献并让每个算法独立运行 50 次, 所得实验结果如表 7 所示。由于原文献中没有对比函数的数据, 故用“—”表示。表 7 中, JADE (adaptive differential evolution

with optional external archive)、jDE (self-adapting control parameters in differential evolution) 和 SaDE (self-adaptive differential evolution algorithm) 的数据来源于文献[21], MPGABC (modified gbest-guided artificial bee colony) 的数据源自文献[24], GABC (Gbest-guided artificial bee colony)、MABC (modified artificial bee colony)、ABCVSS (artificial bee colony algorithm with variable search strategy) 和 DFSABC_elite (novel artificial bee colony algorithm with depth-first search framework and elite-guided search equation) 的数据来自文献[28], MSNNSA (multi-subpopulation based symbiosis and non-uniform Gaussian mutation salp swarm algorithm) 的数据源于文献[29]。由表数据可知, MDSGEO 在 F_1 、 F_5 、 F_6 、 F_9 、 F_{11} 共 5 个函数上寻得最优解且鲁棒性最好, 相比其他 9 种改进算法具有显著优势。在 F_7 、 F_{10} 所得解的质量及鲁棒性都优于其他改进算法。在剩余函数上, MDSGEO 的总

体寻优远高于其他改进算法。这表明 MDSGEO 不仅具有强大的寻优能力, 也具备出色的竞争力。

3.5 与 4 种改进算法在 CEC2017 函数集上对比

为了进一步验证 MDSGEO 的寻优能力, 采用更复杂的 CEC2017 函数集来测试, 其中 No.2 的函数在 CEC2017 函数集上被移除, 即文中所用 CEC2017 函数集仅有 29 个。同时, 将 MDSGEO 与标准粒子群 PSO (particle swarm optimization)^[7]、改进粒子群 FFPSO (hybrid firefly and particle swarm optimization)^[30]、HPSOFF (hybrid particle swarm optimization algorithm and firefly algorithm)^[31] 和 HFPSO (hybrid firefly and particle swarm optimization)^[32] 进行比较。为保证公平对比, 对比数据和实验参数设置与文献[32]一致。函数维度 $D=10$, 种群粒子数量与函数维度一致, 最大迭代次数为 500, 独立运行 20 次, 实验最终结果记录在表 8 中。

表 7 与 9 种改进算法的实验结果对比

Table 7 Comparison of experimental results with 9 improved algorithms

F	Index	JADE	SaDE	jDE	MPGABC	GABC	MABC	ABCVSS	DFSABC_elite	MSNNSA	MDSGEO
F_1	Ave	1.8E-60	4.5E-20	2.5E-28	5.41E-55	1.07E-30	2.36E-40	2.40E-35	4.14E-82	7.35E-36	0.00E+00
	Std	8.4E-60	6.9E-20	3.5E-28	1.95E-54	6.09E-31	1.85E-40	1.77E-43	8.76E-82	1.04E-35	0.00E+00
F_2	Ave	1.8E-25	1.9E-14	1.5E-23	2.60E-29	8.30E-17	1.50E-21	7.03E-19	2.06E-42	—	8.12E-179
	Std	8.8E-25	1.1E-14	1.0E-23	8.89E-29	3.10E-17	6.64E-22	2.18E-18	2.08E-42	—	0.00E+00
F_3	Ave	5.7E-61	9.0E-37	5.2E-14	—	—	—	—	—	9.69E-32	8.76E-302
	Std	2.7E-60	5.4E-36	1.1E-13	—	—	—	—	—	4.52E-32	0.00E+00
F_4	Ave	8.2E-24	7.4E-11	1.4E-15	9.87E+00	2.47E-01	1.93E+00	2.56E-01	5.08E-07	1.47E-21	1.02E-174
	Std	4.0E-23	1.8E-10	1.0E-15	3.32E+00	5.89E-02	1.32E+00	9.19E-02	3.69E-07	4.75E-21	0.00E+00
F_5	Ave	—	—	—	—	—	—	—	—	—	0.00E+00
	Std	—	—	—	—	—	—	—	—	—	0.00E+00
F_6	Ave	—	—	—	—	—	—	—	—	0.00E+00	0.00E+00
	Std	—	—	—	—	—	—	—	—	0.00E+00	0.00E+00
F_7	Ave	6.4E-04	4.8E-03	3.3E-03	2.46E-02	3.08E-02	2.77E-02	2.57E-02	1.20E-02	—	3.05E-04
	Std	2.5E-04	1.2E-03	8.5E-04	6.43E-03	6.44E-03	6.36E-03	5.22E-03	3.80E-03	—	2.96E-04
F_8	Ave	—	—	—	—	—	—	—	—	—	-8.69E+03
	Std	—	—	—	—	—	—	—	—	—	7.28E+02
F_9	Ave	1.0E-04	1.2E-03	1.5E-04	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	6.0E-05	6.5E-04	2.0E-04	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{10}	Ave	8.2E-10	2.7E-03	3.5E-04	1.42E-14	3.71E-14	7.07E-15	6.50E-15	3.80E-15	8.88E-16	8.88E-16
	Std	6.9E-10	5.1E-04	1.0E-04	3.59E-15	5.78E-15	2.36E-15	2.27E-15	1.69E-15	0.00E+00	0.00E+00
F_{11}	Ave	9.9E-08	7.8E-04	1.9E-05	2.11E-09	1.48E-11	0.00E+00	3.45E-11	0.00E+00	0.00E+00	0.00E+00
	Std	6.0E-07	1.2E-03	5.8E-05	1.06E-08	7.39E-11	0.00E+00	1.73E-10	0.00E+00	0.00E+00	0.00E+00
F_{12}	Ave	—	—	—	1.83E-08	5.22E-07	1.43E-21	6.26E-18	3.10E-40	—	5.91E-182
	Std	—	—	—	4.32E-08	1.12E-06	1.75E-21	2.91E-17	1.03E-39	—	0.00E+00
F_{13}	Ave	—	—	—	—	—	—	—	—	—	2.17E-36
	Std	—	—	—	—	—	—	—	—	—	8.37E-36

表8 CEC2017函数集上实验结果对比

Table 8 Comparison of experimental results on CEC2017

No.	Index	PSO	FFPSO	HPSOFF	HFPSO	EO	MDSGEO
1	Ave	3.216 5E+08	2.281 2E+10	1.734 5E+08	2.454 1E+08	1.549 0E+04	3.282 4E+04
	Std	2.386 1E+08	7.510 1E+09	2.802 5E+08	9.806 0E+08	3.897 6E+04	5.850 9E+04
3	Ave	1.790 8E+04	8.901 0E+07	2.276 7E+04	1.174 4E+04	1.178 1E+03	1.516 6E+03
	Std	4.971 1E+03	1.836 7E+08	1.066 8E+04	5.956 8E+03	1.030 6E+03	8.841 0E+02
4	Ave	5.478 7E+02	3.317 3E+03	4.589 5E+02	4.410 1E+02	8.078 4E+02	8.046 9E+02
	Std	3.289 9E+02	1.666 1E+03	4.384 4E+01	4.545 2E+01	5.710 9E+02	5.710 1E+02
5	Ave	5.724 8E+02	6.796 4E+02	5.666 1E+02	5.529 2E+02	8.831 7E+02	8.831 0E+02
	Std	1.676 3E+01	2.475 3E+01	1.722 3E+01	1.840 2E+01	5.156 1E+02	5.156 4E+02
6	Ave	6.303 2E+02	6.844 7E+02	6.249 4E+02	6.219 4E+02	9.340 9E+02	9.337 6E+02
	Std	1.513 0E+01	1.710 8E+01	9.643 5E+00	1.352 1E+01	4.789 2E+02	4.791 6E+02
7	Ave	7.814 6E+02	1.014 6E+03	7.810 7E+02	7.634 3E+02	1.021 5E+03	1.028 0E+03
	Std	1.190 2E+01	6.146 1E+01	1.513 8E+01	1.726 3E+01	4.161 6E+02	4.115 1E+02
8	Ave	8.605 5E+02	9.314 0E+02	8.522 1E+02	8.395 2E+02	1.079 9E+03	1.082 0E+03
	Std	9.026 6E+00	2.001 2E+01	1.354 4E+01	1.443 8E+01	3.740 9E+02	3.725 9E+02
9	Ave	1.109 9E+03	3.678 4E+03	1.223 8E+03	1.177 5E+03	1.136 5E+03	1.136 1E+03
	Std	2.226 9E+02	8.931 4E+02	2.750 6E+02	3.066 7E+02	3.334 5E+02	3.252 3E+02
10	Ave	3.008 0E+03	3.861 4E+03	2.627 0E+03	2.267 0E+03	1.735 9E+03	1.702 4E+03
	Std	2.992 3E+02	2.873 1E+02	3.771 5E+02	3.793 3E+02	2.270 8E+02	2.681 8E+02
11	Ave	1.675 9E+03	6.422 1E+04	1.262 5E+03	1.195 9E+03	1.294 3E+03	1.288 6E+03
	Std	9.914 0E+02	1.694 8E+05	1.269 6E+02	5.238 0E+01	2.333 0E+02	2.322 2E+02
12	Ave	1.861 1E+07	3.034 3E+09	2.824 3E+06	2.899 8E+06	3.033 0E+05	2.824 5E+05
	Std	8.876 6E+06	1.525 3E+09	3.077 2E+06	4.130 7E+06	7.605 2E+05	7.549 9E+05
13	Ave	4.952 8E+05	6.569 3E+08	1.457 3E+04	1.183 5E+04	7.370 3E+03	6.797 9E+03
	Std	3.949 2E+05	6.423 9E+08	7.420 6E+03	7.684 5E+03	8.411 8E+03	7.804 1E+03
14	Ave	7.642 8E+03	8.500 2E+05	5.041 4E+03	4.647 8E+03	3.404 4E+03	2.890 0E+03
	Std	7.014 6E+03	1.089 2E+06	3.443 9E+03	4.177 6E+03	2.995 4E+03	2.284 1E+03
15	Ave	4.826 6E+04	5.998 0E+07	1.360 9E+04	2.284 5E+04	3.692 6E+03	2.777 2E+03
	Std	4.663 1E+04	9.855 9E+07	1.105 0E+04	2.365 0E+04	3.953 5E+03	1.510 6E+03
16	Ave	2.120 1E+03	2.881 6E+03	2.051 6E+03	1.962 9E+03	1.718 4E+03	1.718 0E+03
	Std	1.960 9E+02	3.181 4E+02	1.526 9E+02	1.587 7E+02	1.611 2E+02	1.449 8E+02
17	Ave	1.865 0E+03	2.478 0E+03	1.821 7E+03	1.824 8E+03	1.716 1E+03	1.702 7E+03
	Std	5.921 5E+01	2.798 3E+02	4.683 6E+01	8.399 7E+01	9.034 9E+01	7.558 4E+01
18	Ave	7.053 3E+05	2.029 3E+09	8.167 5E+04	2.973 2E+04	1.386 6E+04	1.194 2E+04
	Std	8.230 7E+05	1.887 5E+09	1.973 0E+05	1.794 1E+04	1.416 1E+04	1.158 9E+04
19	Ave	3.586 8E+04	2.434 5E+07	1.803 0E+04	3.517 7E+04	6.997 5E+03	3.931 4E+03
	Std	2.599 9E+04	4.184 2E+07	9.094 0E+03	3.830 0E+04	8.603 9E+03	4.568 4E+03
20	Ave	2.266 8E+03	2.568 6E+03	2.183 4E+03	2.202 3E+03	1.928 3E+03	1.910 3E+03
	Std	1.162 8E+02	1.614 5E+02	9.349 3E+01	1.075 3E+02	2.463 0E+02	2.272 5E+02
21	Ave	2.355 3E+03	2.463 8E+03	2.348 8E+03	2.337 7E+03	2.080 5E+03	2.053 2E+03
	Std	3.700 9E+01	3.519 4E+01	4.662 2E+01	4.775 0E+01	3.456 8E+02	3.292 0E+02
22	Ave	2.668 4E+03	4.366 3E+03	2.419 6E+03	2.597 5E+03	2.139 3E+03	2.068 2E+03
	Std	7.765 4E+02	6.463 6E+02	2.026 0E+02	5.875 5E+02	4.434 7E+02	3.367 5E+02
23	Ave	2.690 6E+03	3.029 3E+03	2.678 3E+03	2.675 7E+03	2.281 2E+03	2.280 6E+03
	Std	3.423 4E+01	2.091 7E+02	2.337 8E+01	2.866 8E+01	4.899 7E+02	4.895 1E+02
24	Ave	2.797 2E+03	3.186 5E+03	2.733 7E+03	2.712 8E+03	2.359 2E+03	2.369 2E+03
	Std	9.484 2E+01	1.520 8E+02	1.116 8E+02	1.467 2E+02	5.480 0E+02	5.532 8E+02
25	Ave	2.964 8E+03	4.400 5E+03	2.951 5E+03	2.958 1E+03	2.488 8E+03	2.486 3E+03
	Std	1.575 5E+01	5.427 4E+02	1.842 5E+01	5.019 3E+01	6.395 2E+02	6.377 0E+02
26	Ave	3.454 5E+03	5.079 7E+03	3.447 1E+03	3.214 7E+03	2.546 6E+03	2.496 0E+03
	Std	6.325 6E+02	4.202 9E+02	5.358 5E+02	3.422 7E+02	7.240 6E+02	6.758 9E+02
27	Ave	3.152 9E+03	3.750 7E+03	3.135 6E+03	3.153 8E+03	2.601 6E+03	2.598 4E+03
	Std	3.979 3E+01	2.673 2E+02	3.356 4E+01	3.937 0E+01	7.205 0E+02	7.180 8E+02
28	Ave	3.345 0E+03	4.360 3E+03	3.406 4E+03	3.442 2E+03	2.752 5E+03	2.756 4E+03
	Std	1.049 8E+02	2.803 7E+02	9.458 5E+01	1.079 4E+02	8.352 7E+02	8.360 3E+02
29	Ave	3.429 9E+03	4.105 2E+03	3.372 1E+03	3.386 4E+03	2.675 2E+03	2.687 7E+03
	Std	8.876 9E+01	3.266 7E+02	1.018 3E+02	9.397 6E+01	7.735 5E+02	7.832 8E+02
30	Ave	4.145 6E+06	3.036 6E+08	3.600 5E+06	4.272 3E+06	3.035 2E+05	2.479 4E+05
	Std	4.266 6E+06	2.469 0E+08	2.995 0E+06	3.754 9E+06	5.425 2E+05	4.610 6E+05
Avg.Rank		3.76	5.26	2.79	3.14	3.31	2.74
Rank		5	6	2	3	4	1

由表8可知,与PSO相比,MDSGEO在24个函数上的收敛精度更高,尤其在No.1函数上远高于PSO;MDSGEO在所有函数上获得的收敛精度都优于FFPSO和HPSOFF;HFPSO仅在6个函数的收敛精度高于MDSGEO;与EO相比,除了No.1函数,MDSGEO的整体收敛精度更高。MDSGEO能够在众多先进算法中取得更高的优化精度,正弦池策略和自适应优先引力策略扮演重要角色,前者通过动态平衡算法的勘探与开发能力,提高算法的优化精度;后者丰富种群进化信息,加快算法收敛。另外,MDSGEO在所测29个函数中,不仅在21个函数上平均值优势突出,而且Friedman检验的结果也表明MDSGEO秩均值最小且排名第一,再一次验证MDSGEO性能更优。

4 工程优化问题

4.1 约束优化问题

约束优化条件与工程问题结合属于一类常见的数学优化问题,该类问题通常很难求解,但因其在各科学领域被广泛应用,所以研究这类问题具有重要的研究价值。约束优化问题定义如下:

$$\begin{cases} \min f(x) \\ \text{s.t. } g_i(x) \leq 0, i = 1, 2, \dots, m \\ h_j(x) = 0, j = 1, 2, \dots, q \\ x \in X \end{cases} \quad (25)$$

其中, f 为目标函数, $h_j(x)=0$ 是等式约束条件, $g_i(x) \leq 0$ 为不等式约束条件, X 是实数 \mathbf{R}^n 的子集且用来约束优化问题的搜索空间,若所求解满足约束条件,则称为可行解;反之,称为不可行解。约束优化问题的难点在于搜索空间可行域的分布受约束影响,而优化过程中不仅需要考虑约束对这些分布影响,还需要考虑如何平衡约束和优化。

许多学者对约束优化问题都有研究,主要分为传统方法和元启发式算法两大类。传统方法包含文

献[33]提到移动渐进线法(method of moving asymptotes, MMA)、广义凸近似(generalized convex approximation, GCA),来提高算法的收敛速度,但这类算法结构复杂不易实现;元启发式算法有布谷鸟算法(cuckoo search algorithm, CS)^[34]、矿山爆炸算法(mine blast algorithm, MBA)^[35]、具有动态随机选择的差分进化(differential evolution with dynamic stochastic selection, DEDS)^[36]、人工原子算法(artificial atom algorithm, AAA)^[37]等,这类算法虽结构简单,但算法的收敛精度不高且收敛速度慢。

为了解决上述问题,将MDSGEO用来求解优化问题,证明其有效性和可行性。选用三杆桁架和悬臂梁设计带约束的工程问题进行分析。

4.2 三杆桁架设计

三杆桁架,以使其重量最小化^[34-35]。目标函数非常简单,但是这个问题受到了很大的限制。结构设计问题通常具有很多约束。这里的约束是应力、挠曲和屈曲约束。

桁架的整体结构如图2所示,图中 $L=100\text{ cm}$ 、 $P=2\text{ kN/cm}^2$ 、 $\sigma=2\text{ kN/cm}^2$ 。选择进行比较的算法有CS、MBA、Ray and Sain^[38]、DEDS和AAA,比较结果列于表9。

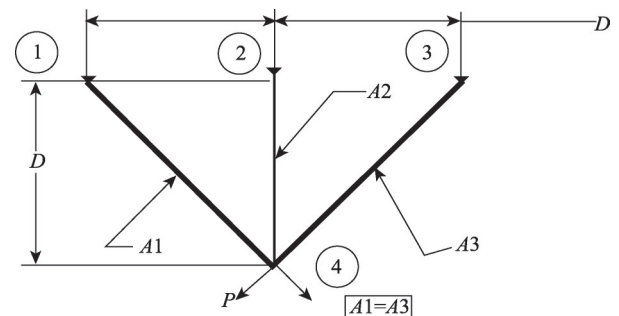


图2 三杆桁架模型

Fig.2 Three bar truss model

表9 三杆桁架实验结果对比

Table 9 Comparison of experimental results of three bar truss

Algorithms	Optimal values for variables		Optimum	Max.eval.
	x_1	x_2		
Ray and Sain	0.795 000 000	0.395 000 000	264.300 000 00	N/A
m-EO	0.788 345 650	0.409 182 560	263.896 077 83	15 000
DEDS	0.788 675 130	0.408 248 280	263.895 843 40	15 000
MBA	0.788 565 000	0.408 559 700	263.895 852 20	20 000
CS	0.788 670 000	0.409 020 000	263.971 600 00	15 000
AAA	0.788 735 400	0.408 078 000	263.895 880 00	N/A
EO	0.790 719 516	0.402 496 410	263.895 845 50	15 000
MDSGEO	0.789 060 299	0.407 159 975	263.895 843 40	15 000

表 10 悬梁臂实验结果对比

Table 10 Comparison of experimental results of cantilever

Algorithms	Optimal values for variables					Optimum	Max.eval.
	x_1	x_2	x_3	x_4	x_5		
SOS	6.018 780 0	5.303 440 000	4.495 870 000	3.498 960 000	2.155 640 000	1.339 960 000	15 000
CS	6.008 900 0	5.304 900 000	4.502 300 000	3.507 700 000	2.150 400 000	1.339 990 000	2 500
MMA	6.010 000 0	5.300 000 000	4.490 000 000	3.490 000 000	2.150 000 000	1.340 000 000	N/A
GCA_I	6.010 000 0	5.304 000 000	4.490 000 000	3.498 000 000	2.150 000 000	1.340 000 000	N/A
EO	6.030 933 5	5.321 276 399	4.479 971 559	3.489 046 781	2.152 741 014	1.339 961 006	15 000
MDSGEO	6.065 803 5	5.296 354 328	4.477 009 924	3.519 971 759	2.116 942 436	1.339 958 131	15 000

表 9 中 Max.eval. 表示最大评估次数, N/A 表示原文献未说明评估次数。由表 9 可知, MDSGEO 算法提供了非常有竞争力的结果, 其获得最佳解决方案。MDSGEO 不仅能够有效解决实际的约束问题, 而且相同评估次数下 MDSGEO 能够找到最优设计权重 263.895 843, 这比其他算法要少。

$$\begin{cases} \min f(x) = (2\sqrt{2}x_1 + x_2) \times L \\ g_1 = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0 \\ g_2 = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0 \\ g_3 = \frac{1}{x_1 + \sqrt{2}x_2} P - \sigma \leq 0 \\ 0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1 \end{cases} \quad (26)$$

4.3 悬臂梁设计

悬臂梁包括 5 个横截面为正方形的空心元件。如图 3 所示, 梁的自由端(节点 6)上也施加了垂直载荷, 梁的右侧(节点 1)受到了刚性支撑。其目的是使梁的重量最小。在最终的优化设计中, 还存在一个不应违反的纵向位移约束。问题表述如下:

将 MDSGEO 和 EO 分别对悬臂梁设计求解最优值, 并与 m-EO(modified equi optimizer)^[11]、SOS(symbiotic organisms search)^[34]、CS、MMA、GCA_I^[33] 进行比较。表 10 所用符号表示与表 9 一致。表 10 的优化结果反映出 MDSGEO 优于其他算法。这显示了 MDSGEO 在逼近此问题的全局最优值方面拥有出色的性能。

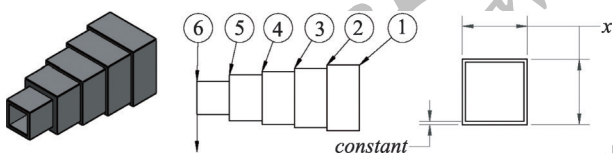


图 3 悬梁臂模型

Fig.3 Cantilever model

$$\begin{cases} \min f(x) = 0.622 4(x_1 + x_2 + x_3 + x_4 + x_5) \\ g(x) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} \leq 1 \\ 0.01 \leq x_1, x_2, x_3, x_4, x_5 \leq 100 \end{cases} \quad (27)$$

5 结束语

为提高标准平衡优化器算法全局寻优能力, 本文提出 MDSGEO 算法。在种群粒子初始化后, 利用正弦池策略自适应平衡勘探与开发, 提高算法寻优精度。此外, 通过自适应优先引力策略加速粒子收敛到全局最优。同时将平衡优化器算法应用于多类型基准函数及 CEC2017 函数集进行寻优, 使用平均值和标准差等指标进行评估以及非参数统计 Friedman 检验、Wilcoxon 秩和检验进行统计验证算法的有效性和鲁棒性。最后, 利用两个带约束的工程优化问题测试算法的全局寻优能力。研究结果表明, 改进平衡优化器算法不仅能够自适应平衡勘探与开发, 而且寻优能力更强。未来研究方向准备将 MDSGEO 应用于多目标的工程优化及高维的特征选择领域。

参考文献:

- [1] ANSEL D T, SEVINÇ E, KUCUKYILMAZ T, et al. A survey on new generation metaheuristic algorithms[J]. Computers & Industrial Engineering, 2019, 137: 106040.
- [2] NEUMANN F, WITT C. Combinatorial optimization and computational complexity[M]//Bioinspired Computation in Combinatorial Optimization. Berlin, Heidelberg: Springer, 2010.
- [3] 刘二辉, 姚锡凡, 刘敏, 等. 基于改进灰狼优化算法的自动导引小车路径规划及其实现原型平台[J]. 计算机集成制造系统, 2018, 24(11): 2779-2791.
- LIU E H, YAO X F, LIU M, et al. AGV path planning based on improved grey wolf optimization algorithm and

- its implementation prototype platform[J]. *Computer Integrated Manufacturing Systems*, 2018, 24(11): 2779-2791.
- [4] 史春天, 曾艳阳, 侯守明. 群体智能算法在图像分割中的应用综述[J]. *计算机工程与应用*, 2021, 57(8): 36-47.
SHI C T, ZENG Y Y, HOU S M. Summary of application of swarm intelligence algorithms in image segmentation[J]. *Computer Engineering and Applications*, 2021, 57(8): 36-47.
- [5] 王贞, 李旭飞. 精英学习人工蜂群算法的PID控制器参数优化[J]. *数学的实践与认识*, 2020, 50(16): 177-186.
WANG Z, LI X F. An elite learning artificial bee colony algorithm for parameter optimization of PID controller[J]. *Mathematics in Practice and Theory*, 2020, 50(16): 177-186.
- [6] FARAMARZI A, HEIDARINEJAD M, BRENT STEPHENS B E, et al. Equilibrium optimizer: a novel optimization algorithm[J]. *Knowledge-Based Systems*, 2020, 191: 105190.
- [7] KENNEDY J, EBERHART R. Particle swarm optimization [C]//*Proceedings of the 1995 International Conference on Neural Networks*, Perth, Nov 27-Dec 1, 1995. Piscataway: IEEE, 1995: 1942-1948.
- [8] HOLLAND J H. *Adaptation in natural and artificial systems* [R]. Ann Arbor: University of Michigan, 1975.
- [9] DAS S S P N. Differential evolution: a survey of the state-of-the-art[J]. *IEEE Transactions on Evolutionary Computation*, 2011, 15(1): 4-31.
- [10] 张哲辰, 刘三阳. 基于拓扑改进与交叉策略的萤火虫算法[J]. *计算机工程与应用*, 2019, 55(7): 1-8.
ZHANG Z C, LIU S Y. Firefly algorithm based on topology improvement and crossover strategy[J]. *Computer Engineering and Applications*, 2019, 55(7): 1-8.
- [11] FAN Q, HUANG H, YANG K, et al. A modified equilibrium optimizer using opposition-based learning and novel update rules[J]. *Expert Systems with Applications*, 2021, 170: 114575.
- [12] SAYED G I, KHORIBA G, HAGGAG M H. A novel chaotic equilibrium optimizer algorithm with S-shaped and V-shaped transfer functions for feature selection[J]. *Journal of Ambient Intelligence and Humanized Computing*, 2022, 13(6): 3137-3162.
- [13] DINKAR S K, DEEP K, MIRJALILI S, et al. Opposition-based Laplacian equilibrium optimizer with application in image segmentation using multilevel thresholding[J]. *Expert Systems with Applications*, 2021, 174: 114766.
- [14] AHMED S, GHOSH K K, MIRJALILI S, et al. AIEOU: automata-based improved equilibrium optimizer with U-shaped transfer function for feature selection[J]. *Knowledge-Based Systems*, 2021, 228: 107283.
- [15] KARDANI N, BARDHAN A, GUPTA S, et al. Predicting permeability of tight carbonates using a hybrid machine learning approach of modified equilibrium optimizer and extreme learning machine[J]. *Acta Geotechnica*, 2022, 17: 1239-1255.
- [16] SHANKAR N, SARAVANAKUMAR N, KUMAR C, et al. Opposition-based equilibrium optimizer algorithm for identification of equivalent circuit parameters of various photovoltaic models[J]. *Journal of Computational Electronics*, 2021, 20: 1560-1587.
- [17] MIRJALILI S. SCA: a sine cosine algorithm for solving optimization problems[J]. *Knowledge-Based Systems*, 2016, 96: 120-133.
- [18] FARAMARZI A, HEIDARINEJAD M, MIRJALILI S, et al. Marine predators algorithm: a nature-inspired metaheuristic [J]. *Expert Systems with Applications*, 2020, 152: 113377.
- [19] KAUR S, AWASTHI L K, SANGAL A L, et al. Tunicate swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization[J]. *Engineering Applications of Artificial Intelligence*, 2020, 90: 103541.
- [20] DERRAC J, GARCÍA S, MOLINA D, et al. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms[J]. *Swarm & Evolutionary Computation*, 2011, 1(1): 3-18.
- [21] ZHANG J Q, SANDERSON A C. JADE: adaptive differential evolution with optional external archive[J]. *IEEE Transactions on Evolutionary Computation*, 2009, 13(5): 945-958.
- [22] BREST J, GREINER S, BOSKOVIC B, et al. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems[J]. *IEEE Transactions on Evolutionary Computation*, 2006, 10(6): 646-657.
- [23] QIN A K, SUGANTHAN P N. Self-adaptive differential evolution algorithm for numerical optimization[C]//*Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, Edinburgh, Sep 2-4, 2005. Piscataway: IEEE, 2005: 1785-1791.
- [24] CUI L Z, ZHANG K, LI G H, et al. Modified Gbest-guided artificial bee colony algorithm with new probability model [J]. *Soft Computing*, 2018, 22(7): 2217-2243.
- [25] ZHU G P, SAM K. Gbest-guided artificial bee colony algorithm for numerical function optimization[J]. *Applied Mathematics and Computation*, 2010, 217(7): 3166-3173.
- [26] GAO W F, LIU S Y. A modified artificial bee colony algorithm[J]. *Computers and Operations Research*, 2012, 39(3): 687-697.
- [27] KIRAN M S, HAKLI H, GÜNDÜZ M, et al. Artificial bee colony algorithm with variable search strategy for continuous optimization[J]. *Information Sciences*, 2015, 300: 140-157.
- [28] CUI L Z, LI G H, LIN Q Z, et al. A novel artificial bee

- colony algorithm with depth-first search framework and elite-guided search equation[J]. *Information Sciences*, 2016, 367/368: 1012-1044.
- [29] 陈忠云, 张达敏, 辛梓芸. 多子群的共生非均匀高斯变异樽海鞘群算法[J]. *自动化学报*, 2022, 48(5): 1307-1317.
CHEN Z Y, ZHANG D M, XIN Z Y. Multi-subpopulation based symbiosis and non-uniform Gaussian mutation salp swarm algorithm[J]. *Acta Automatica Sinica*, 2022, 48(5): 1307-1317.
- [30] PADMAVATHI K, RAMAKRISHNA K S. Hybrid firefly and particle swarm optimization algorithm for the detection of bundle branch block[J]. *International Journal of the Cardiovascular Academy*, 2016, 2(1): 44-48.
- [31] ARUNACHALAM S, AGNESBHOMILA T, BABU M R. Hybrid particle swarm optimization algorithm and firefly algorithm based combined economic and emission dispatch including valve point effect[C]//LNCS 8947: Proceedings of the 5th International Conference on Swarm, Evolutionary, and Memetic Computing, Bhubaneswar, Dec 18-20, 2014. Cham: Springer, 2014: 647-660.
- [32] AYDILEK I B. A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems [J]. *Applied Soft Computing*, 2018, 66: 232-249.
- [33] CHICKERMANE H, GEA H C. Structural optimization using a new local approximation method[J]. *International Journal for Numerical Methods in Engineering*, 1996, 39(5): 829-846.
- [34] GANDOMI A H, YANG X S, ALAVI A H. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems[J]. *Engineering with Computers*, 2013, 29(1): 17-35.
- [35] SADOLLAH A, BAHREININEJAD A, ESKANDAR H, et al. Mine blast algorithm: a new population based algorithm for solving constrained engineering optimization problems [J]. *Applied Soft Computing Journal*, 2013, 13(5): 2592-2612.
- [36] ZHANG M, LUO W J, WANG X F. Differential evolution with dynamic stochastic selection for constrained optimization [J]. *Information Sciences*, 2008, 178(15): 3043-3074.
- [37] YILDIRIM A E, KARCI A. Application of three bar truss problem among engineering design optimization problems using artificial atom algorithm[C]//Proceedings of the 2018 International Conference on Artificial Intelligence and Data Processing, Malatya, Sep 28-30, 2018. Piscataway: IEEE, 2018: 1-5.
- [38] RAY T, SAINI P. Engineering design optimization using a swarm with an intelligent information sharing among individuals[J]. *Engineering Optimization*, 2001, 33(6): 735-748.



李守玉(1996—),男,贵州安顺人,硕士研究生,主要研究方向为进化计算、深度学习。

LI Shouyu, born in 1996, M.S. candidate. His research interests include evolutionary computation and deep learning.



何庆(1982—),男,贵州黔南州人,博士,副教授,主要研究方向为大数据应用、进化计算。

HE Qing, born in 1982, Ph.D., associate professor. His research interests include big data application and evolutionary computation.



陈俊(1996—),男,贵州毕节人,硕士研究生,主要研究方向为进化计算、深度学习。

CHEN Jun, born in 1996, M.S. candidate. His research interests include evolutionary computation and deep learning.