The Improvement on Controlling Exploration and Exploitation of Firework Algorithm

Jianhua Liu[1,2], Shaoqiu Zheng[2], and Ying Tan[2,⋆]

[1] School of Information Science and Engineering, Fujian University of Technology, Fuzhou, 350108, P.R. China

[2] Department of Machine Intelligence, School of EECS, Peking University
Key Laboratory of Machine Perception (MOE), Peking University
jhliu@fjnu.edu.cn, {zhengshaoqiu,ytan}@pku.edu.cn

Abstract. Firework algorithm (FWA) is a new Swarm Intelligence (SI) based optimization technique, which presents a different search manner and simulates the explosion of fireworks to search the optimal solution of problem. Since it was proposed, fireworks algorithm has shown its significance and superiority in dealing with the optimization problems. However, the calculation of number of explosion spark and amplitude of firework explosion of FWA should dynamically control the exploration and exploitation of searching space with iteration. The mutation operator of FWA needs to generate the search diversity. This paper provides a kind of new method to calculate the number of explosion spark and amplitude of firework explosion. By designing a transfer function, the rank number of firework is mapped to scale of the calculation of scope and spark number of firework explosion. A parameter is used to dynamically control the exploration and exploitation of FWA with iteration going on. In addition, this paper uses a new random mutation operator to control the diversity of FWA search. The modified FWA have improved the performance of original FWA. By experiment conducted by the standard benchmark functions, the performance of improved FWA can match with that of particle swarm optimization (PSO).

Keywords: Firework Algorithm, Swarm Intelligence Algorithm, Exploration and Exploitation, PSO.

1 Introduction

Firework algorithm (FWA) is a new intelligence optimization algorithm based on Swarm Intelligence (SI) developed by Y. Tan and Y. Zhu [14]. Like the other SI algorithms, such as Particle Swarm Optimization (PSO) [8], Ant System [3], Colonel Selection Algorithm [2, 13], and Swarm Robots [1], Different Evolution (DE) [11], Artificial Bee Colony (ABC) [7] etc., firework algorithm also is a population based optimization technique. Firework algorithm simulates the explosion of fireworks to search the optimal solution of problem. Compared to the other SI algorithms, firework algorithm has distinctive advantages in solving many optimization problems and presents a different search manner.

⋆ Senior Member, IEEE

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part I, LNCS 7928, pp. 11–23, 2013.

Algorithm 1. Conventional Fireworks Algorithm.

1: Select n position for initial fireworks;
2: Calculate the number of sparks for each firework;
3: Calculate the amplitude of explosion for each firework;
4: Generate the sparks of explosion for each firework;
5: Generate the sparks of Gaussian mutation for each firework;
6: Select n location for next generation fireworks
7: If condition does not meet, algorithm turns to 2
8: Output results

Since it was proposed, fireworks algorithm has shown its significance and superiority in dealing with the optimization problems and has been seen many improvement and application with practical optimization problems. Andreas and Tan [5, 6] used FWA to compute the non-negative matrix factorization and gains a little advantages compared to SPSO, FSS, GA. Pie et al. [9] investigated the influences of approximation approach on accelerating the fireworks algorithm search by elite strategy. In [9], they compared the approximation models, sampling methods, and sampling number on the FWA acceleration performance, and the random sampling method with two-degree polynomial model gains better performance on the benchmark functions. Zheng et al. [16] proposed a hybrid algorithm between FWA and differential evolution (DE), which shows superiority to the previous FWA and DE. H. Gao and M. Diao [4] have designed a cultural firework algorithm which was used to search optimal value of filter design parameters with parallel search. Computer simulations have showed that FIR and IIR digital filters based on the cultural firework algorithm are superior to previous filters based on the other SI algorithm in the convergence speed and optimization results.

However, firework algorithm needs a policy that dynamically controls the exploration and exploitation of searching solution of problem with iteration. In the FWA, the calculation of number of explosion spark and amplitude of firework explosion have too more parameter to set. Its mutation operator cannot availably control the diversity of FWA. This paper will design a transfer function to map the rank number of firework to percentage value in order to calculate the number of explosion spark and firework explosion, and then a rand mutation is presented to generate the diversity of FWA. By the experitment, the modified techniques on FWA can improve the perfromance of original FWA. The paper is organized as follows: section 2 introduces the FWA and analyzes the drawback of FWA; the improved FWA was provided in section 3. Section 4 conducted the experiment to test the effect of improved FWA. The section 5 is the conclusion of this paper.

2 Fireworks Algorithm

Firework algorithm searches the optimal solution of problem using several fireworks explosion to generate the sparks in the space of problem, where the spark and firework are the potential solutions of problem. The procedure of original firework algorithm can be seen in Algorithm 1.

According to the idea of FWA, a good firework denotes the better fitness,which means that the firework may be close to the optimal location. Therefore, the good firework should generate more sparks in the smaller amplitude of explosion. In the contrast, for a bad firework, the search radius should be larger and it should generate less sparks in the larger amplitude of explosion. For every firework, the number of explosion sparks and amplitude of explosion must be calculated before it explodes, which formulas are as follows:

– Calculating the Number of Sparks:

$$S_i = m\frac{y_{\max} - f(x_i) + \xi}{\sum\limits_{i=1}^{n} (y_{\max} - f(x_i)) + \xi} \tag{1}$$

$$\widehat{s_i} = \begin{cases} round(a \cdot m) \;\; if\, s_i < am \\ round(b \cdot m) \;\; if\, s_i > bm \\ round(s_i) \;\; otherwise \end{cases} \tag{2}$$

Where S_i is the number of the spark of the ith firework explosion, m is the total number of sparks generated by the n fireworks. y_{max} is the maximum value of the objective function among the n fireworks, and is a small constant which is utilized to avoid zero-division-error. The constant a and b are the const parameters.

– Calculating the Amplitude of Explosion:

$$A_i = \tilde{A}\frac{f(x_i) - y_{\min} + \xi}{\sum\limits_{i=1}^{n} (f(x_i) - y_{\min}) + \xi} \tag{3}$$

Where $\tilde{A}$ denotes the maximum explosion amplitude and y_{min} is the minimum value of the objective function among n fireworks.

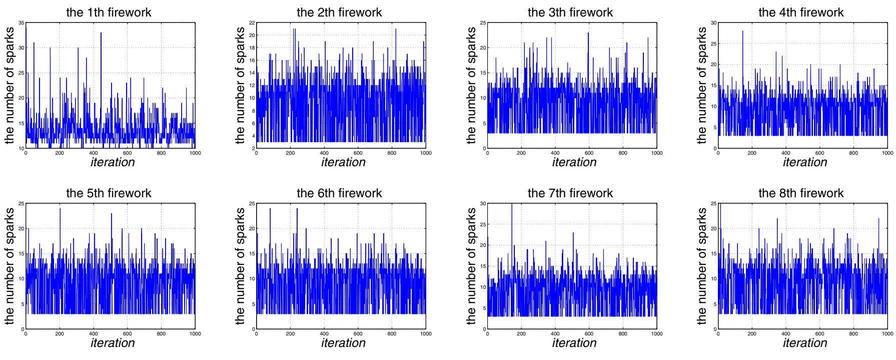

Fig. 1. The number of sparks for every firework in original FWA

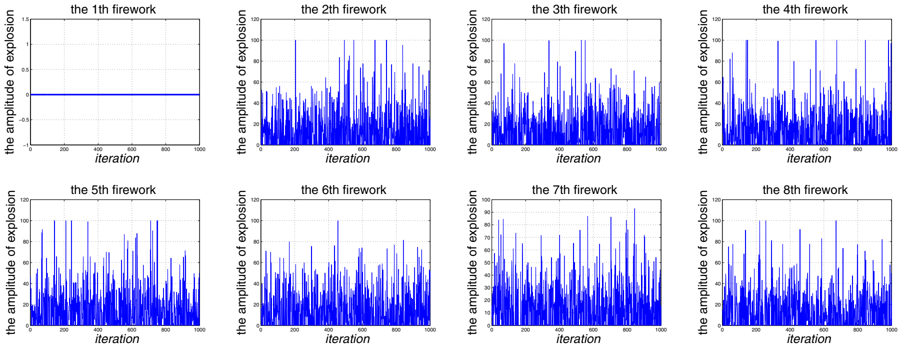

Fig. 2. The amplitude of explosion for every firework in the proposed algorithm

In order to investigate the effect which Eq.(1) and Eq.(3) impact on firework algorithm, we can observe the Fig.1 and Fig.2. Fig.1 is the number of sparks generated by eight fireworks with the FWA iteration, while Fig.2 is the amplitude of explosion generated by eight fireworks with the FWA iteration. The data in the two figures is gained with Eq.(1) and Eq.(3) provided that every iteration of FWA has only eight fireworks.

From Fig.1, different fireworks in the different iteration have no regularity to gain the number of sparks. The number of sparks generated by different firework all are vibrated almost between 5 and 15.The best firework (the first firework) has not always generated the most spark number among eight fireworks. From Fig.2, the first firework (the best firework) explodes in the smaller amplitude than the amplitude of other firework. The explosion amplitude of the first fireworks is always about 10^{-4}, while the amplitude of the other firework is about between 0 and 50. It can be found that the amplitude of first firework is too small in the early time and constant while the other firework' amplitude is variable and have not regularity.

In term of Fig.1 and Fig.2, the Eq.(1) and Eq.(3) can embody random of FWA, because the explosion number and explosion amplitude of fireworks is variable with the fitness of firework. However, spark number and amplitude of firework explosion have not dynamically changed as the algorithm iterates. Especially, the best firework (the 1th firework) has constant explosion amplitude and its sparks number of explosion does not increase with iteration. Therefore,it is difficult for the formula of Eq.(1) and Eq.(3) to effectively control the local exploration and global exploitation of FWA in the solution space. In the next section, we will provide two new equations to modify the Eq.(1) and Eq.(3).

3 The Improvement of FWA

3.1 The Improvement of Computing the Scope and Sparks Number of Firework Explosion

The above drawback of FWA is account of using the fitness of firework to compute the scope and sparks number of firework explosion. In order to improve the equations on calculating the number of firework sparks and the amplitude of firework explosion, we use the sequence number of fireworks to compute the two values. Therefore, a transfer function must be designed to map the sequence number of fireworks to function value which is used to better calculate the amplitude and spark number of firework explosion. The transfer function is cited from the sigmoid function as the Eq.(4):

$$f(x) = \frac{1}{1 + e^x} \tag{4}$$

The function is further improved and added a parameter a as the following Eq.(5):

$$f(x) = \frac{1}{1 + e^{(x-1)/a}} \tag{5}$$

Where a is a control parameter to change the shape of the above function. Eq.(5) can transfer sequence number of firework rank of fitness to different value of function, which is used to calculate the spark number and amplitude of firework explosion. The function of Eq.(5) is named as transfer function. When the parameter $a = 1, 5, 9, 13$ and 21,the figures of the transfer function with different parameter value are plotted as Fig.3 which x axis denotes sequence number. From the Fig.3, it can be found that the function fitness of different sequence number is more and more mean as the parameter a is increasing. So, the calculating number of explosion sparks is designed as the following equation:

$$S_n = m \frac{f(n)}{\sum\limits_{n=1}^{N} f(n)} \tag{6}$$

Where m is the total of number of spark, n is the sequence number of a firework. S_n denotes the spark number of the nth firework explosion. The calculating the amplitude of firework is designed as following equation:

$$A_n = A \frac{f(N-n+1)}{\sum\limits_{n=1}^{N} f(N-n+1)} \tag{7}$$

Where A is the maximum amplitude of firework explosion, n is the sequence number of a firework. A_n denotes the amplitude of the nth firework explosion. In the Eq.(6) and Eq.(7), N is the total number of firework in FWA. The function $f(x)$ is the Eq.(5), which parameter a is varied with the iteration from 20 to 1. With variable parameter a, the explosion number of spark and explosion amplitude of firework is dynamically changed as the iteration goes on.

In order to compare Eq.(1) and Eq.(3) to Eq.(6) and Eq.(7), Fig.4 and Fig.5 plot the number of sparks and amplitude of explosion of eight fireworks with iteration which are calculated using Eq.(6) and Eq.(7).

Compared to the Fig.1 and Fig.2, Fig.4 shows that the number of spark generated by the firework of FWA with the modified equation is very regular. For the sorting front fireworks, the number of sparks is more and more with the iteration, while for the last fireworks, the number of sparks is less and less. Fig.5 shows that the amplitude

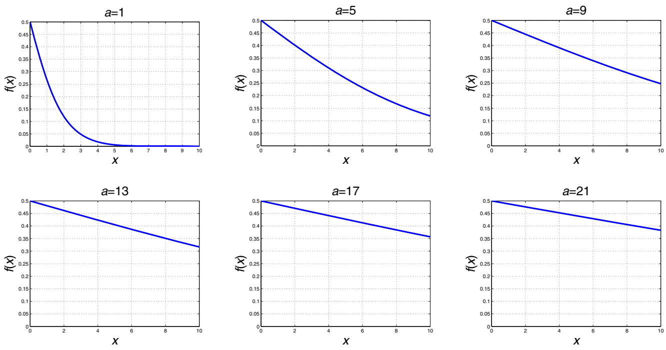

Fig. 3. The plotting figure of transfer function with different parameter

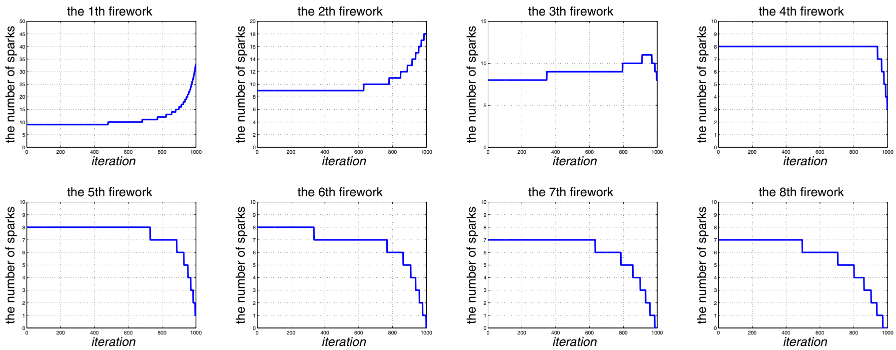

Fig. 4. The number of sparks for every firework in modifying FWA

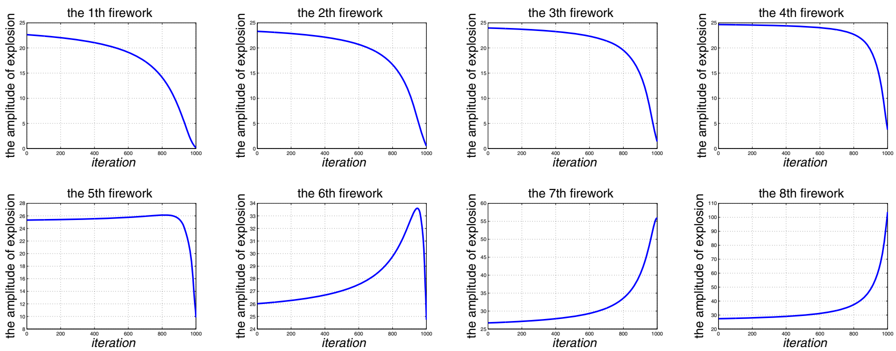

Fig. 5. The amplitude of explosion for every firework in modifying FWA

of the better firework explosion is smaller and smaller as the iteration goes on, while the amplitude of bad firework explosion is bigger and bigger with iteration. The policy can embody the idea of algorithm that a good firework has more number of generating sparks and less amplitude of explosion while a bad firework generate less number of sparks during the larger amplitude of explosion. There may be global optimal solution near the good firework, so the explosion of a good firework undertakes the local searching, while a bad firework exploding undertakes the global exploitation of solution space. In the new method, the dynamic change of number of sparks and amplitude of explosion with iteration can embody that the global exploitation is done in the early time of algorithm running, while the local exploration is enhanced during the later time of algorithm's iteration. So, the new calculating equations can better control the exploitation and exploration of firework algorithm with iteration.

3.2 The Mutation Improvement

To keep the diversity, original firework algorithm employed Gaussian mutation to generate sparks. The jth dimensions of the ith firework, x_{ij}, mutates as x_{ij} by the following equation:

$$x_{ij} = x_{ij} Gaussian(1, 1) \tag{8}$$

However, the above mutation makes original FWA easily converged to zero point of the search space, and it is difficult for FWA to generate the diversity. In order to add the diversity of FWA, the random mutation is employed to make the firework mutated. The mutation formula is as follows:

$$x_{ij} = x_{ij}^{min} + rand()(x_{ij}^{max} - x_{ij}^{min}) \tag{9}$$

Where x_{ij} denotes the position of the jth dimensions of ith firework; x_{ij}^{min} denotes the minimal bound of the jth dimensions of the ith firework; x_{ij}^{max} denotes maximal bound of the jth dimensions of the ith firework. The function $rand()$ gains the sampling value in the interval [0, 1] with the uniform distribution.

3.3 The Selection of the Next Generation Fireworks

Original FWA selects n location for next generation fireworks by the Eq.(10) and Eq.(11):

$$R(x_i) = \sum_{j \in K} d(x_i, x_j) = \sum_{j \in K} ||x_i - x_j|| \tag{10}$$

$$p(x_i) = \frac{R(x_i)}{\sum\limits_{j \in K} R(x_i)} \tag{11}$$

Where the x_i is the location of ith sparks or firework, $d(x_i, x_j)$ is the distance between two sparks or fireworks. K is the set of sparks and firework generated in current generation. The $p(x_i)$ is the probability which the ith firework or spark is selected as the firework of next generation. Eq.(10) and Eq.(11) do not consider the fitness of sparks or fireworks for the selection of next generation fireworks' location. This is not consist with the idea of the equations of Eq.(6) and Eq.(7). Because Eq.(6) and Eq.(7) use the sequence number of sorting fireworks' fitness to calculate the sparks number and amplitude of fireworks explosion, but the Eq.(10) and Eq.(11) of original FWA don't consider the fitness to select the firework location. Therefore, there are two methods to be provided to modify the selection operator.

1. Fitness Selection using the roulette
   Like the original FWA, the best of the set will be selected first. Then the others are selected base on fitness proportion using the roulette. So, the selection probability of every spark or firework must be calculated with the following formula:

$$p(x_i) = \frac{y_{\max} - f(x_i)}{\sum\limits_{i \in K} (y_{\max} - f(x_i))} \tag{12}$$

   Where y_{max} is the maximum value of the objective function among the set K which consist of the fireworks and sparks in the current generation. The other fireworks will be selected using the roulette according to the probability gained by Eq.(12).
2. Best Fitness Selection
   In [15], Zheng et al used a random selection operator to replace the previous time consuming one. It is as the following, when the algorithm has decided the number of firework of every generation, all the sparks and fireworks of the current generation are sorted according to their fitness and then select the best n sparks or fireworks with the best fitness as the location of next generation. The method is very simple and is consistent with the new calculation of explosion number and explosion amplitude of fireworks in the Eq.(6) and Eq.(7).

4 Experiment and Analysis

4.1 Experimental Design

In order to evaluate the performance of the improved FWA, fourteen benchmark functions provided by CEC 2005 are employed [12]. These benchmark functions include five unimodal functions, nine multimodal functions. The optimal fitness of these functions is not zero and is added bias. These functions are shifted and the optimal locations are shifted to different location from zero point in solution space. More details on the benchmark functions can be seen in [12].

In order to test the performance of improved FWA in this paper, the improved FWA with best fitness selection and random mutation (IFWABS) , the improved FWA with the fitness selection using the roulette and random mutation (IFWAFS), original FWA and global PSO are compared with each other. The global PSO is employed the decreasing weight w from 0.9 to 0.4 proposed in [10], and the neighbor particles of each particle is all particles. The particle population size is 100. The factor c_1 and c_2 of PSO are set as 2. The FWA and improved FWA are set the number of firework as 10, the total number of explosion sparks S as 80 and the amplitude of explosion A as the range length of problem space. The experiment is conducted in Matlab 2012b and executed in windows 7.

4.2 Experimental Results and Analysis

The experiment is conducted to compute the mean error fitness ($f(x) - f(x^*)$, $f(x^*)$ is real optimal fitness of the benchmark functions), standard square error and the best error fitness in the 25 run on the 14 benchmark functions. Each run of all algorithm is evaluated 1000, 10000, 10000 and $D * 10000$(D is the dimension of benchmark function), respectively. Each algorithm will be conducted in the 10 dimensions and 30 dimensions, respectively. Table1 is the results of mean error fitness, standard square error and the best fitness in the 25 run on 14 functions in 10 dimensions, which the Fitness Evaluated number(FEs) is 1000, 10000,100000. Table2 is the results of that in 30 dimensions, which the FEs is 10000, 100000 and 300000.

In term of Table 1, compared to FWA, the performance of IFWABS and IFWAFS is better than that of FWA. In the 10 dimensions, the mean error finesses of FWA on 14 functions all are worse than that of IFWABS or IFWAFS whether FEs is 1000, 10000 or 100000. In term of Table2, the performance of IFWABS and IFWAFS is better than that of FWA on all functions except for the 8th function in 30 dimensions. Therefore, the improved FWA has improved the performance of firework algorithm. Compared the performance of two improved FWA (IFWABS and IFWAFS), it can be found from Table1 and Table2 that IFWABS is advantage to IFWAFS. In more part of cases, the mean error fitness of function of IFWABS is outstanding to the IFWAFS. Compared to PSO, improved FWA is more optimal performance on most of functions, especially in 100000 FEs and 300000 FEs. As the FEs is more and more, the performance of improved FWA is better and better than PSO, so the improved FWA can match with PSO. Fig.6 plots the convergence process for four algorithms to optimize the 14 functions with 300000 FEs in 30 dimensions These figures are visual to illustrate the effect of four algorithms that improved FWA is excel to original FWA and can match with PSO.

5 Conclusion

Firework algorithm is a novel swarm intelligence based algorithm that can availably search the optimal solution of parameter space. FWA imitates the firework explosion to generate sparks and provide the idea that better firework can take up the local search and the bad firework do the global exploitation. This paper modifies the calculation of scope and amplitude of firework explosion, and designs a transfer function to map rank number of firework fitness to allocate the total sparks number and explosion aptitude. A parameter of transfer function was used to control the dynamical calculation of two values with iteration. This way is more effective to control the local and global search of FWA in solution space. In addition, a random mutation was presented to enhance the diversity of FWA. At last, in order to accord with the new idea of calculation of scope and amplitude of firework explosion, the best sparks selection and fitness selection was employed to improve the selection operator of FWA. By experiment conducted on 14 standard benchmark functions in CEC 2005, the improved FWA can be superior to the original FWA and can match with PSO.

Table 1. Statistical Results of Mean, Std and Best of Benchmark Functions in 10 Dimension

FES	No.		PSO	FWA	IFWAFS	IFWABS	FES	No.		PSO	FWA	IFWAFS	IFWABS
		Mean	3.32E+03	1.86E+04	4.33E+03	**3.19E+03**			Mean	2.07E+01	2.06E+01	2.06E+01	**2.06E+01**
	f_1	Std	1.45E+03	3.88E+03	1.67E+03	1.43E+03		f_8	Std	1.55E-01	1.49E-01	1.18E-01	1.16E-01
		Best	5.17E+02	1.32E+04	1.12E+03	6.31E+02			Best	2.03E+01	2.02E+01	2.04E+01	2.04E+01
		Mean	1.21E+04	3.60E+04	1.23E+04	**1.20E+04**			Mean	5.17E+01	1.21E+02	6.24E+01	**5.01E+01**
	f_2	Std	4.25E+03	1.08E+04	6.24E+03	4.51E+03		f_9	Std	1.27E+01	2.63E+01	1.88E+01	1.37E+01
		Best	4.90E+03	1.38E+04	1.32E+03	3.83E+03			Best	2.89E+01	5.54E+01	2.0E+01	2.05E+01
		Mean	**4.38E+07**	9.56E+08	7.67E+07	5.07E+07			Mean	8.0E+01	1.96E+02	9.81E+01	**8.66E+01**
	f_3	Std	3.11E+07	6.74E+08	4.73E+07	4.52E+07		f_{10}	Std	1.68E+01	3.46E+01	2.0E+01	2.11E+01
		Best	6.85E+06	2.28E+08	1.24E+07	1.42E+06			Best	4.41E+01	1.23E+02	5.82E+01	4.32E+01
		Mean	**1.39E+04**	3.19E+04	1.49E+04	1.51E+04			Mean	**9.0E+0**	1.25E+01	1.06E+01	9.58E+0
1.0E+03	f_4	Std	6.63E+03	1.24E+04	5.41E+03	5.99E+03	1.0E+03	f_{11}	Std	1.29E+0	1.03E+0	1.25E+0	1.09E+0
		Best	3.40E+03	5.74E+03	5.63E+03	3.65E+03			Best	5.55E+0	9.54E+0	8.52E+0	6.90E+0
		Mean	**1.09E+04**	2.33E+04	1.26E+04	1.12E+04			Mean	3.67E+04	2.84E+05	4.58709e+04	**3.50379e+04**
	f_5	Std	2.62E+03	3.48E+03	3.06E+03	2.66E+03		f_{12}	Std	1.79E+04	8.58E+04	2.63E+04	2.36E+04
		Best	5.90E+03	1.64E+04	5.43E+03	5.94E+03			Best	1.18E+04	1.04E+05	6.23E+03	7.83E+03
		Mean	**3.10E+08**	1.96E+10	1.17E+09	6.70E+08			Mean	7.81E+0	2.73E+02	1.06E+01	**7.76E+0**
	f_6	Std	3.09E+08	1.08E+10	1.23E+09	6.44E+08		f_{13}	Std	3.06E+0	2.95E+02	4.95E+0	2.90E+0
		Best	2.11E+06	4.18E+09	3.40E+07	3.84E+07			Best	3.76E+0	5.52E+01	4.10E+0	3.27E+0
		Mean	1.40E+03	3.56E+03	1.65E+03	**1.39E+03**			Mean	**4.04E+0**	4.37E+0	4.20E+0	4.13E+0
	f_7	Std	1.18E+02	4.84E+02	2.11E+02	1.11E+02		f_{14}	Std	2.79E-01	2.62E-01	2.64E-01	2.33E-01
		Best	1.27E+03	2.53E+03	1.31E+03	1.27E+03			Best	3.28E+0	3.63E+0	3.53E+0	3.71E+0 [b]
		Mean	2.09E+02	3.76E+03	5.97E-01	**1.08E-01**			Mean	2.05E+01	2.03E+01	**2.04E+01**	2.04E+01
	f_1	Std	3.75E+02	2.88E+03	4.58E-01	**7.24E-02**		f_8	Std	1.05E-01	1.17E-01	**1.38E-01**	1.05E-01
		Best	5.65075e-05	2.59E+02	3.99E-02	**2.0E-02**			Best	2.02552e+01	2.01E+01	**2.01E+01**	2.03E+01
		Mean	**1.30E+02**	8.07E+03	4.09E+02	9.89E+02			Mean	**8.36E+0**	4.61E+01	1.98E+01	**2.37E+0**
	f_2	Std	**1.31E+02**	2.62E+03	2.67E+02	5.91E+02		f_9	Std	**3.27E+0**	1.41E+01	6.69E+0	**1.23E+0**
		Best	**1.93164e-01**	1.76E+03	2.80E+01	1.99E+02			Best	**2.75172e+00**	1.73E+01	9.20E+0	**2.01E-01**
		Mean	3.0E+06	2.73E+07	1.37E+06	**7.91E+05**			Mean	**2.05E+01**	8.50E+01	4.59E+01	5.78E+01
	f_3	Std	6.57E+06	2.73E+07	1.25E+06	**7.35E+05**		f_{10}	Std	**7.11E+0**	1.69E+01	1.60E+01	2.42E+01
		Best	1.04099e+05	2.90E+06	3.47E+04	**3.65E+04**			Best	**8.40258e+00**	5.67E+01	1.45E+01	1.41E+01
		Mean	**1.42E+02**	1.11E+04	1.99E+03	3.13E+03			Mean	**3.96E+0**	8.50E+0	6.45E+0	6.55E+0
1.0E+04	f_4	Std	**1.42E+02**	3.91E+03	1.72E+03	1.97E+03	1.0E+04	f_{11}	Std	**1.38E+0**	1.36E+0	1.15E+0	1.39E+0
		Best	**1.68170e+00**	3.38E+03	3.02E+02	6.60E+02			Best	**1.21034e+00**	6.26E+0	3.41E+0	4.71E+0
		Mean	**7.76E-05**	9.30E+03	5.88E+02	2.04E+02			Mean	3.66E+03	3.12E+04	4.32E+03	**2.20E+03**
	f_5	Std	**8.06E-05**	3.73E+03	6.09E+02	4.31E+02		f_{12}	Std	5.64E+03	1.07E+04	5.21E+03	**2.37E+03**
		Best	**8.55801e-06**	8.95E+02	1.62E+01	1.12E+01			Best	1.08496e+01	1.11E+04	1.85E+02	**3.77E+01**
		Mean	7.75E+06	2.26E+08	2.49E+03	**2.11E+03**			Mean	1.71E+0	5.62E+0	1.71E+0	**8.31E-01**
	f_6	Std	1.80E+07	4.24E+08	3.56E+03	**3.63E+03**		f_{13}	Std	8.59E-01	3.51E+0	8.05E-01	**1.88E-01**
		Best	4.39794e+00	2.45E+06	1.93E+01	**1.07E+01**			Best	6.32417e-01	2.41E+0	8.59E-01	**5.84E-01**
		Mean	1.27E+03	1.27E+03	1.27E+03	**1.27E+03**			Mean	**3.50E+0**	3.86E+0	3.71E+0	3.76E+0
	f_7	Std	2.54E-01	3.23E-02	7.10E-02	**7.63E-02**		f_{14}	Std	**3.20E-01**	2.74E-01	2.90E-01	3.05E-01
		Best	1.26723e+03	1.27E+03	1.27E+03	**1.27E+03**			Best	**2.8502e+00**	3.34E+0	2.98E+0	3.02E+0
		Mean	4.13E+01	1.28E+02	1.44E-04	**1.01E-04**			Mean	2.03E+01	2.01E+01	2.02E+01	**2.02E+01**
	f_1	Std	5.66E+01	2.75E+02	6.73E-05	**6.56E-05**		f_8	Std	6.94E-02	7.53E-02	1.06E-01	**8.69E-02**
		Best	0.0E+0	1.72E+0	3.06E-05	**2.17E-05**			Best	2.01566e+01	2.0E+01	2.01E+01	**2.0E+01**
		Mean	9.11E+01	2.61E+02	**1.30E+0**	1.26E+01			Mean	5.27E+0	1.64E+01	5.86E+0	**4.14E-05**
	f_2	Std	7.93E+01	1.38E+02	**8.45E-01**	8.39E+0		f_9	Std	5.93E+0	1.04E+01	2.99E+0	**2.21E-05**
		Best	5.68E-14	1.26E+02	**1.04E-01**	9.45E-01			Best	0.000e+00	5.03E+0	2.98E+0	**5.08E-06**
		Mean	1.84E+06	5.86E+06	**2.77E+05**	3.13E+05			Mean	**2.03E+01**	6.36E+01	3.30E+01	5.10E+01
	f_3	Std	3.52E+06	3.52E+06	**1.53E+05**	1.60E+05		f_{10}	Std	**8.80E+0**	1.38E+01	1.19E+01	2.24E+01
		Best	9.96846e+04	2.30E+06	**3.24E+04**	4.15E+04			Best	**8.95463e+00**	4.12E+01	1.79E+01	1.29E+01
		Mean	2.91E+02	2.65E+03	**5.61E+0**	2.64E+02			Mean	**3.50E+0**	6.07E+0	5.72E+0	6.73E+0
1.0E+05	f_4	Std	1.47E+02	1.81E+03	**3.66E+0**	2.56E+02	1.0E+05	f_{11}	Std	**1.30E+0**	1.92E+0	1.69E+0	1.27E+0
		Best	5.63891e+01	1.20E+03	**8.97E-01**	1.19E+01			Best	**1.22304e+00**	3.91E+0	1.83E+0	4.47E+0
		Mean	**0.0E+0**	6.63E+02	6.17E+0	1.80E+02			Mean	1.66E+03	3.19E+03	1.33E+03	**1.20E+03**
	f_5	Std	**0.0E+0**	8.58E+02	4.82E+0	6.25E+02		f_{12}	Std	3.31E+03	1.38E+03	2.04E+03	**1.79E+03**
		Best	**0.000e+00**	5.69E+01	1.84E-01	1.08E-01			Best	5.41956e-02	1.82E+03	2.15E+0	**7.09E+0**
		Mean	2.35E+07	4.95E+03	1.81E+03	**3.27E+02**			Mean	6.90E-01	1.17E+0	7.85E-01	**4.50E-01**
	f_6	Std	6.47E+07	4.15E+03	3.46E+03	**1.49E+03**		f_{13}	Std	2.59E-01	3.87E-01	2.79E-01	**1.70E-01**
		Best	8.90593e+00	1.57E+03	1.77E-02	**1.01E+0**			Best	3.66839e-01	7.80E-01	3.79E-01	**1.88E-01**
		Mean	1.32E+03	1.27E+03	1.27E+03	**1.27E+03**			Mean	**2.84E+0**	3.73E+0	3.28E+0	3.46E+0
	f_7	Std	1.84E+02	6.70E-05	1.09E-04	**5.77E-05**		f_{14}	Std	**3.88E-01**	2.42E-01	3.11E-01	3.99E-01
		Best	1.26723e+03	1.27E+03	1.27E+03	**1.27E+03**			Best	1.74585e+00	3.39E+0	2.56E+0	2.47E+0

Table 2. Statistical Results of Mean, Std and Best of Benchmark Functions in 30 Dimension

FES	No.		PSO	FWA	IFWAFS	IFWABS	FES	No.		PSO	FWA	IFWAFS	IFWABS
		Mean	4.98E+03	4.63E+04	6.64E+02	**9.67E+01**			Mean	2.11E+01	**2.07E+01**	2.10E+01	2.10E+01 [t]
	f_1	Std	3.94E+03	9.20E+03	3.07E+02	**3.18E+01**		f_8	Std	6.20E-02	**9.39E-02**	7.81E-02	8.34E-02
		Best	6.39E+02	3.47E+04	1.86E+02	**4.31E+01**			Best	2.10E+01	**2.05E+01**	2.08E+01	2.07E+01
		Mean	**1.20E+04**	5.17E+04	4.14E+04	4.78E+04			Mean	1.52E+02	3.01E+02	1.88E+02	**8.61E+01**
	f_2	Std	**4.77E+03**	1.00E+04	1.16E+04	8.42E+03		f_9	Std	2.52E+01	2.61E+01	2.77E+01	**1.65E+01**
		Best	**6.41E+03**	3.05E+04	2.40E+04	3.42E+04			Best	1.01E+02	2.60E+02	1.37E+02	**5.68E+01**
		Mean	5.98E+07	4.92E+08	6.98E+07	**3.75E+07**			Mean	**2.51E+02**	6.21E+02	3.71E+02	3.58E+02
	f_3	Std	2.74E+07	2.22E+08	2.64E+07	**2.02E+07**		f_{10}	Std	**2.10E+01**	6.22E+01	7.13E+01	9.29E+01
		Best	1.97E+07	1.16E+08	1.95E+07	**8.57E+06**			Best	**2.15E+02**	5.20E+02	2.16E+02	2.07E+02
		Mean	**2.08E+04**	6.17E+04	6.44E+04	7.11E+04			Mean	3.13E+01	3.93E+01	3.32E+01	**3.04E+01**
1.00E+04	f_4	Std	**4.25E+03**	1.14E+04	1.42E+04	2.06E+04	1.00E+04	f_{11}	Std	3.56E+00	2.48E+00	4.03E+00	**3.50E+00**
		Best	**1.29E+04**	4.28E+04	4.03E+04	3.42E+04			Best	2.48E+01	3.29E+01	2.40E+01	**2.36E+01**
		Mean	**8.66E+03**	2.98E+04	1.61E+04	1.24E+04			Mean	1.15E+05	7.87E+05	1.94E+05	**1.25E+05**
	f_5	Std	**3.07E+03**	3.94E+03	2.87E+03	2.89E+03		f_{12}	Std	5.83E+04	2.05E+05	9.81E+04	**6.26E+04**
		Best	**1.40E+03**	2.35E+04	1.07E+04	7.92E+03			Best	1.66E+04	4.22E+05	4.06E+04	**3.83E+04**
		Mean	6.84E+08	1.54E+10	2.78E+07	**3.88E+05**			Mean	2.01E+01	5.45E+01	1.89E+01	**1.14E+01**
	f_6	Std	9.82E+08	4.39E+09	3.11E+07	**2.54E+05**		f_{13}	Std	2.30E+00	3.44E+01	3.96E+00	**2.21E+00**
		Best	6.34E+06	5.25E+09	2.68E+06	**7.02E+04**			Best	1.65E+01	2.05E+01	1.25E+01	**7.60E+00**
		Mean	4.80E+03	4.77E+03	4.76E+03	**4.72E+03**			Mean	1.35E+01	1.37E+01	**1.33E+01**	1.34E+01
	f_7	Std	1.31E+02	4.81E+01	4.53E+01	**2.20E+01**		f_{14}	Std	2.48E-01	3.19E-01	**3.58E-01**	3.90E-01
		Best	4.70E+03	4.73E+03	4.71E+03	**4.70E+03**			Best	1.30E+01	1.29E+01	**1.25E+01**	1.27E+01 [b]
		Mean	5.03E+03	8.81E+03	4.21E-02	**1.51E-02**			Mean	2.09E+01	2.03E+01	2.07E+01	2.07E+01 [t]
	f_1	Std	3.03E+03	5.27E+03	1.42E-02	**4.64E-03**		f_8	Std	6.64E-02	9.01E-02	1.12E-01	8.73E-02
		Best	6.07E+02	1.32E+03	1.71E-02	**6.80E-03**			Best	2.08E+01	2.01E+01	2.04E+01	2.05E+01
		Mean	**1.96E+03**	2.68E+04	4.00E+03	1.34E+04			Mean	6.16E+01	1.73E+02	7.16E+01	**2.88E+00**
	f_2	Std	**2.01E+03**	4.56E+03	1.22E+03	3.98E+03		f_9	Std	1.72E+01	3.80E+01	1.16E+01	**1.54E+00**
		Best	**2.44E+02**	1.63E+04	2.52E+03	5.93E+03			Best	2.76E+01	1.01E+02	5.39E+01	**8.99E-03**
		Mean	1.55E+07	8.68E+07	1.15E+07	**7.60E+06**			Mean	**1.34E+02**	4.66E+02	2.78E+02	3.29E+02
	f_3	Std	9.18E+06	3.47E+07	4.24E+06	**3.79E+06**		f_{10}	Std	**4.97E+01**	7.26E+01	7.11E+01	8.77E+01
		Best	4.68E+06	3.57E+07	6.47E+06	**3.88E+06**			Best	**7.14E+01**	3.14E+02	1.41E+02	2.04E+02
		Mean	**4.00E+03**	3.79E+04	2.85E+04	4.26E+04			Mean	**2.23E+01**	3.52E+01	3.04E+01	2.89E+01
1.00E+05	f_4	Std	**3.99E+03**	5.14E+03	9.81E+03	1.34E+04	1.00E+05	f_{11}	Std	**2.92E+00**	3.52E+00	3.29E+00	4.31E+00
		Best	**1.06E+03**	2.73E+04	1.24E+04	2.35E+04			Best	**1.77E+01**	2.51E+01	2.27E+01	2.26E+01
		Mean	8.27E+03	2.15E+04	9.46E+03	**8.08E+03**			Mean	7.12E+04	2.09E+05	5.29E+04	**3.55E+04**
	f_5	Std	2.17E+03	4.22E+03	2.85E+03	**2.10E+03**		f_{12}	Std	6.44E+04	7.24E+04	2.80E+04	**2.75E+04**
		Best	5.25E+03	1.37E+04	4.09E+03	**4.78E+03**			Best	1.15E+04	1.03E+05	1.82E+04	**6.33E+03**
		Mean	5.85E+08	3.30E+08	**1.50E+03**	3.36E+03			Mean	4.19E+00	1.17E+01	5.65E+00	**3.05E+00**
	f_6	Std	5.47E+08	4.12E+08	**3.07E+03**	4.61E+03		f_{13}	Std	1.76E+00	3.95E+00	1.41E+00	**6.91E-01**
		Best	3.17E+07	6.65E+06	**3.36E+01**	2.71E+01			Best	1.89E+00	6.50E+00	3.24E+00	**1.78E+00**
		Mean	4.82E+03	4.70E+03	4.70E+03	**4.70E+03**			Mean	**1.27E+01**	1.33E+01	1.31E+01	1.30E+01
	f_7	Std	1.35E+02	7.22E-03	3.02E-01	**2.80E-03**		f_{14}	Std	**3.60E-01**	3.12E-01	3.78E-01	3.71E-01
		Best	4.70E+03	4.70E+03	4.70E+03	**4.70E+03**			Best	**1.16E+01**	1.22E+01	1.21E+01	1.19E+01
		Mean	3.26E+03	3.70E+03	1.45E-03	**7.91E-04**			Mean	2.09E+01	**2.01E+01**	2.06E+01	2.05E+01 [t]
	f_1	Std	1.55E+03	3.50E+03	4.07E-04	**2.88E-04**		f_8	Std	5.87E-02	**9.29E-02**	1.22E-01	1.05E-01
		Best	6.79E+02	3.29E+02	7.70E-04	**3.53E-04**			Best	2.07E+01	**2.00E+01**	2.03E+01	2.03E+01
		Mean	2.50E+03	1.48E+04	**7.40E+02**	4.05E+03			Mean	6.04E+01	1.33E+02	3.99E+01	**8.01E-02**
	f_2	Std	2.38E+03	3.93E+03	**2.86E+02**	1.30E+03		f_9	Std	1.98E+01	2.87E+01	9.30E+00	**2.76E-01**
		Best	2.74E+02	8.92E+03	**3.27E+02**	2.02E+03			Best	3.52E+01	8.21E+01	2.59E+01	**2.51E-04**
		Mean	2.26E+07	4.52E+07	5.90E+06	**3.33E+06**			Mean	**1.28E+02**	4.10E+02	3.00E+02	3.34E+02
	f_3	Std	2.50E+07	1.83E+07	2.11E+06	**1.39E+06**		f_{10}	Std	**2.85E+01**	7.45E+01	8.15E+01	8.13E+01
		Best	5.58E+06	2.29E+07	2.20E+06	**9.61E+05**			Best	**7.28E+01**	2.78E+02	1.47E+02	1.82E+02
		Mean	**2.93E+03**	3.20E+04	1.09E+04	2.77E+04			Mean	**2.07E+01**	3.38E+01	3.02E+01	2.94E+01
3.00E+05	f_4	Std	**2.79E+03**	5.03E+03	4.48E+03	1.14E+04	3.00E+05	f_{11}	Std	**3.59E+00**	3.36E+00	3.16E+00	3.12E+00
		Best	**5.02E+02**	2.39E+04	4.04E+03	9.32E+03			Best	**1.45E+01**	2.41E+01	2.51E+01	2.43E+01
		Mean	7.61E+03	1.62E+04	8.15E+03	**7.36E+03**			Mean	7.35E+04	9.31E+04	2.92E+04	**1.57E+04**
	f_5	Std	1.99E+03	3.89E+03	2.18E+03	**1.66E+03**		f_{12}	Std	3.75E+04	4.03E+04	1.87E+04	**1.69E+04**
		Best	3.30E+03	7.42E+03	5.24E+03	**4.29E+03**			Best	1.48E+04	3.56E+04	1.21E+04	**6.53E+02**
		Mean	6.98E+08	2.02E+07	2.76E+03	**1.50E+03**			Mean	3.24E+00	6.61E+00	3.45E+00	**1.98E+00**
	f_6	Std	1.05E+09	3.72E+07	3.95E+03	**3.43E+03**		f_{13}	Std	2.14E+00	4.54E+00	6.81E-01	**4.19E-01**
		Best	4.11E+07	1.02E+05	2.90E+01	**2.75E+01**			Best	1.85E+00	2.20E+00	2.40E+00	**1.28E+00**
		Mean	4.86E+03	4.70E+03	4.70E+03	**4.70E+03**			Mean	**1.23E+01**	1.31E+01	1.30E+01	1.31E+01
	f_7	Std	1.62E+02	2.00E-04	1.25E-04	**1.55E-05**		f_{14}	Std	**4.30E-01**	3.06E-01	3.37E-01	2.87E-01
		Best	4.70E+03	4.70E+03	4.70E+03	**4.70E+03**			Best	**1.13E+01**	1.22E+01	1.22E+01	1.25E+01

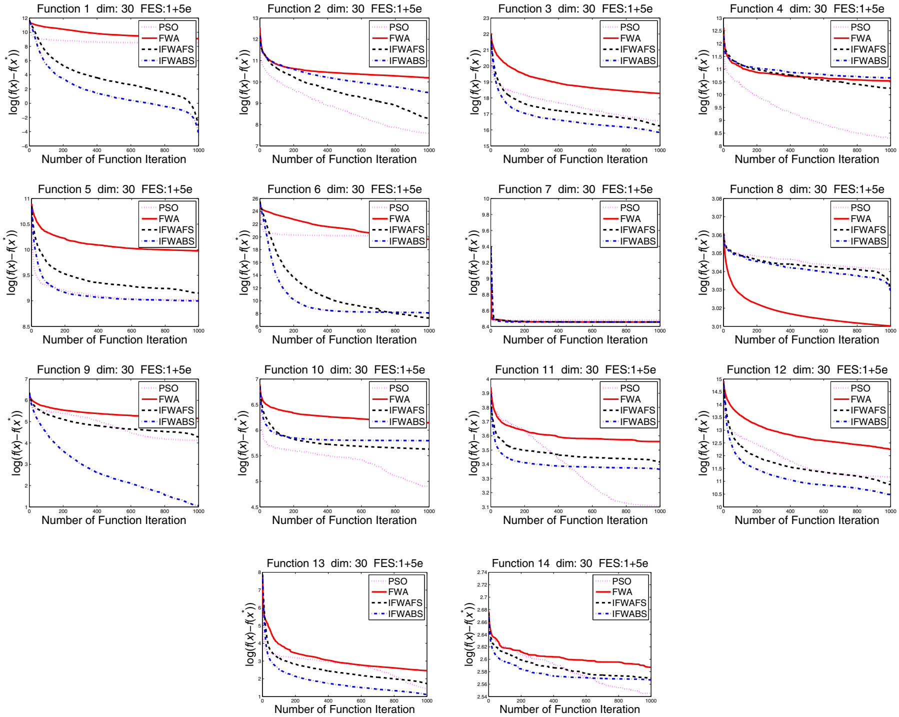

Fig. 6. Convergence cures on the benchmark functions

Acknowledgements. This work was supported by the Provincial Natural Science Foundation of Fujian under grants number 2012J01246, Foundation of supporting Province's College and University of Fujian under grants number JK2011035 and Foundation for Starting Research of Fujian university of technology under grants number E0600100.

References

1. Beni, G., Wang, J.: Swarm intelligence in cellular robotic systems. Robots and Biological Systems: Towards a New Bionics? 703–712 (1993)
2. De Castro, L.N., Von Zuben, F.J.: Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation 6(3), 239–251 (2002)
3. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 26(1), 29–41 (1996)
4. Gao, H., Diao, M.: Cultural firework algorithm and its application for digital filters design. International Journal of Modelling, Identification and Control 14(4), 324–331 (2011)
5. Janecek, A., Tan, Y.: Iterative improvement of the multiplicative update nmf algorithm using nature-inspired optimization. In: 2011 Seventh International Conference on Natural Computation (ICNC), vol. 3, pp. 1668–1672. IEEE (2011)

6. Janecek, A., Tan, Y.: Swarm intelligence for non-negative matrix factorization. International Journal of Swarm Intelligence Research (IJSIR) 2(4), 12–34 (2011)
7. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. Journal of Global Optimization 39(3), 459–471 (2007)
8. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
9. Pei, Y., Zheng, S., Tan, Y., Takagi, H.: An empirical study on influence of approximation approaches on enhancing fireworks algorithm. In: IEEE International Conference on System, Man and Cybernetics (SMC 2012), pp. 14–17. IEEE, Seoul (2012)
10. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: The 1998 IEEE International Conference on Evolutionary Computation Proceedings: IEEE World Congress on Computational Intelligence, pp. 69–73. IEEE (1998)
11. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11(4), 341–359 (1997)
12. Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., et al.: Problem definitions and evaluation criteria for the cec 2005 special session on real-parameter optimization. In: 2005 IEEE Congress on Evolution Computation (CEC), pp. 1–15. IEEE (2005)
13. Tan, Y., Xiao, Z.: Clonal particle swarm optimization and its applications. In: IEEE Congress on Evolutionary Computation (CEC 2007), pp. 2303–2309. IEEE (2007)
14. Tan, Y., Zhu, Y.: Fireworks algorithm for optimization. Advances in Swarm Intelligence pp. 355–364 (2010)
15. Zheng, S., Janecek, A., Tan, Y.: Enhanced fireworks algirithm. In: IEEE International Conference on Evolutionary Computation. IEEE (submitted, 2013)
16. Zheng, X.X., Y.J., H.F., L.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Accepted by Neurocomputing (2013)