

A Modified Differential Evolution Algorithm Based on Invasive Weed Optimization

Qingbo Lu, Wei Cui, Shenghui Yan, Tingfeng Li

Department of Mechanical Engineering, Zhengzhou Technical College, Zhengzhou Henan
Email: jh0262@126.com

Received: Jan. 16th, 2019; accepted: Jan. 28th, 2019; published: Feb. 12th, 2019

Abstract

DE/rand/1/bin model of DE Algorithm has good global performance, but its convergence speed is slow. A modified differential evolution optimization algorithm named IWOMDE was presented in this paper based on Invasive Weed Optimization algorithm. The IWOMDE algorithm incorporated IWO's design philosophy into Differential Evolution (DE) algorithm. The IWOMDE algorithm divided the evolution population into elite individuals and poor individuals based on the average fitness of population. The elite individuals can be evolutionary many times, but the poor individuals are less evolved or "stagnate" in order to maintain the diversity of population. The simulation results showed the hybrid optimization algorithm has the advantage of searching effectively and being fairly robust to initial conditions.

Keywords

Differential Evolution (DE) Algorithm, Invasive Weed Optimization (IWO), Elite Individual, Function Optimization

一种基于入侵杂草算法改进的差分进化算法

卢青波, 崔巍, 闫生辉, 李廷锋

郑州职业技术学院机械工程系, 河南 郑州
Email: jh0262@126.com

收稿日期: 2019年1月16日; 录用日期: 2019年1月28日; 发布日期: 2019年2月12日

摘要

差分进化算法(Differential Evolution, DE)的DE/rand/1/bin模式具有良好的全局性能, 但其收敛速度慢, 针对此, 应用入侵杂草算法(Invasive Weed Optimization, IWO)的设计思想, 研究提出一种改进的

差分进化算法(Modified Differential Evolution Algorithm based on Invasive Weed Optimization, IWOMDE)。该算法依据IWO算法思想,以群体平均适应度为划分依据,将群体中的个体划分为优秀个体与较差个体,对优秀个体使其能够多进化,而较差个体少进化或者“停滞”以维持种群的多样性。仿真实验结果表明, IWOMDE算法对多峰多模态函数具有优化效率高、优化精度高的特性。

关键词

差分进化算法, 入侵杂草算法, 精英个体, 函数优化

Copyright © 2019 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

进化算法是模仿生物进化过程设计的现代优化方法,作为一种有效的随机优化方法,被广泛应用于求解复杂优化问题。DE 算法[1]使用浮点矢量进行个体编码,通过简单的变异、交叉及竞争算子实现在连续空间中的随机搜索。DE 算法原理简单,易于理解和实现,在许多复杂优化问题中得到了应用[2] [3] [4] [5] [6]。DE 算法同其它进化算法类似,需要平衡算法的全局搜索能力与其收敛速度。这种平衡可以通过调节算法的控制参数来实现,但往往控制参数的调节效果有限[7] [8],对于 DE 算法,交叉率大则收敛速度快,但其全局搜索能力下降,易陷入局部最优,从而使算法的收敛精度下降。因此,许多学者从不同类型算法的优势出发,结合 DE 算法的全局性能和一些局部优化算法的局部搜索能力,研究提出了不同的混合优化算法[9] [10]。这些对 DE 算法的改进从不同的侧面使算法的性能得到了提升,但都未对 DE 算法的候选解产生过程有较大的改变。DE 算法候选解的产生采用的从种群中随机选择不同个体,通过线性运算产生,该变异过程并没有考虑所选择个体的适应度信息,随机性强,有利于保持种群的多样性,但同时也使算法的收敛速度较慢。

入侵杂草算法(Invasive Weed Optimization, IWO) [11]于 2006 年提出,是模拟杂草在自然界中生长过程的一种新的优化算法,该算法的种群是由杂草组成,通过杂草在空间的扩散、生长、繁殖和竞争进行寻优,算法具有鲁棒性强、易于实现等特点。IWO 算法在进化过程中,杂草会依据适应度值进行繁殖,适应度较优的个体会产生较多的种子,且所产生的种子是以正态分布在该种子的周围。这种思想使得较优个体占有较强的优势,且兼顾了局部搜索。本文将基于该思想研究提出一种改进的差分进化算法(A Modified Differential Evolution Algorithm based on Invasive Weed Optimization, IWOMDE)。

2. IWOMDE 算法

2.1. 差分进化算法

DE 算法的进化个体采用实向量进行编码,采用均匀分布的随机数产生初始个体。令 $\mathbf{x}_i(g)$ 代表第 g 代的第 i 个个体, $\mathbf{x}_i^L \leq \mathbf{x}_i(g) \leq \mathbf{x}_i^U$, 则

$$\mathbf{x}_i(g) = (\mathbf{x}_{i1}(g), \mathbf{x}_{i2}(g), \dots, \mathbf{x}_{in}(g)), i = 1, 2, \dots, NP; g = 1, 2, \dots, T_{\max} \quad (1)$$

式中, \mathbf{x}_i^U 、 \mathbf{x}_i^L 分别为个体的上、下限, NP 为种群大小, T_{\max} 为最大进化代数。

DE 算法就是由这 NP 个个体构成的种群在搜索空间不断进化来进行寻优的。DE 算法候选解产生模式有很多策略[1],其中 DE/rand/1/bin 策略应用较为广泛,且具有全局搜索能力较好,收敛速度较慢的特性,

因此, 本文选择该进化模式为研究对象。以求解最小值问题为例说明差分进化算法的进化过程如下。

1) 初始化种群

在 n 维实数空间按式(2)随机产生 NP 个个体

$$\mathbf{x}_{ij}(0) = \mathbf{x}_{ij}^L + \text{rand}(0,1)(\mathbf{x}_{ij}^U - \mathbf{x}_{ij}^L) \quad (2)$$

式中, $\text{rand}(0,1)$ 是 $[0,1]$ 上服从均匀分布的随机数。

2) 变异算子

首先随机从种群中选择 3 个不同个体 $\mathbf{x}_{p1}, \mathbf{x}_{p2}, \mathbf{x}_{p3}$, 且 $p1 \neq p2 \neq p3 \neq i$ 则变异算子为

$$\mathbf{h}_{ij}(g) = \mathbf{x}_{p1j} + F(\mathbf{x}_{p2j} - \mathbf{x}_{p3j}) \quad (3)$$

式中, F 为缩放因子。

3) 交叉算子

交叉算子可以增加种群的多样性, 其操作过程如式(4)所示。

$$\mathbf{v}_{ij}(g+1) = \begin{cases} \mathbf{h}_{ij}(g), \text{rand}(0,1) \leq \text{CR} \text{ 或 } j = j_{rand} \\ \mathbf{x}_{ij}(g), \text{rand}(0,1) > \text{CR} \text{ 或 } j \neq j_{rand} \end{cases} \quad (4)$$

式中, CR 为交叉率, $\text{CR} \in [0,1]$, j_{rand} 是 $[1,n]$ 上的随机整数, 这种交叉模式确保 $\mathbf{v}_{ij}(g+1)$ 中至少有一位来自 $\mathbf{h}_{ij}(g)$ 。

4) 选择算子

DE 算法的选择策略采用“贪婪”策略, 由评价函数对向量 $\mathbf{v}_i(g+1)$ 和向量 $\mathbf{x}_i(g)$ 比较, 保留较优个体, 即

$$\mathbf{x}_i(g+1) = \begin{cases} \mathbf{v}_i(g+1), f(\mathbf{v}_i(g+1)) < f(\mathbf{x}_i(g)) \\ \mathbf{x}_i(g), f(\mathbf{v}_i(g+1)) \geq f(\mathbf{x}_i(g)) \end{cases} \quad (5)$$

反复执行式(3)到(5), 直到达到算法预设的终止条件。

2.2. IWOMDE 算法

DE 算法的变异策略有多种, 其中 DE/rand/1/bin 变异策略具有全局搜索能力强、收敛速度慢的特性。由式(3)可知, DE/rand/1/bin 变异过程中, 其候选解的产生是采用的是种群中随机选择的 3 个不同个体, 通过线性运算产生的。该变异过程并没有考虑所选择个体的适应度信息, 随机性强, 有利于保持种群的多样性, 但同时也使算法的收敛速度较慢。

DE 算法的随机搜索过程, 使得算法的局部搜索能力较差, 收敛速度慢, 这主要是由于进化过程中, 尤其是在候选解的产生过程中未考虑个体的适应度引起的, 因此, 借鉴 IWO 算法的进化思想, 将适应度的信息融入候选解的生成过程, 使较优的个体能够产生较多的候选解, 较差的个体产生较少的候选解或者保持停滞状态。这样不但能够使最优个体引导群体快速收敛, 而且由于有个体随机停滞现象, 能够很好地保持群体的多样性。

1) 优秀个体的选取

群体适应度均值

$$f_{avg} = \frac{1}{NP} \sum_{i=1}^{NP} f_i \quad (6)$$

式中, f_i 为第 i 个个体的适应度。

选取 $f_i < f_{avg}$ 个体为最优个体, 其余个体为较差个体。

2) 修正的变异算子

基于以上个体定义, 针对差分进化算法的变异算子, 对不同类型的个体进行不同的变异算子设计。对于优秀个体, 使其以较大概率进化, 若进化成功, 则继续执行进化过程以加快算法的收敛速度。对于较差个体, 以较小概率进化或者停滞, 使其能够保证群体的多样性。

优秀个体进化过程如下:

Step1: 变异算子

if ($rand(0,1) < P_1$)

$$h_{ij}(g) = [1 + \sigma(0,1)]x_{ij} + F(\mathbf{x}_{gbestj} - \mathbf{x}_{ij}) \quad (7)$$

else

$$h_{ij}(g) = \mathbf{x}_{p1j} + F(\mathbf{x}_{p2j} - \mathbf{x}_{p3j})$$

Step2: 执行式(4)的交叉算子

Step3: 执行式(5)的选择算子

Step4: if [$f(\mathbf{x}_i(g+1)) < f(\mathbf{x}_i(g))$], 转 Step1 继续进化。

较差个体的进化过程如下:

if ($rand(0,1) < 1 - P_1$)

$$h_{ij}(g) = \mathbf{x}_{p1j} + F(\mathbf{x}_{p2j} - \mathbf{x}_{p3j})$$

执行式(4)的交叉算子与式(5)的选择算子。

else

停滞。

式中 P_1 为优秀个体进化概率, $\sigma(0,1)$ 为服从均值为 0, 方差为 1 的正态分布随机数。

设置 P_1 为较大值(如 0.9)则优秀个体会以该概率执行局部搜索, 能够增强算法的局部探索能力; 而较差个体进化概率为 $1 - P_1$, 这些个体起到维持种群多样性的作用。若 P_1 为 0, 则优秀个体的变异算子中的式(7)不会被执行, 变异算子退化为 DE 算法的变异过程, 只是在优秀个体更新完后, 若竞争成功, 则继续进化; 而此时, 对于较差个体, 会按照 DE 算法的进化过程正常进化, 不会出现停滞现象。若 P_1 设置为 1, 则对于优秀个体的变异算子完全由式(7)决定, 而对于较差个体则处于完全停滞状态, 即整个算法只有优秀个体进化, 其余个体处于停滞状态。IWOMDE 算法正是基于以上思想进行设计的, 其实现步骤如下。

Step1: 设置种群规模 NP、交叉概率 CR、缩放因子 F , 计算精度 ε 及优秀个体进化概率 P_1 , 在参数空间随机初始化每一个个体, 设置最大进化代数 T , 令 $t = 1$ 。

Step2: 计算当前种群的最优适应度 $bestfitness$ 及最优个体 \mathbf{x}_{gbest} 。

Step3: 若最优适应度 $bestfitness$ 达到精度要求或者迭代次数达到最大, 则输出当前最优适应度值, 退出; 否则, 转 Step4。

Step4: 依据式(6)计算当前群体的平均适应度;

Step5: 对群体中的所有个体执行

若 $f_i < f_{avg}$ 按优秀个体进行进化; 否则, 按较差个体进化。

Step6 $t = t + 1$, 转 Step2。

3. 仿真结果及分析

3.1. 仿真测试设置

由于 IWOMDE 算法在 DE 算法的基础上改进的, 并且引入了优秀个体进化概率 P_1 , 因此, 首先对该

引入参数对算法的影响进行仿真分析，然后对 IWOMDE 算法与其它算法进行比较分析。

采用 5 个典型的测试函数，与 DE 算法(DE/rand/1/bin 模式)进行比较。 f_1 为 Schaffer 函数，该函数在全局极大点范围内有无限多的局部极大点，很难全局最优化。 f_2 为 Ackley 函数，该函数带有指数项，存在大量局部最优解。 f_3 为 Rosenbrock 函数，该函数是一个非凸病态函数，很难实现全局最优化。 f_4 为 Rastrigin 函数，该函数是一个多峰函数，在定义域内大约存在 $10n$ 个局部极小点。 f_5 为 Griewank 函数，该函数是一个多峰函数，存在大量局部极小点。

$$f_1 = 0.5 - \frac{(\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5)}{(1 + 0.001(x_1^2 + x_2^2))^2} \quad -100 \leq x_1, x_2 \leq 100,$$

$$f_2 = -20 \cdot \exp\left(-0.2 \sqrt{\frac{1}{10} \sum_{i=1}^{10} x_i^2}\right) - \exp\left(\frac{1}{10} \sum_{i=1}^{10} \cos(2\pi x_i)\right) + 20 + e, \quad -32.768 \leq x_i \leq 32.768$$

$$f_3 = \sum_{i=1}^{29} \left(100(x_i^2 - x_{i+1}^2)^2 + (x_i - 1)^2\right) \quad -2.048 \leq x_i \leq 2.048,$$

$$f_4 = \sum_{i=1}^{30} (x_i - 10 \cos(2\pi x_i) + 10) \quad -5.12 \leq x_i \leq 5.12,$$

$$f_5 = \sum_{i=1}^{30} \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad -600 \leq x_i \leq 600,$$

为了使算法的比较更公平，采用函数评价次数为终止条件。算法的其它参数设置如表 1 所示。

Table 1. Experimental parameter setting of algorithm

表 1. 算法实验参数设置

函数	种群规模	F	CR	评价次数
f_1	20	0.5	0.1	1e3
f_2	50	0.5	0.1	5e4
f_3	100	0.5	0.1	5e4
f_4	50	0.5	0.1	5e4
f_5	50	0.5	0.1	5e4

3.2. 测试结果及分析

为避免仿真结果的随机性，IWOMDE 算法对每个函数均独立运行 30 次。图 1~图 3 为 IWOMDE 算法在取不同优秀个体进化概率 P_1 时，求解 3 个 Benchmarks 函数时 30 次运行的平均适应度及最优适应度的比较。其中横坐标为 P_1 的取值，纵坐标为适应度值。从图中可以看出，优化结果随着 P_1 值的增大，平均最优适应度越来越优，而最优适应度存在随机性，这主要原因是群体的进化过程由式(5)决定的成分越来越多，能够进化的个体越来越少，较差个体在群体中的作用逐渐被忽略，从而改变了群体原有的进化进程，尽管在 P_1 取值为 1 时，获得的平均适应度最优，但其最优适应度不全是最优，因此在实验中建议设置 P_1 为较接近 1.0 的值，本文后面的实验中取该参数为 0.9。

表 2 给出了 IWOMDE 算法与 SMDE 算法[9]的优化结果比较，从表中数据可以看出，IWOMDE 算法除 Ronsenbrock 函数的优化结果要差于 SMDE 获得的结果之外，其余结果都要优于 SMDE 的优化结果。SMDE 算法采用的是固定进化代数作为终止条件，函数评价次数是未知的，而 IWOMDE 算法采用固定的

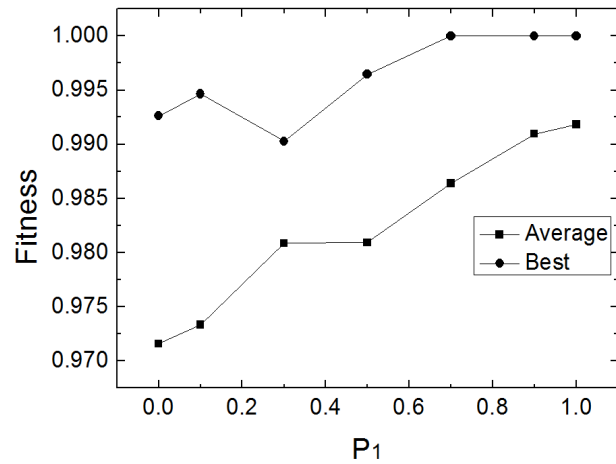


Figure 1. Comparison of mean fitness and optimal fitness of 30 runs at different P_1 of $f_1(x)$
图 1. 不同 P_1 值时 $f_1(x)$ 30 次平均适应度及最优适应度比较

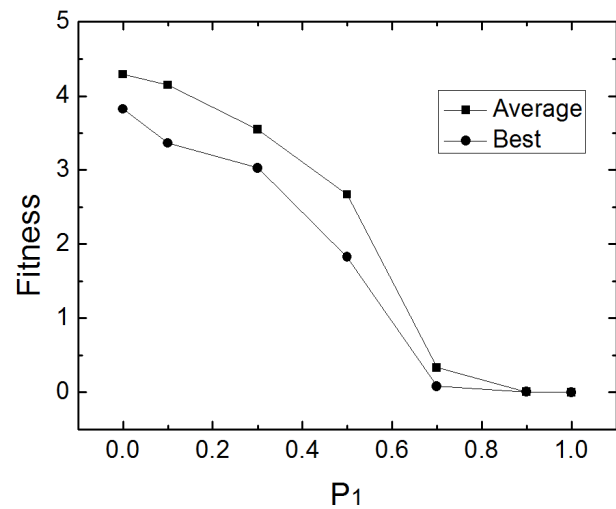


Figure 2. Comparison of mean fitness and optimal fitness of 30 runs at different P_1 of $f_2(x)$
图 2. 不同 P_1 值时 $f_2(x)$ 30 次平均适应度及最优适应度比较

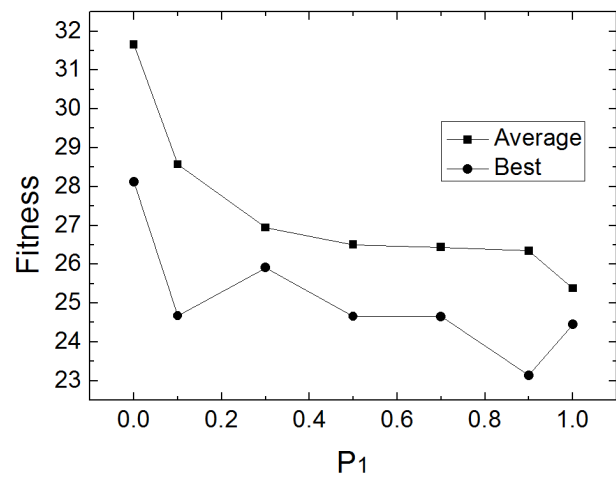


Figure 3. Comparison of mean fitness and optimal fitness of 30 runs at different P_1 of $f_3(x)$
图 3. 不同 P_1 值时 $f_3(x)$ 30 次平均适应度及最优适应度比较

函数评价次数为终止条件，因此，IWOMDE 算法在相同函数评价次数下，性能应优于 SMDE 算法的性能。

Table 2. Comparison of the optimization results of functions by IWOMDE and SMDE
表 2. IWOMDE 与 SMDE 函数优化结果比较

函数	方法	最优值	平均值	均方差
f_1	IWOMDE	1	0.9991	0.0021
	SMDE	1	0.9987	0.0034
f_2	IWOMDE	8.1747e-7	6.2791e-6	9.4309e-6
	SMDE	2.0971e-6	2.9977e-5	2.5258e-5
f_3	IWOMDE	27.2336	26.9906	0.1719
	SMDE	2.5940e-4	0.0029	0.0019
f_4	IWOMDE	0	4.7294e-6	3.5937e-6
	SMDE	6.9514e-7	2.0512e-5	2.5862e-5
f_5	IWOMDE	0	0	0
	SMDE	2.0122e-9	1.7481e-7	2.4493e-7

表 3 给出了 IWOMDE 算法与 ADE 算法[7]的优化结果比较，其中 IWOMDE 算法的最大函数评价次数对于 f_4 和 f_5 为 $2e4$ ，而对于 f_3 设置为 $2e5$ 。由于 ADE 算法 F 的范围为 $[0.5, 1.0]$ ，CR 的范围为 $[0.5, 1.0]$ ，因此 IWOMDE 设置 F 为 0.5，CR 设置为 $[0.5, 1.0]$ 上的均匀分布的随机数。三个测试函数的维数均为 20 维。表 3 给出了 IWOMDE 与 ADE 算法的优化结果比较，从表中数据可以看出，对于 Rosenbrock 函数，ADE 的优化结果要优于 IWOMDE 算法，但对于 Rastrigin 函数与 Griewank 函数，IWOMDE 算法表现出了更好的优化性能，不仅获得了较好的函数优化结果，而且只用了 ADE 函数评价次数的 1/10，表明收敛速度要远远优于 ADE 算法。

Table 3. Comparison of the optimization results of functions by IWOMDE and ADE
表 3. IWOMDE 与 ADE 函数优化结果比较

函数	方法	最优值	平均值	均方差
f_3	IWOMDE	11.6386	12.2816	0.3118
	ADE	5.09	9.81	7.04
f_4	IWOMDE	0	0	0
	ADE	82.88	1.07e2	97.72
f_5	IWOMDE	0	0	0
	ADE	6.46e-09	9.98e-09	9.15e-09

表 4 给出了 IWOMDE 算法与 ONDE 算法[12]的优化结果比较，四个测试函数的维数设置为 100 维，函数评价次数设置为 $6e4$ 。从表中数据可以看出，IWOMDE 算法与 ONDE 算法的结果各有优劣，IWOMDE 算法在 f_2 与 f_5 上的优化结果要优于 ONDE 算法，在 f_4 的最优值上也要优于 ONDE 算法，但在 f_3 上优化结果要比 ONDE 算法差很多。

Table 4. Comparison of the optimization results of functions by IWOMDE and ONDE
表 4. IWOMDE 与 ONDE 函数优化结果比较

函数	方法	最优值	平均值	均方差
f_2	IWOMDE	2.8753e-6	7.8734e-6	6.5832e-6
	ONDE	1.8659e-5	1.8667e-5	5.39e-9
f_3	IWOMDE	89.8923	93.2786	36.2768
	ONDE	1.7941e-3	1.813e-3	1.592e-5
f_4	IWOMDE	1.578e-10	3.76e-9	2.53e-9
	ONDE	2.217e-10	2.84e-10	5.0e-10
f_5	IWOMDE	8.1416e-10	7.6158e-9	2.794e-9
	ONDE	1.222e-8	1.373e-8	1.208e-9

综合以上分析可以发现, IWOMDE 算法在多峰函数上的优化结果要优于比较算法, 但对于单峰病态函数 Rosenbrock 函数, 优化结果较差。这主要原因是对于多峰函数, 由于群体进化过程中, 优秀个体使群体能够快速收敛到较优点, 而较差个体的停滞进化, 使群体维持了较好的多样性, 从而为优秀个体跳出局部最优点提供了较好的帮助。

4. 结语

DE 算法的 DE/rand/1/bin 进化模式具有全局搜索能力的特性, 被广泛应用于实际问题求解中, 但其收敛速度较慢, 局部搜索能较差。为使 DE 算法能够实现快速收敛, 且具有很强的搜索精度, 本文提出了一种简单易实现的 IWOMDE 算法, 该算法受入侵杂草算法(IWO)启发, 将进化群体依据群体适应度均值进行划分为优秀个体与较差个体, 对两种不同个体采用不同的变异算子。多个 Benchmarks 函数的优化结果表明, IWOMDE 具有收敛速度快, 对于多峰函数有较强的优化效率, 但对于单峰病态函数优化效率不高的特点, 可广泛应用于各种实际工程优化问题中。

参考文献

- [1] Storn, R. and Price, K. (1997) Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, **11**, 341-359. <https://doi.org/10.1023/A:1008202821328>
- [2] 刘波, 王凌, 金以慧. 差分进化算法研究进展[J]. 控制与决策, 2007, 22(7): 721-729.
- [3] Vesterstrom, J. and Thomsen, R. (2004) A Comparative Study of Differential Evolution, Particle Swarm Optimization, and Evolutionary Algorithms on Numerical Benchmark Problems. *Proceedings of the 2004 Congress on Evolutionary Computation*, Portland, 19-23 June 2004, 1980-1987. <https://doi.org/10.1109/CEC.2004.1331139>
- [4] 卢青波, 张学良, 温淑花, 等. 差异演化算法改进与应用[J]. 农业机械学报, 2010, 41(2): 193-197
- [5] 温淑花, 张学良, 卢青波, 等. 差异演化算法及其在机械优化设计中的应用[J]. 农业机械学报, 2008, 39(8): 135-139.
- [6] 卢青波, 张学良, 温淑花, 等. 基于差异演化算法的动压滑动轴承多目标优化[J]. 农业机械学报, 2013, 44(3): 230-236, 245.
- [7] 陈华, 范宜仁, 邓少贵. 基于 Logistic 模型的自适应差分进化算法[J]. 控制与决策, 2011, 26(7): 1105-1108.
- [8] Fu, H.J., Ouyang, D.T. and Xu, J.M. (2011) A Self-Adaptive Differential Evolution Algorithm for Binary CSPs. *Computers & Mathematics with Applications*, **62**, 2712-2718. <https://doi.org/10.1016/j.camwa.2011.06.053>
- [9] 刘洁, 吴亮红, 刘建勋. 基于单纯形算子的混合差分进化算法[J]. 计算机工程, 2009, 35(13): 179-182.
- [10] 阳春华, 钱晓山, 桂卫华. 一种混沌差分进化和粒子群优化混合算法[J]. 计算机应用研究, 2011, 28(2): 439-441.

-
- [11] Mehrabian, A.R. and Lucas, C. (2006) A Novel Numerical Optimization Algorithm Inspired from Weed Optimization. *Ecological Informatics*, **1**, 355-366. <https://doi.org/10.1016/j.ecoinf.2006.07.003>
- [12] 拓守恒, 汪文勇. 求解高维多模优化问题的正交小生境自适应差分演化算法[J]. 计算机应用, 2011, 31(4): 1094-1098.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2161-8801, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: csa@hanspub.org