

基于反向学习与机动爆炸烟花优化算法

李席广, 韩守飞, 拱长青

(沈阳航空航天大学 计算机学院, 辽宁 沈阳 110136)

摘 要: 针对烟花算法(Fireworks Algorithm, FWA)性能提升瓶颈和收敛速度较慢的问题, 通过引入反向学习策略和机动爆炸的机制, 提出了基于反向学习与机动爆炸烟花优化算法(Fireworks Algorithm based on Reverse learning and Maneuver explode, RLMEFWA). 该算法首先采用反向学习策略取代随机初始化生成初始种群以保证群体的多样性; 然后每个烟花根据其在当前群体中的位置的优劣情况来选择不同爆炸的方式, 处于较优位置的烟花选择机动爆炸方式, 以当前种群最优位置为基准, 改变自身位置信息向其靠近; 处于较劣位置的烟花选择非机动爆炸方式, 随机改变自身的位置信息. 分别把烟花算法(FWA)、标准粒子群算法(PSO)、增强烟花算法(EFWA)和 RLMEFWA 在 10 个典型的基准测试函数中进行仿真对比, 结果表明在收敛速度和计算精度以及稳定性方面 RLMEFWA 均优于其他三种算法.

关键词: 烟花算法; 机动爆炸; 基准函数; 最优位置; 反向学习

中图分类号: TP3 **文献标识码:** A **文章编号:** 1000-7180(2017)07-0105-08

Fireworks Algorithm Based on Reverse Learning and Maneuver Explode

LI Xi-guang, HAN Shou-fei, GONG Chang-qing

(School of Computer Science, Shenyang Aerospace University, Shenyang 110136, China)

Abstract: Aiming at the performance bottlenecks and slow convergence of fireworks algorithm (Fireworks Algorithm, FWA), By putting reverse learning and maneuver explode into FWA, this paper proposes fireworks algorithm based on reverse learning and maneuver explode (Fireworks Algorithm based on Reverse learning and Maneuver explode, RLMEFWA). In the algorithm, reverse learning strategy was introduced to generate initial population, which strengthened the diversity of population, then each firework applies the position in the current group to choose a different explode way. Fireworks in the better position to select maneuvering explosion mode whose random orbit closer to the optimum position. Fireworks in an inferior position explode select non-motorized mode with random orbit. In our simulation, we compare FWA, SPSO, EFWA and RLMEFWA with 10 typical benchmark functions. The results show that RLMEFWA is better than FWA, SPSO, and EFWA in terms of convergence speed and accuracy and stability.

Key words: fireworks algorithm; maneuver explode; benchmark functions; best position; reverse learning

1 引言

与其他智能优化算法类似,烟花算法也存在收敛速度慢、求解精度不高等缺陷,因此许多优化算法被相继提出.文献[1]提出对烟花算法改进主要有二

类:一类是在原有的烟花算法基础上进行算子的分析和改进.另一类是与其他启发式算法的混合算法的研究.文献[2]针对基本烟花算法的不足提出了增强的烟花算法 EFWA;文献[3]提出了基于适应度函数估计的加速型烟花算法(AcFWA);文献[4]提

出一种构造型烟花算法(IFWA),使用一种新型的烟花爆炸半径和爆炸火花数的计算方式,相对于基本烟花算法,获得了性能上的提升.文献[5-6]分别提出了二种自适应半径策略—动态搜索烟花算法(dynFWA)和自适应烟花算法(AFWA).以上烟花算法主要针对烟花的爆炸半径的改进,相对于基本烟花算法性能得到了提升.

针对基本烟花算法的缺点,本文提出了基于反向学习与机动爆炸烟花优化算法(RLMEFWA),该算法不但在收敛速度和计算精度上有一定的提高,而且增加了算法的稳定性.

2 烟花算法

烟花算法的基本原则:若烟花对应的适应度函数值越小,则该烟花爆炸产生的火花数量越多,爆炸幅度越小;反之,若烟花对应的适应度函数值越大,则该烟花爆炸产生的火花数量越少,且爆炸幅度越大.烟花算法一般由爆炸算子、变异爆炸、映射规则和选择策略四部分组成^[7].基于以上的原则,烟花算法的基本步骤可以概括如下:

```

1: 初始化随机选择  $N$  个烟花位置
2: while (当前迭代次数 < 最大迭代次数) do
3:   对于  $N$  个烟花
4:   for 所有烟花  $xi$  do
5:     计算每个烟花产生的火花数  $Si$ 
6:     计算每个烟花产生火花的幅度  $Ai$ 
7:     随机产生火花
8:   end for
9:   for  $k=1 \rightarrow m$  do //  $m$  是烟花高斯变异产生的火花数
10:    随机选择一个烟花  $xi$  并产生一个火花
11:  end for
12: 依据映射规则对火花进行映射
13: 应用选择策略得到下一代烟花群体
14: end while

```

3 基于反向学习与机动爆炸烟花优化算法

3.1 算法主要思想

一个烟花在爆炸的过程中可以有二种方式:(1)采用机动的爆炸方式,烟花比较集中,呈现好看的图案;(2)采用非机动爆炸的方式,烟花爆炸的时候周边零散的烟花.

针对以上的思想,本文对烟花的爆炸方式作如下规定:(1)烟花只采用机动爆炸和非机动爆炸二种方式;(2)烟花只根据当前的位置和当前最优的位置来确定自己下一步位置.

3.2 算法描述

3.2.1 基于反向学习的种群初始化

与其他智能优化算法一样,基本的烟花算法在迭代前同样采用随机初始化种群个体,由于事先对全局最优解没有任何先验知识,初始种群应均匀地覆盖在搜索空间中.对基于种群迭代的智能优化算法来说,最优保存策略是将上一代种群中的优良个体保存到下一代种群;若初始种群中的优良个体较少,则会直接影响算法的收敛.另外,初始种群的优劣还会影响算法的收敛速度.因此,本文采用反向学习策略^[8]来产生初始种群,即同时产生当前个体及相应的反向个体,然后比较选择适应度值较好的个体作为初始种群个体,从而提高了算法的搜索效率.

定义 1 反向点.假设在 $[L, U]$ 上存在数 x 的反向点定义为 $x' = L + U - x$.将对立点的定义扩展到 n 维空间,设 $p = (x_1, x_2, \dots, x_n)$ 为 n 维空间中的一个点,其中 $x_i \in [L_i, U_i], i = 1, 2, \dots, n$,则其反向点为 $p' = (x'_1, x'_2, \dots, x'_n)$,其中 $x'_i = L_i + U_i - x_i$.

根据上述定义,基于反向学习策略的种群初始化步骤如下:

(1)在搜索空间中随机初始化 N 个烟花个体作为初始种群 RP, N 为种群规模;

(2)根据定义 1,初始种群 RP 中的每个烟花个体的反向个体构成反向种群 OP;

(3)合并种群 RP 和 OP,将其 $2N$ 个烟花个体按照适应度值进行升序排序,选取适应度值前 N 个烟花个体作为初始种群.

3.2.2 爆炸算子

烟花算法初始化随机 N 个烟花,对于生成的 N 个烟花,应用爆炸算子以产生新的火花,爆炸算子是烟花算法的核心.爆炸火花将在烟花的爆炸半径范围内生成,烟花的爆炸半径也与烟花的适应度值有关.适应值差的烟花爆炸半径较大,这种全局搜索有利于烟花算法跳出局部最优,找出全局最优.式(1)和式(2)分别为烟花爆炸的火花个数和爆炸半径的具体计算公式.

$$S_i = m \frac{Y_{\max} - f(x_i) + \epsilon}{\sum_{i=1}^N (Y_{\max} - f(x_i)) + \epsilon} \quad (1)$$

$$\begin{cases} A_i = \hat{A} \frac{f(x_i) - Y_{\min} + \epsilon}{\sum_{i=1}^N (f(x_i) - Y_{\min}) + \epsilon}, A_i \geq A_{\min} \\ A_i = A_{\min}, \text{其他} \end{cases} \quad (2)$$

式中, S_i 为第 i 个烟花产生的火花个数; m 为常数,

用来限制产生的火花总数; $Y_{\max}$ 是当前种群中适应度最差的个体的适应度值; A_i 为第*i*个烟花的爆炸幅度范围,即爆炸的火花将在这个范围内随机产生位移,但不能超出这个范围; $\hat{A}$ 为常数,表示最大的爆炸幅度; $Y_{\min}$ 是当前种群中适应度最好的个体的适应度值; $f(x_i)$ 为个体 x_i 的适应度值; ϵ 为一个极小的常数。 $A_{\min}$ 是最小的爆炸幅度。

根据计算出的爆炸火花个数和爆炸半径,烟花*i*生成 S_i 个爆炸火花,实现过程为随机选择 d 个维度,对随机选取出的维度 $k \in \{1, 2, \dots, d\}$ 按式(3)进行位置偏移生成爆炸火花。

$$\Delta x_i^k = x_i^k + \text{rand}(0, A_i) \quad (3)$$

式中, $\text{rand}(0, A_i)$ 表示在爆炸幅度 A_i 内生成的均匀随机数。

以上是基本烟花的爆炸方式,RLMEFWA针对不同的烟花规定有不同的爆炸方式。

依据算法的主要思想,RLMEFWA现设计如下:设烟花*i*在*t*时刻的位置为 $X_{it} = (x_{i1t}, \dots, x_{int})$,当前该群体的最优位置为 $X_{*t} = (x_{*1t}, \dots, x_{*nt})$ 。

若烟花*i*按如下方式爆炸:

$$X_i^{t+1} = X_i^t + \text{rand}(0, A_i) \quad (4)$$

式中, A_i 为第*i*个烟花的爆炸幅度范围,即爆炸的火花将在这个范围内随机产生位移,但不能超出这个范围,根据式(2)可以算出; rand 是 $(0, 1)$ 均匀随机分布。则称烟花*i*在*t*+1时刻采用非机动爆炸方式。

若烟花*i*按如下方式爆炸:

$$X_{ik}^{t+1} = X_{*k}^t, k = 1, 2, \dots, n \quad (5)$$

式中, k 为集合 $\{1, \dots, n\}$ 中的随机数。即烟花*i*的 k 维变轨绕到 $X_{*t} = (x_{*1t}, \dots, x_{*nt})$ 的第 k 维,向 X_{*} 靠近。则称烟花*i*在*t*+1时刻采用机动爆炸的方式。

每个烟花个体根据其当前的优劣位置来确定其爆炸方式。设当前烟花的个数为 N ,把当前烟花按照适应值的大小进行排序。适应值最好的烟花则是最优个体,序号为1,次优个体序号为2,以此类推。记烟花*i*在*t*时刻的序号为 $P_i(t)$,则 $P_i(t) = 1$ 表示烟花*i*在当前种群里面最优。

烟花*i*根据如下规则来确定其爆炸方式:

$$\begin{cases} X_i^{t+1} = X_i^t + \text{rand}(0, A_i) \frac{P_i(t)}{N} > \text{rand} \\ X_{ik}^{t+1} = X_{*k}^t \frac{P_i(t)}{N} \leq \text{rand} \end{cases} \quad (6)$$

式中, rand 是 $(0, 1)$ 均匀的随机分布。显然 $0 < P_i(t)/N \leq 1$ 。由此可以看出,处在较优位置的烟花倾向选择机动爆炸方式,而处于较劣位置的烟花则

倾向选择非机动的爆炸方式。

3.2.3 变异算子

为了保证种群的多样性,在烟花算法中引入了高斯变异,高斯变异的过程如下,首先在烟花种群随机选择一个烟花,对于选择得到的烟花再随机选择一定数量的维度进行高斯变异。高斯变异在选中的烟花和最好的烟花之间进行变异,产生新的火花。

3.2.4 映射规则

初始化随机的烟花可能靠近可行域的边界,由于爆炸幅度范围的原因,这些烟花爆炸产生的新的烟花可能覆盖到边界以外的区域。这种烟花对于该算法无用的,因此需要通过一种规则将其拉回可行域范围内,其公式为

$$x_i^k = x_{\min}^k + U(0, 1) \cdot (x_{\max}^k - x_{\min}^k) \quad (7)$$

式中, x_i^k 表示超出边界的第*i*个个体在第 k 维上的位置; $x_{\max}^k$ 和 $x_{\min}^k$ 分别表示第 k 维上的边界上下界; $U(0, 1)$ 表示在 $[0, 1]$ 区间上的均匀分布随机数。

3.2.5 选择策略

运用爆炸算子和变异算子并保证产生的火花在可行域范围之后,需要从产生的火花中选择出一部分作为下一代的烟花。烟花算法用到的是精英-随机选择策略。为了选择进入下一代的个体,选择策略采用的方式是每次都留下最优个体,再随机选择其他的 $N-1$ 个个体,即从烟花、爆炸烟花、高斯变异烟花组成的集合中随机选择。

3.3 算法基本步骤及流程

通过以上分析,RLMEFWA的基本步骤如下:

(1)运用反向学习策略初始化烟花种群,设置相关参数;

(2)对当前烟花种群进行排序,找出当前种群最优的个体,记下其位置信息;

(3)通过式(7)生成火花;

(4)通过变异算子生成火花;

(5)对超出边界的火花应用映射规则;

(6)应用选择策略得到下一代烟花群体;

(7)若满足停止条件(一般是达到一定精度或者是达到一定的迭代次数),则搜索停止,输出结果,否则,转到(2)。

RLMEFWA算法的流程图如图1所示。

3.4 算法收敛性证明

根据文献[7]可知基本的FWA算法是收敛的,基于此给出RLMEFWA收敛性证明过程如下。

假设待解决问题的全局最优是 X_{best} ,当前种群

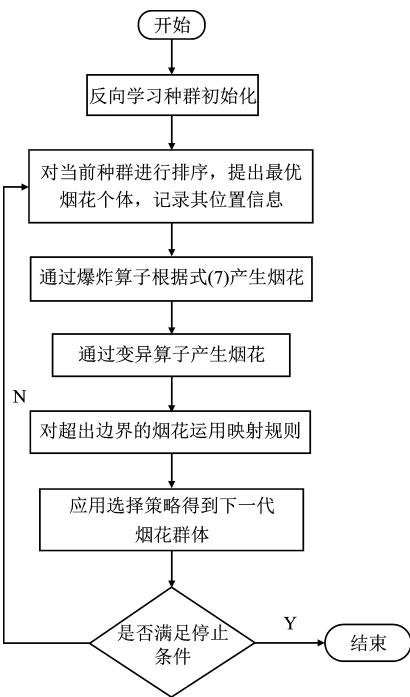


图1 RLMEFWA 流程图

中的最佳解 $X(t)$, 则如下极限等式成立:

$$\lim_{t \rightarrow \infty} X(t) = X_{\text{best}}.$$

RLMEFWA 算法中每个处于优秀位置的烟花个体以当前烟花种群的最优烟花为基准向其靠拢, 故如下极限等式成立: $\lim_{m \rightarrow \infty} X(m) = X(t)$, 因为 $X(t)$ 收敛于 X_{best} , 所以 $X(m)$ 必收敛于 X_{best} .

可得出结束, 如果 FWA 收敛于 X_{best} , 则 RLMEFWA 算法是收敛的.

显然, RLMEFWA 算法的时间复杂度主要体现在步骤 2~6, 如果以迭代次数作为算法的终止条件, 设最大的迭代次数为 M , 解空间的维数为 n , 种群的规模为 m , 算法获取最优个体的过程中一般是先执行排序操作, 则 RLMEFWA 算法的时间复杂度为 $O(M * n) + O(m \log m)$.

4 仿真实验

为了比较分析 RLMEFWA 的算法性能, 本文将 RLMEFWA 算法、FWA 算法、EFWA 算法以及 SPSO 算法^[9]用于 10 个典型的基准测试函数中, 并对实验结果进行比对, 可以直观的看到四种算法的性能.

4.1 实验设计

四种算法的参数设置如表 1 所示.

本实验采用的 10 个典型的基准测试函数均来自全局优化测试函数库. 其函数的基本特征如表 2

表 1 参数设置

算法	参数设计
SPSO	粒子群大小为 40, 权重设置为 0.729 84, c1 为 2.05, c2 为 2.05
FWA	初始烟花个数为 5, 烟花总数 50, 高斯变异个数为 5, 爆炸幅度为 40
EFWA	初始烟花个数为 5, 烟花总数 50, 高斯变异个数为 5, 爆炸幅度为 40, 初始爆炸半径系数为 0.02, 最终爆炸半径系数为 0.001
RLMEFWA	初始烟花个数为 5, 烟花总数 50, 高斯变异个数为 5, 爆炸幅度为 40

所示. 为了使图表更加清晰, 用 $f_1(x)$ 表示 Six-Hump Camel-back 函数, $f_2(x)$ 表示 Axis parallel hyper ellipsoid 函数, $f_3(x)$ 表示 Rotated hyper ellipsoid 函数. * 表示该函数有不止一个最优点.

表 2 基准测试函数基本特征

函数名	维数	搜索范围	最优点	最优值
Sphere	30	$[-100, 100]$	$(0, \dots, 0)$	0
Schwefel	30	$[-100, 100]$	$(0, \dots, 0)$	0
Rosenbrock	30	$[-30, 30]$	$(1, \dots, 1)$	0
Ackley	30	$[-32, 32]$	$(0, \dots, 0)$	0
Griewank	30	$[-600, 600]$	$(0, \dots, 0)$	0
Rastrigin	30	$[-5.12, 5.12]$	$(0, \dots, 0)$	0
$f_1(x)$	2	$[-5, 5]$	*	-1.031 628 5
Goldstein-Price	2	$[-2, 2]$	$(0, -1)$	3
$f_2(x)$	30	$[-5.12, 5.12]$	$(0, \dots, 0)$	0
$f_3(x)$	30	$[-65.536, 65.536]$	$(0, \dots, 0)$	0

由于在测试函数中, 许多函数的最优点在原点位置附近, 而基本烟花算法存在这一个特点, 即具有非常强大的原点位置附近的搜索能力. 因此, 实验在该测试集合上面增加了位置偏移. 位置偏移大小与优化问题的搜索范围有关, 如表 3 所示. 其中 Max、Min 为优化问题的搜索范围, SI 表示偏移索引, SV 表示偏移值大小.

表 3 基准测试函数的位置偏移描述

SI	SV	SI	SV	SI	SV
0	0	1	$0.05 \times \frac{\text{Max}-\text{Min}}{2}$	2	$0.1 \times \frac{\text{Max}-\text{Min}}{2}$
3	$0.2 \times \frac{\text{Max}-\text{Min}}{2}$	4	$0.3 \times \frac{\text{Max}-\text{Min}}{2}$	5	$0.5 \times \frac{\text{Max}-\text{Min}}{2}$
6	$0.7 \times \frac{\text{Max}-\text{Min}}{2}$				

为了评估算法的收敛性能, 要求最大的函数评估次数设置为 300 000, 连续运行 20 次, 分别进行了三次实验.

4.2 实验结果分析

在测试函数集合上进行性能测试, 对于不同算法

之间的性能比较主要有比较均值综合的排名. 为了验证不同算法在测试函数的差异显著性,通常使用 t 检验^[1]. 对此,本文分别进行了三次实验去比较 SPSO、FWA、EFWA 和 RLMEFWA 之间的性能差异.

4. 2. 1 SPSO、FWA、EFWA 和 RLMEFWA 的位置偏移曲线的对比

设定相同的迭代次数,对以上 10 个基准函数进行仿真实验,如图 2~11 所示. 由于有的坐标太小,图会产生重合的现象,因此在每个图下面附上了函数最优值的表格.

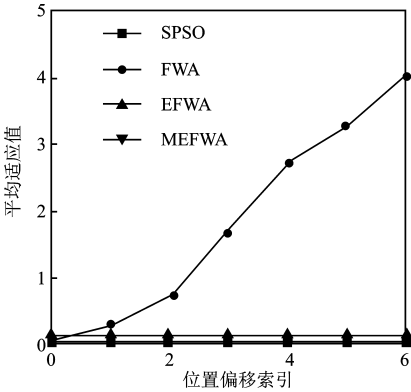


图 2 Sphere 函数对比

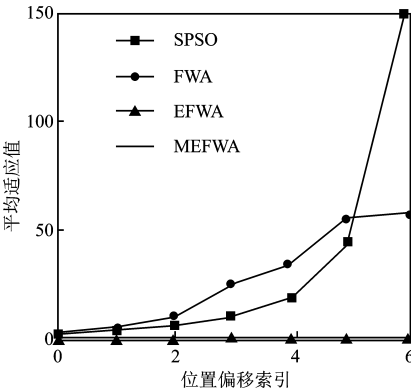


图 3 Schwefel 函数曲线对比

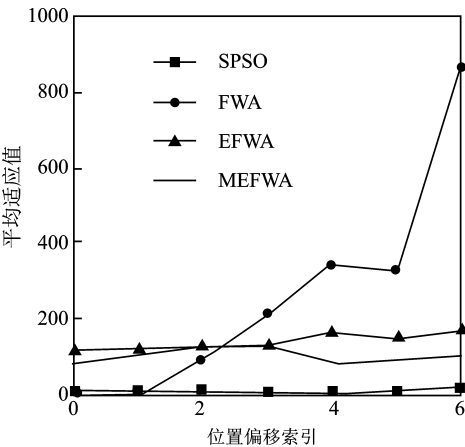


图 4 Rosenbrock 函数对比

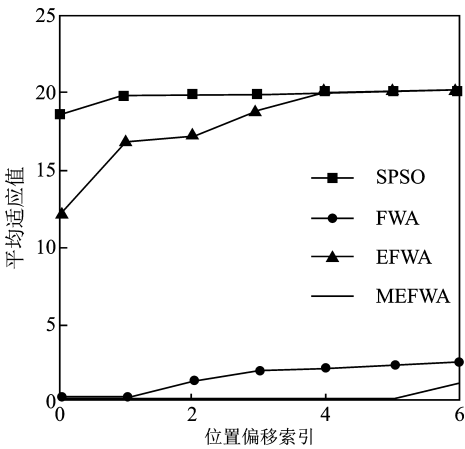


图 5 Ackley 函数对比

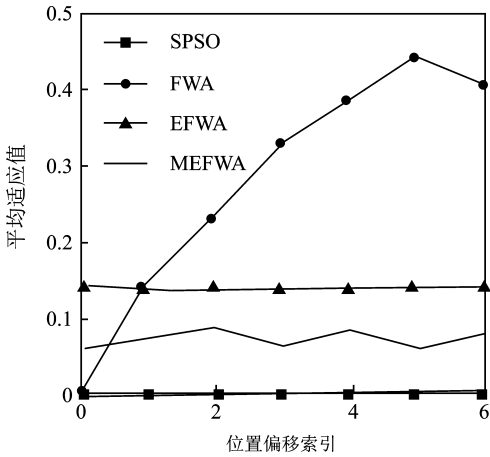


图 6 Griewank 函数对比

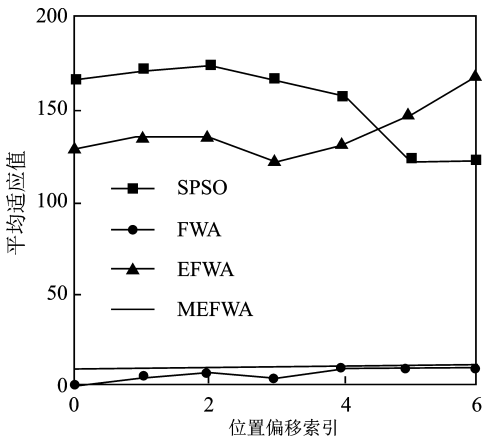


图 7 Rastrigin 函数对比

由图 2 和表 4 可知,随着函数的位置偏移的变大,RLMEFWA 相对于 FWA、EFWA 取得了更加明显的性能优势,但是 SPSO 比 RLMEFWA 性能好.

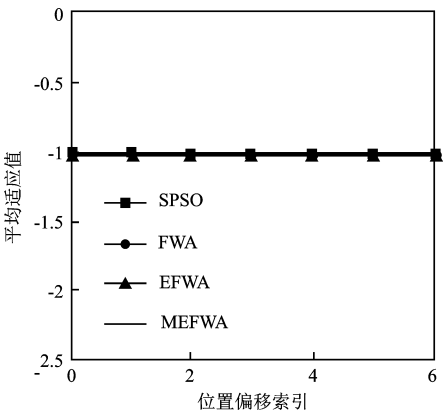


图 8 $f_1(x)$ 函数对比

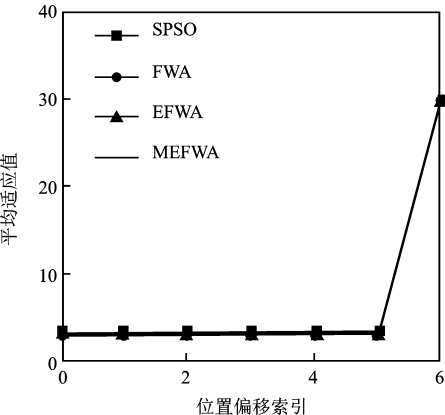


图 9 Goldstein-Price 函数对比

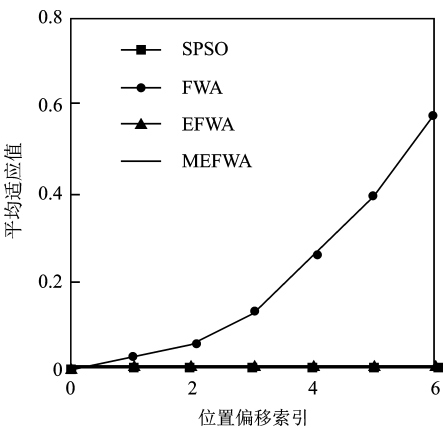


图 10 $f_2(x)$ 函数对比

表 4 Sphere 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	0	0	0	0	0	0	0
FWA	0	0.234	0.709	1.666	2.676	3.255	4.025
EFWA	0.072	0.066	0.068	0.069	0.064	0.072	0.076
RLMEFWA	0.005	0.005	0.004	0.003	0.004	0.004	0.004

由图 3 和表 5 可知,随着函数的位置偏移的不断变大,SPSO 和 FWA 相对于 EFWA 和 RLMEF-

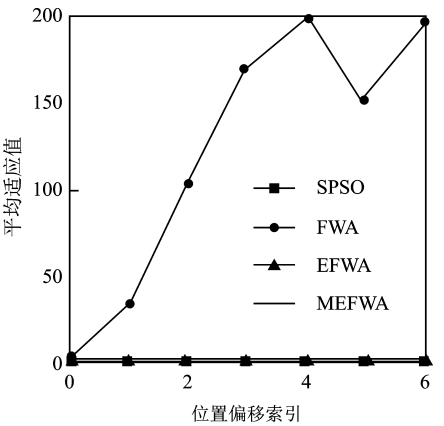


图 11 $f_3(x)$ 函数对比

WA 有很差的性能,且全局最优值也在不断变大,稳定性差. RLMEFWA 和 EFWA 都具有稳定性,但是 RLMEFWA 比 EFWA 具有更好的收敛性.

表 5 Schwefe 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	3.721	3.798	5.619	9.738	18.20	42.15	148.7
FWA	0	3.745	10.12	24.61	33.87	53.28	57.39
EFWA	0.177	0.185	0.191	0.185	0.187	0.188	0.169
RLMEFWA	0.055	0.065	0.059	0.055	0.054	0.046	0.060

由图 4 和表 6 可知,随着函数的位置偏移的变大,四种算法的全局最优值都在不断的变大,但是 SPSO 比其他三种算法具有更好的性能,RLMEFWA 相对于 EFWA 和 FWA 具有更好的收敛性和稳定性.

表 6 Rosenbrock 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	9.314	11.127	12.25	10.72	12.28	13.86	16.565
FWA	25.55	3.847 5	88.02	210.1	335.2	323.3	851.63
EFWA	120.2	110.71	119.4	124.3	158.2	138.2	165.99
RLMEFWA	83.59	91.735	116.6	120.0	84.68	85.28	111.04

由图 5 和表 7 可知,随着函数的位置偏移的变大,RLMEFWA 相对于 SPSO、EFWA 和 FWA 具有更好的收敛性和稳定性.

表 7 Ackley 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	18.53	19.756	19.52	19.93	19.98	20.00	20.114
FWA	0	0.243 7	0.987	1.737	2.033	2.358	2.418 2
EFWA	11.86	16.734	17.20	18.92	19.96	19.98	19.997
RLMEFWA	0.018	0.017 9	0.021	0.019	0.018	0.019	1.1477

由图 6 和表 8 可知,随着函数的位置偏移的变大,RLMEFWA 相对于 EFWA 和 FWA 具有更好的收敛性和稳定性. SPSO 相对于 RLMEFWA 具有更好的性能优势.

表 8 Griewank 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	0.000	0.000 8	0.001	0.001	0.001	0.002	0.006 1
FWA	0	0.140 6	0.228	0.327	0.381	0.438	0.405 7
EFWA	0.139	0.131	0.136	0.132	0.134	0.137	0.136 5
RLMEFWA	0.057	0.070 7	0.086	0.060	0.076	0.060	0.075 1

由图 7 和表 9 可知,随着函数的位置偏移的变大,FWA 相对于其他三种算法具有更好的收敛性. 但是位移偏大,RLMEFWA 的稳定性和收敛性优于 FWA,且在收敛性和稳定性上比 EFWA 和 SPSO 具有一定的优势.

表 9 Rastrigin 函数全局最优平均值对比
(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	165.1	170.23	172.6	165.4	155.6	122.1	120.59
FWA	0	3.444 6	6.764	4.441	8.890	9.059	9.691 1
EFWA	128.1	133.85	135.2	122.1	128.9	145.2	166.52
RLMEFWA	8.821	8.023 2	8.820	8.619	8.315	7.919	7.573 5

由图 8 和表 10 可知,随着函数的位置偏移的变大,RLMEFWA 和其他三种算法收敛性和稳定性都一样.

表 10 $f_1(x)$ 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03
FWA	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03
EFWA	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03
RLMEFWA	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03	-1.03

由图 9 和表 11 可知,随着函数的位置偏移的变大,RLMEFWA 和 SPSO 以及 EFWA 算法收敛性和稳定性都一样. 比 FWA 性能好.

表 11 Goldstein-Price 函数全局最优平均值对比
(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	3	3	3	3	3	3	30
FWA	3.001	3.001	3.001	3.001	3.002	3.000 1	30.00 1
EFWA	3	3	3	3	3	3	30
RLMEFWA	3	3	3	3	3	3	30

由图 10 和表 12 可知,随着函数的位置偏移的变大,RLMEFWA 相对于 FWA、EFWA 取得了更加明显的性能优势,但是没有 SPSO 性能好.

表 12 $f_2(x)$ 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	0	0	0	0	0	0	0
FWA	0	0.017	0.053 1	0.125	0.261	0.392	0.57
EFWA	0.003 1	0.003 1	0.003 1	0.003	0.003	0.003	0.002
RLMEFWA	1.48e-04	1.42e-04	1.56e-04	1.46e-04	1.55e-04	1.57e-04	1.53e-04

由图 11 和表 13 可知,随着函数的位置偏移的变大,RLMEFWA 相对于 FWA 取得了更加明显的性能优势,SPSO、EFWA 相对于 RLMEFWA 具有更好的收敛性和稳定性.

表 13 $f_3(x)$ 函数全局最优平均值对比(SI 为偏移索引)

算法	SI=0	SI=1	SI=2	SI=3	SI=4	SI=5	SI=6
SPSO	0	0	0	0	0	0	0
FWA	0	32.25	103.02	167.7	195.91	148.6	197.4
EFWA	0.412	0.474	0.482 4	0.482	0.417 1	0.471	0.482
RLMEFWA	0.581	0.573	0.492 4	0.489	0.553 1	0.530	0.516

4.2.2 SPSO、FWA、EFWA 和 RLMEFWA 在测试函数上的平均排名

为了研究 RLMEFWA 算法的性能,在同等实验的条件下,设置最大函数评估次数为 30 000,连续运行 20 次,取这 20 次得出的全局最优值的综合平均值的排名作对比. 结果如表 14 所示.

表 14 SPSO、FWA、EFWA 和 RLMEFWA 在测试函数上的平均排名

函数名	SPSO	FWA	EFWA	RLMEFWA
Sphere	1	4	3	2
Schwefel	4	3	2	1
Rosenbrock	1	3	4	2
Ackley	4	2	3	1
Griewank	1	4	3	2
Rastrigin	3	2	4	1
$f_1(x)$	1	1	1	1
Goldstein-Price	1	2	1	1
$f_2(x)$	1	4	3	2
$f_3(x)$	1	4	2	3
平均排名	1.8	2.9	2.6	1.6

由表 14 的平均排名结果可知,RLMEFWA 在四种算法中排名第一,远远高于 EFWA 和 FWA 的排名,比 SPSO 排名也高. 即基于反向学习与机动爆炸烟花优化算法均优于 FWA、EFWA 和 SPSO 的搜索能力.

四种算法在 10 个基准函数独立运行 20 次,因而在每个测试函数上,每种算法都有 20 个样本. 本文采用 T 检验来比较分析四种算法性能. 该 T 检验

采用自由度为 20,显著水平为 0.05 的双尾检验. 其结果如表 15 所示.

表 15 RLMEFWA 算法和 FWA、EFWA、SPSO 的 T 检验

函数名	FWA	EFWA	SPSO
Sphere	+	+	—
Schwefel	+	+	+
Rosenbrock	+	+	—
Ackley	+	+	+
Griewank	+	+	—
Rastrigin	—	+	+
$f_1(x)$	=	=	=
Goldstein-Price	+	=	=
$f_2(x)$	+	+	—
$f_3(x)$	+	—	—

表 15 的检验结果中“+”表示 RLMEFWA 算法在相应的测试函数上效果优于该列算法,“=”表示二种算法在该行的测试函数上的效果一致,“—”表示 RLMEFWA 算法在这行的测试函数的效果不如该列算法.

从表 15 可以看出,RLMEFWA 相对于 FWA,除了在函数 Rastrigin 上,RLMEFWA 效果没有 FWA 好,在函数 Six-Hump Camel-back 上,二种算法效果一样,在其他函数上 RLMEFWA 的效果均优于 FWA;RLMEFWA 相对于 EFWA,在函数 Six-Hump Camel-back 和 Goldstein-Price 上,二个算法效果一样,在函数 Rotated hyper ellipsoid 上,RLMEFWA 效果没有 EFWA 好除此之外,在其他函数上,RLMEFWA 的效果均优于 EFWA 算法;RLMEFWA 相对于 SPSO,二种算法在 10 个测试函数上,效果有好有坏,再结合表 14 的排名情况,RLMEFWA 在总体上的效果和稳定性比 SPSO 好. 所以从统计的角度可得出结论:RLMEFWA 算法在大部分测试问题上均优于 FWA 算法、EFWA 算法和 SPSO 算法.

5 结束语

本文提出一种采用反向学习与机动爆炸的方式的烟花算法(RLMEFWA). 该算法初始化种群采用反向学习,增加了种群初始化的多样性,,然后每个烟花个体根据其自身的优劣位置来选择不同的爆炸方式,让处于优秀的位置的烟花像种群最优烟花进行靠拢,加快了收敛速度. 通过对 10 个基准测试函

数的仿真实验表明,RLMEFWA 算法收敛性能和计算精度在不同程度上优于 FWA 算法、EFWA 算法和 SPSO 算法. 且随着位移位置的不断增大,RLMEFWA 相对于其他三种算法具有很好的稳定性.

参考文献:

[1] 谭莹,郑少秋. 烟花算法研究进展[J]. 智能系统学报, 2014, 9(5):515-528.

[2] Zheng S, Janecek A, Tan Y. Enhanced fireworks algorithm[C]// Evolutionary Computation (CEC), 2013 IEEE Congress on. IEEE,2013.

[3] PEI Y, ZHENG S, TAN Y, et al. An empirical study on influence of approximation[C]//IEEE International Conference on Systems, Man, and Cybernetics. Seoul, South Korea, 2012:1322-1327.

[4] LIU J, ZHENG S, TAN Y. The improvement on controlling exploration and exploitation of fireworks algorithm[M] // Advances in Swarm Intelligence. Berlin: Springer, 2013: 11-23.

[5] ZHENG S, JANECEK A, LI J, et al. Dynamic search in fireworks algorithm[C] // 2014 IEEE Congress on Evolutionary Computation. Beijing, China, 2014: 3222-3229.

[6] LI J, ZHENG S, TAN Y. Adaptive Fireworks Algorithm[C]//2014 IEEE Congress on Evolutionary Computation. Beijing, China, 2014: 3214-3221.

[7] 谭莹. 烟花算法引论[M]. 北京:科学出版社, 2015.

[8] Tizhoosh H. Opposition-based learn: A new scheme for machine intelligence[C]//Proceedings of the International Conference on Computational Intelligence for Modeling Control and Automation. Vienna, Austria: IEEE, 2005:695-701.

[9] Bratton D, Kennedy J. Defining a standard for particle swarm optimization[C]//Swarm Intelligence Symposium, 2007. IEEE,2007.

作者简介:

李席广 男,(1979-),硕士,讲师. 研究方向为网络安全、云安全和信任管理.

韩守飞(通讯作者) 男,(1990-),硕士. 研究方向为智能算法优化和云计算安全. E-mail:1492519973@qq.com.

拱长青 男,(1965-),博士,教授. 研究方向为云计算安全和网络通信.