

基于差分进化改进的蝗虫优化算法*

宋长新¹, 马 克²

(1. 上海城建职业学院 机电工程与信息学院, 上海 201415;

2. 青海师范大学 档案馆, 青海 西宁 810008)

摘 要: 蝗虫算法(Grasshopper optimization algorithm, GOA), 是一种以蝗虫群体行为为理论基础的新群智能算法, 其性能优越, 但仍然存在求解精度不高, 收敛速度慢, 陷入局部最优的不足。为了提高蝗虫算法的求解精度和收敛速度, 减低其搜索盲目性, 提出了一种基于差分进化改进的蝗虫优化算法(DE-GOA)。在蝗虫算法的迭代后期, 加入差分进化策略增加种群多样性, 提高整个算法的全局寻优能力。最后将改进的差分进化蝗虫算法与基本蝗虫算法 GOA, PSO 算法, GA 算法, FPA 算法, FA 算法, BA 算法在 11 个标准测试函数进行寻优实验对比, 实验结果表明 DE-GOA 算法相比其他算法具有明显的优势。

关键词: 蝗虫算法; 差分进化; 迭代寻优; 函数优化; 种群多样性

中图分类号: TP18 文献标识码: A 文章编号: 1003-7241(2022)03-0012-05

Improved Grasshopper Optimization Algorithm Based on Differential Evolution

SONG Chang-xin¹, MA Ke²

(1. School of Electromechanical Engineering & Information, Shanghai Urban Construction Vocational College, Shanghai 201415 China;

2. Archive, Qinghai Normal University, Xining 810008 China)

Abstract: Grasshopper optimization algorithm is a new Swarm Intelligence Algorithms based on the theory of swarm behavior of Grasshopper. Its performance is superior, but there are still shortcomings such as low accuracy, slow convergence and falling into local optimum. In order to improve the accuracy and convergence speed of Grasshopper algorithm and reduce its search blindness, an improved Grasshopper optimization algorithm based on differential evolution (DE-GOA) is proposed. In the later iteration stage of Grasshopper algorithm, differential evolution strategy is added to increase the diversity of population and improve the global optimization ability of the whole algorithm. Finally, the improved DE-GOA algorithm is compared with the basic GOA algorithm, GA algorithm, PSO algorithm, FPA algorithm, FA algorithm and BA algorithm on 11 Standard Test functions. The experimental results show that DE-GOA algorithm has obvious advantages over other algorithms.

Key words: Grasshopper Optimization Algorithm; Differential Evolution; iterative optimization; function optimization; population diversity

1 引言

智能群优化算法被广泛应用于各类工程优化问题^[1-5]。蝗虫算法是 S Saremi 等于 2017 年提出的一种新兴群智能算法, 该算法模拟蝗虫群的迁移群体行为来对各类问题进行优化。该算法被应用于工程实践各领域, 如文献

[6]利用基于蝗虫优化算法分布式模型控制方法对太阳能式无人机的协同目标跟踪轨迹进行优化。文献[7]利用蝗虫优化算法对微电网进行快速频率控制。文献[8]利用蝗虫优化算法对交换膜燃料电池特性进行优化, 取得了良好的效果。文献[9]利用蝗虫算法对 SAR 图像进行分割优化, 取得了较好的效果。

传统的蝗虫优化算法与其他智能优化算法相似, 存在着容易陷入局部最优值, 特别是在多峰优化问题时, 容

*基金项目: 青海省自然科学基金项目(20180ZJ-776); 国家教育部春晖计划(Z2017051); 国家社科基金(19XTQ004)

收稿日期: 2020-09-28

易陷入局部最优,过早收敛,无法获得全局最优值。

本文针对蝗虫算法的不足,利用差分进化算法对蝗虫优化算法进行改进,提高种群多样性,提高蝗虫优化算法跳出局部最优值的能力,进一步提升蝗虫优化算法的全局搜索能力。

2 蝗虫算法

蝗虫算法是S Saremi等于2017年提出的一种新兴群智能算法^[10-11],该算法模拟蝗虫群的迁移群体行为,来对各类问题进行优化。蝗虫算法的数学表达式如(1)所示,在蝗虫算法中,蝗虫算法的位置即为优化问题的解,第*i*个蝗虫的位置用 X_i 表示。

$$X_i = S_i + G_i + A_i \quad (1)$$

其中 S_i 代表第*i*个蝗虫的社会行为, G_i 代表第*i*个蝗虫的重力, A_i 代表第*i*个蝗虫受到的风平流的作用。

其中蝗虫的社会行为可以用如下表达式所示:

$$S_i = \sum_{\substack{j=1 \\ j \neq i}}^N s(d_{ij}) \hat{d}_{ij} \quad (2)$$

其中 d_{ij} 代表第*i*只蝗虫到第*j*只蝗虫的距离,即 $d_{ij} = |X_i - X_j|$ 。 s 代表蝗虫社会互动行为影响的函数,如式(3)所示。 \hat{d}_{ij} 代表第*i*只蝗虫到第*j*只蝗虫的单位向量,即 $\hat{d}_{ij} = \frac{x_j - x_i}{d_{ij}}$ 。

蝗虫的社会互动行为影响函数 s 如下式所示:

$$s(r) = fe^{\frac{-r}{l}} - e^{-r} \quad (3)$$

其中 f 代表吸引力, l 代表吸引尺度。

蝗虫在飞行式会受到重力的作用,重力 G 对蝗虫的影响可以用如下式子表示:

$$G_i = -g\hat{e}_g \quad (4)$$

其中 g 代表引力常数, \hat{e}_g 代表朝向地球中心的统一矢量。

蝗虫在飞行时会受到风的作用,风平流 A 对蝗虫的影响可以用如下式子表示:

$$A_i = u\hat{e}_w \quad (5)$$

其中 u 代表恒定漂移, \hat{e}_w 代表朝向风的统一矢量。

于是将式(2),(3),(4),(5)代入式(1)可得:

$$X_i = \sum_{\substack{j=1 \\ j \neq i}}^N s(|x_j - x_i|) \frac{x_j - x_i}{d_{ij}} - g\hat{e}_g + u\hat{e}_w \quad (6)$$

将蝗虫的边界信息,即优化问题的边界信息包括上边界 ub 和下边界 lb 引入式(6)中,由于重力蝗虫的影响非常小,省去重力的影响,并且假设风平流是朝向目标解

(\hat{T}_d)的,可得蝗虫的位置更新方程为:

$$X_i^d = c \left(\sum_{\substack{j=1 \\ j \neq i}}^N c \frac{ub_d - lb_d}{2} s(|x_j^d - x_i^d|) \frac{x_j^d - x_i^d}{d_{ij}^d} \right) + \hat{T}_d \quad (7)$$

其中,参数 c 的更新表达式如下所示:

$$c = c_{\max} - l \frac{c_{\max} - c_{\min}}{L} \quad (8)$$

其中 c_{\min} 代表最小值, c_{\max} 代表最大值, L 代表最大迭代次数, l 代表当前迭代次数。通常 c_{\max} 的值为1, c_{\min} 的值为0.00001。

3 差分进化算法

差分进化算法是一种整体结构类似与遗传算法的优化算法。该算法的主要思想是通过群体内个体变异,再经过交叉,选择以达到对群体优化的目的。该算法主要分为初始化,变异,交叉与选择这几个过程。

(1) 初始化一个种群,设种群的数量为 N : $X(t) = \{X_1(t), X_2(t), \dots, X_N(t)\}$;其中 t 表示迭代次数。

(2) 差分进化的变异过程是差分进化算法的关键,随机从群体中选择两个不同的个体,进行如下操作:

$$V_{ij}(t+1) = X_{ij}(t) + F(X_{im}(t) - X_{in}(t)) \quad (9)$$

其中 ij 代表第*i*维第*j*个个体, m, n 都是大于1小于 N 且不相同的随机整数。 F 为0到1之间的缩放因子。 $V_{ij}(t+1)$ 是变异后的个体向量。

(3) 差分进化算法的交叉操作如下:

$$U_{ij}(t) = \begin{cases} V_{ij}(t), \text{rand}(0,1) \leq CR \text{ or } j = \text{rand}(i,n) \\ X_{ij}(t), \text{else} \end{cases} \quad (10)$$

其中 CR 为0到1之间的数,叫做交叉概率。

(4) 差分进化算法的选择过程:

$$X_i(t+1) = \begin{cases} U_i(t+1), f(U_i(t+1)) < f(X_i(t)) \\ X_i(t), \text{esle} \end{cases} \quad (11)$$

其中, f 为适应度函数,这里对应最小化问题,从差分进化的选择过程中可以看到,差分进化算法用适应度值较低的结果代替原来的结果,即用变异后产生的更好的结果来代替之前较差的结果,起到了优化选择的作用,使整个种群向更加优化的方向靠近。

4 基于差分进化改进的蝗虫算法

本文在蝗虫算法的基础上引入差分进化算法,将差分进化算法的优点融入蝗虫算法中提出了一种基于差分进化改进的蝗虫算法(DE-GOA)。在算法的搜索过程引进差分进化,增加种群的多样性,提高算法的全局寻优能

力。在基本的蝗虫算法搜索结束后,对蝗虫种群进行差分进化操作,使得群体之间的信息共享性更好,同时使种群的多样性增加。基于差分进化改进的蝗虫算法具体流程如图1所示。

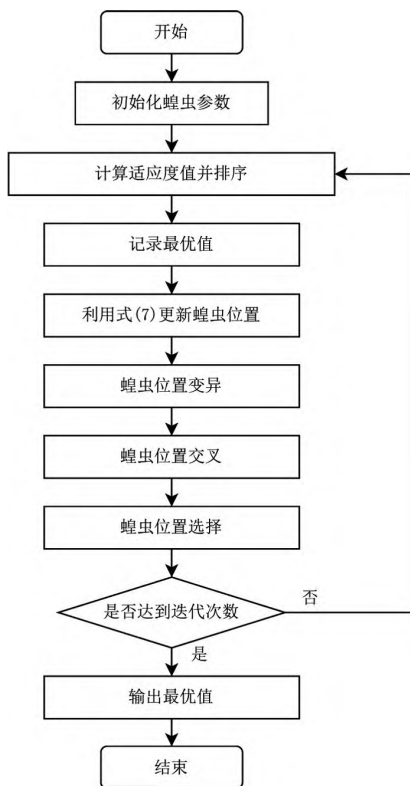


图1 基于差分进化改进的蝗虫算法流程图

主要步骤如下:

- (1) 初始化蝗虫种群数量,迭代次数, C_{min} , C_{max} 的值,权重系数,交叉,变异概率等等。
- (2) 计算各蝗虫个体的适应度值。
- (3) 对蝗虫个体的适应度值进行排序并记录最优位置。
- (4) 利用式(7)对蝗虫的位置进行更新。
- (5) 对蝗虫个体进行变异操作。
- (6) 对蝗虫个体进行交叉操作。
- (7) 对蝗虫个体进行选择操作。
- (8) 是否达到迭代结束条件,如果没有则返回(2),否则停止迭代输出最优个体位置及适应度值。

5 实验与分析

为了验证本文的DE-GOA算法的优化效果和稳定性。将本文算法与其他6种优化算法包括粒子群算法PSO^[12],遗传算法GA^[13],蝙蝠算法BA^[14],蝗虫算法GOA,花授粉算法FPA^[15],萤火虫算法FA^[16],在11个标准测试函数上进行寻优求解,对算法性能进行对比。为了保证

实验结果的可信性,各算法的种群数量均为30,最大迭代次数为500,对每个测试函数测试300次,取各数据的平均值进行对比。

本文在Centrino 2 PC,2.3 Ghz,Matlab 2014a环境中进行实验,各算法的参数设置如表1所示。各测试函数如表2所示。表3为各优化算法的测试结果,测试结果主要包括平均误差和标准差。平均误差衡量算法的寻优能力,标准差衡量算法的可重复性。

表1 各算法的参数设置

Algorithm	Parameter	Value
PSO	Topology	Fully connected
	Cognitive and social constants	1.5,1.5
	Inertial weight	decrease from 0.6 to 0.3
GA	Type	Real coded
	Selection	Roulette wheel
	Crossover	Single point(Probability=1)
BA	Mutation	Uniform(probability=0.01)
	Loudness(A), pulse rate(r)	0.5,0.5
FPA	Frequency min and max	0.2
FA	probability switch(p)	0.4
GOA	Alpha,beta,and gamma	0.2,1,1
DE-GOA	cmin,cmax	0.00001,1
	Crossover probability	0.9
	Differential weight	0.5

表2 测试函数信息

Function	Dim	Range	fmin
$f_1(x) = \sum_{i=1}^n x_i^2$	30	[-100,100]	0
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10,10]	0
$f_3(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	30	[-100,100]	0
$f_4(x) = \max \{ x_i , 1 \leq i \leq n\}$	30	[-100,100]	0
$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2]$	30	[-30,30]	0
$f_6(x) = \sum_{i=1}^{n-1} ([x_i + 0.5])^2$	30	[-100,100]	0
$f_7(x) = \sum_{i=1}^n ix_i^4 + \text{random}(0,1)$	30	[-128,128]	0
$f_8(x) = \sum_{i=1}^n -x \sin(\sqrt{ x_i })$	30	[-500,500]n	-418.9829*n
$f_9(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-512,512]	0
$f_{10}(x) = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	30	[-32,32]	0
$f_{11}(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30	[-600,600]	0

表3 实验结果

F	PSO		GA		BA		FPA	
	ave	std	ave	std	ave	std	ave	std
f1	0.2390	0.5612	0.4221	0.5202	0.9682	1.0000	0.0329	0.0684
f2	0.0094	0.0013	0.0100	0.0003	1.0000	1.0000	0.0131	0.0007
f3	0.2612	0.3547	0.2022	0.0710	1.0000	1.0000	0.0000	0.0000
f4	0.4766	0.4730	0.7345	0.2384	0.9138	1.0000	0.3219	0.4115
f5	0.0386	0.0944	0.0746	0.0931	1.0000	1.0000	0.0060	0.0345
f6	0.7786	0.4808	0.1433	0.1932	1.0000	1.0000	0.0078	0.0189
f7	0.1349	0.1648	0.4416	0.2028	1.0000	1.0000	0.0376	0.0459
f8	0.7425	0.0016	0.0000	0.0006	0.0148	1.0000	0.0351	0.0006
f9	0.6520	1.0000	1.0000	0.0000	0.7022	0.7517	0.6558	0.4179
f10	0.6140	0.2426	0.8528	0.1415	0.9645	0.1155	0.7150	0.3878
f11	0.8184	0.3512	0.1641	0.2865	0.9902	1.0000	0.0124	0.0058
F	FA		GOA		DE-GOA			
	ave	Std	ave	std	ave	std		
f1	0.1581	0.0758	0.0031	0.0010	0.0004		0.0001	
f2	0.0076	0.0001	0.0020	0.0010	0.0000		0.0002	
f3	0.0607	0.0160	0.0010	0.0203	0.0003		0.0001	
f4	0.3736	0.2146	0.0004	0.0000	0.0000		0.0000	
f5	0.0078	0.0058	0.0009	0.0000	0.0001		0.0000	
f6	0.0434	0.0190	0.0000	0.0000	0.0000		0.0000	
f7	0.0550	0.0164	0.0000	0.0000	0.0000		0.0000	
f8	0.6160	0.0000	1.0000	0.0002	0.5670		0.0002	
f9	0.8357	0.0329	0.0000	0.0007	0.0000		0.0000	
f10	0.7038	0.0410	0.0975	1.0000	0.0235		0.0032	
f11	0.0548	0.0070	0.0000	0.0000	0.0000		0.0000	

从表3的实验结果来看,对于f1,f2,f3,f4,f5,f8,f10函数,DE-GOA算法的平均误差最小,次之为GOA算法,但是DE-GOA的误差仍然比GOA算法的误差小一个数量级,而其他算法的误差相比DE-GOA算法的误差相差比较大,其他5种算法的误差比DE-GOA算法的误差大几个甚至几十个数量级,表明DE-GOA算法的寻优能力最好。从标准差来看,DE-GOA算法的标准差最小,GOA算法次之,其他5种算法的标准差均较大。表明DE-GOA算法的稳定性更强。对于f6,f7函数,DE-GOA算法与GOA算法均能取得较好的结果,并且相比其他5种算法,平均误差和标准差值均最小,相比其他5种算法仍然具有明显的优势。对于f9函数,虽然DE-GOA算法与GOA算法的平均误差一样,但是DE-GOA算法的标准差仍然比GOA算法的标准差小一个数量级,表明DE-GOA算法相比GOA算法在算法稳定性上仍然有较大的提升。综合来看,DE-GOA算法由于引进差分进化,改善了GOA算法的寻优能力,无论是算法的性能还是算法的稳

定性均得到了提高。

6 结束语

本文成功的将差分进化算法与蝗虫优化算法结合,提出了一种基于差分进化改进的蝗虫优化算法。该算法针对基本蝗虫算法,容易陷入局部最优,收敛精度不高的缺点,将差分进化引入蝗虫算法中,增加了种群的多样性,实验结果表明本文提出的优化算法相比基础的蝗虫算法能够起到明显地改进的作用。并且相比其他5种智能优化算法无论是在精度还是稳定性上均具有明显的优势。

参考文献:

- [1] 范旭,曹中清,陈彬彬.基于风驱动优化的霍夫变换算法[J].计算机工程与设计,2017(6):1522-1525.
- [2] 范旭,曹中清.基于灰色关联与风驱动优化的图像匹配算法[J].计算机工程与设计,2018(8):35.
- [3] 魏晶茹,马瑜,夏瑞,等.基于分数阶粒子群的Otsu图像分割算法[J].计算机工程与设计,2017(12):3284-3290.
- [4] 张伟伟,刘勇进,彭君君.改进鸟群算法用于SVM参数选择[J].计算机工程与设计,2017(12):3267-3271.
- [5] 刘笥笥,梁弼,袁小艳,等.基于改进布谷鸟算法的彩色图像多阈值分割[J].计算机工程与设计,2016,37(12):3322-3326.
- [6] Wu J,Wang H,Li N,et al.Distributed trajectory optimization for multiple solar-powered UAVs target tracking in urban environment by Adaptive Grasshopper Optimization Algorithm[J].Aerospace Science and Technology,2017(70):497-510.
- [7] Barik A K,Das D C.Expeditious frequency control of solar photovoltaic/biogas/biodiesel generator based isolated renewable microgrid using grasshopper optimisation algorithm[J].IET Renewable Power Generation,2018,12(14):1659-1667.
- [8] El-Fergany A A.Electrical characterisation of proton exchange membrane fuel cells stack using grasshopper optimiser[J].IET Renewable Power Generation,2017,12(1):9-17.
- [9] Saremi,Shahzad,Mirjalili,Seyedali,Lewis,Andrew. Grasshopper Optimisation Algorithm:Theory and application[J].Advances in Engineering Software,2017,105(5):30-47.
- [10] Saremi S,Mirjalili S,Lewis A.Grasshopper optimisation algorithm:Theory and application[J].Advances in Engineering Software,2017(105):30-47.
- [11] Mirjalili S Z,Mirjalili S,Saremi S,et al.Grasshopper optimization algorithm for multi-objective optimization problems[J].Applied Intelligence,2018,48(4):805-820.
- [12] Chen W,Panahi M,Pourghasemi H R.Performance evaluation of GIS-based new ensemble data mining

techniques of adaptive neuro-fuzzy inference system(ANFIS) with genetic algorithm(GA), differential evolution(DE), and particle swarm optimization(PSO) for landslide spatial modelling[J]. Catena, 2017(157): 310–324.

[13] Tan J, Kerr W L. Determination of glass transitions in boiled candies by capacitance based thermal analysis(CTA) and genetic algorithm(GA)[J]. Journal of Food Engineering, 2017(193): 68–75.

[14] Tharwat A, Hassanien A E, Elnaghi B E. A ba-based algorithm for parameter optimization of support vector machine[J]. Pattern Recognition Letters, 2017(93): 13–22.

[15] Oda E S, Abdelsalam A A, Abdel-Wahab M N, et al. Distributed generations planning using flower pollination algorithm for enhancing distribution system voltage stability[J]. Ain Shams Engineering Journal, 2017, 8(4):

593–603.

[16] Kaabeche A, Diaf S, Ibtouen R. Firefly-inspired algorithm for optimal sizing of renewable hybrid system considering reliability criteria[J]. Solar Energy, 2017(155): 727–738.

(上接第6页)

Based Methodology for Simultaneous Concrete Defect Detection and Geolocalization[J]. Computer-Aided Civil and Infrastructure Engineering, 2018, 33(7): 527–544.

[41] Dorafshan S, Thomas R J, Maguire M. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete[J]. Construction and Building Materials, 2018(186): 1031–1045.

[42] Yang X, Li H, Yu Y, et al. Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network[J]. Computer-Aided Civil and Infrastructure Engineering, 2018, 33(12): 1090–1109.

[43] Ni F, Zhang J, Chen Z. Pixel-level crack delineation in images with convolutional feature fusion[J]. Structural Control & Health Monitoring, 2019, 26(1): e2286.

[44] Liu Y F, Nie X, Fan J S, et al. Image-based crack assessment of bridge piers using unmanned aerial vehicles and three-dimensional scene reconstruction[J]. Computer-Aided Civil and Infrastructure Engineering, 2020, 35(5): 511–529.

[45] Mondal T G, Jahanshahi M R, Wu R T, et al. Deep learning-based multi-class damage detection for autonomous post-disaster reconnaissance[J]. Structural Control & Health Monitoring, 2020, 27(4): e2507.

[46] Ju H Y, Li W, Tighe S S, et al. Cracku-net: A novel deep convolutional neural network for pixelwise pavement crack detection[J]. Structural Control & Health Monitoring, 2020, 27(8): e2551.

[47] Zhang Y, Sun X W, Loh K J, et al. Autonomous bolt loosening detection using deep learning[J]. Structural Health Monitoring—an International Journal, 2020, 19(1): 105–122.

[48] Kim H, Sim S H. Automated peak picking using region-based convolutional neural network for opera-

作者简介: 宋长新(1971-), 女, 博士, 教授, 研究方向: 数据挖掘与分析、生物信息学和图像处理。

tional modal analysis[J]. Structural Control & Health Monitoring, 2019, 26(11): e2436.

[49] Zhang B, Hong X, Liu Y. Multi-Task Deep Transfer Learning Method for Guided Wave-Based Integrated Health Monitoring Using Piezoelectric Transducers[J]. IEEE Sensors Journal, 2020, 20(23): 14391–14400.

[50] Zhang Z, Pan H, Wang X, et al. Machine Learning-Enriched Lamb Wave Approaches for Automated Damage Detection[J]. Sensors, 2020, 20(6): 1790.

[51] Sajedi S O, Liang X. Vibration-based semantic damage segmentation for large-scale structural health monitoring[J]. Computer-Aided Civil and Infrastructure Engineering, 2020, 35(6): 579–596.

[52] Chen C M. Citespace II: Detecting and visualizing emerging trends and transient patterns in scientific literature[J]. Journal of the American Society for Information Science and Technology, 2006, 57(3): 359–377.

[53] Chen C, Hu Z, Liu S, et al. Emerging trends in regenerative medicine: a scientometric analysis in citespace[J]. Expert Opinion on Biological Therapy, 2012, 12(5): 593–608.

作者简介: 张学森(1981-), 男, 高级工程师, 研究方向: 风力发电塔筒; 海上风电单桩; 导管架。

通信作者: 吴香国(1976-), 男, 教授, 研究方向: 新材料及新型结构、能源工程结构。