2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

A Hybrid Biogeography-Based Optimization and Fireworks Algorithm

Bei Zhang, Min-Xia Zhang and Yu-Jun Zheng
College of Computer Science & Technology
Zhejiang University of Technology
Hangzhou, Zhejiang, China 310023
Email: yujun.zheng@computer.org

***Abstract*—The paper presents a hybrid biogeography-based optimization (BBO) and fireworks algorithm (FWA) for global optimization. The key idea is to introduce the migration operator of BBO to FWA, in order to enhance information sharing among the population, and thus improve solution diversity and avoid premature convergence. A migration probability is designed to integrate the migration of BBO and the normal explosion operator of FWA, which can not only reduce the computational burden, but also achieve a better balance between solution diversification and intensification. The Gaussian explosion of the enhanced FWA (EFWA) is reserved to keep the high exploration ability of the algorithm. Experimental results on selected benchmark functions show that the hybrid BBO_FWA has a significantly performance improvement in comparison with both BBO and EFWA.**

I. Introduction

The complexity of real-world engineering optimization problems gives rise to various kinds of metaheuristics that use stochastic techniques to effectively explore the search space for a global optimum. Many of their names, such as genetic algorithms [1] and simulated annealing [2], attest to the influence of natural or biological analogies, and ingeniously harnessing such analogies often leads to very effective computer algorithms [3].

Fireworks algorithm (FWA) [4], [5] is a relatively new global optimization method inspired by the phenomenon of fireworks explosion, where fireworks and sparks are analogues to solutions to a given problem, and an explosion can be viewed as a search in the solution space around the firework. The main principle of the algorithm is that "good" fireworks generate more sparks within smaller explosion areas to facilitate exploitation (local search), while "bad" fireworks generate fewer sparks within larger explosion areas to enhance exploration (global search). Numerical experiments on a set of benchmark functions show that, the FWA can converge to a global optimum much faster than typical particle swarm optimization (PSO) algorithms including [6] and [7]. FWA has also been applied to and shown its efficiency on many practical problems [8]–[11].

In the original FWA, the individuals solutions (fireworks and sparks) do not directly share information among each other. The enhanced FWA (EFWA) [5] modifies the mutation operator such that a part of fireworks can generate sparks towards the location of the best firework of the population (which is similar to learning from $gbest$ in PSO). Even so, the information sharing among the population is still very limited, which often leads to low solution diversity and premature convergence. Thus, there is much potential to improve the performance of the population-based evolutionary algorithm by enhancing the information exchange between the individuals.

Biogeography-based optimization (BBO) [12] is another bio-inspired optimization method which has received much attention in recent years. It is known as a metaheuristic that makes the best use of information sharing among the whole population based on the migration operator [13]. In this paper, we propose a new hybrid FWA by integrating the BBO's migration operator to enhance the information sharing among the fireworks and sparks. This strategy helps to improve the FWA in two aspects: (1) Reducing the computational burdens of normal explosions in FWA to a certain extent; (2) Increasing the potential solution diversity of the population and thus improving the exploration ability of FWA. Numerical experiments show that the hybrid algorithm has significant performance advantage over the FWA and BBO on benchmark functions.

In the remainder of the paper, we first briefly introduce FWA and BBO in Section II, and present our hybrid algorithm in Section III. Section IV presents the experimental results, and finally Section V concludes.

II. Background

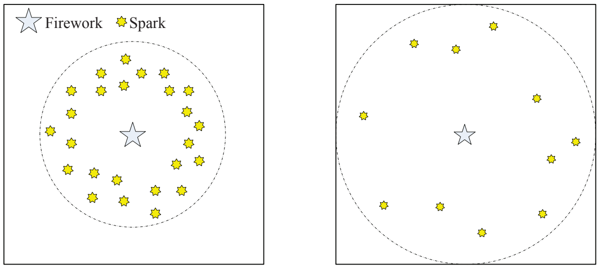
A. Fireworks Algorithm

Originally proposed by Tan and Zhu [4], FWA is a nature-inspired optimization method simulating the explosion process of fireworks. At beginning, the algorithm randomly selects in the search space a certain number of locations, each for exploding a firework to produce a set of sparks; High quality individuals among the fireworks and sparks are probably chosen as the new fireworks to be exploded in the next generation, and the process continues until the termination criterion is met.

As mentioned, fireworks with better fitness have a smaller explosion amplitude and a larger number of explosion sparks than those with lower fitness, as illustrated in Fig. 1 [4]. Without loss of generality, assume the problem is to minimize the objective function f, then the explosion amplitude A_i and the number s_i of sparks for each firework X_i are respectively

This work was supported by National Natural Science Foundation of China under Grant No. 61105073 and 61325019.

978-1-4799-1488-3/14/$31.00 ©2014 IEEE



(a) good firework (b) bad firework

Fig. 1. Illustration of fireworks explosion in FWA.

calculated as follows:

$$A_i = \hat{A} \cdot \frac{f(X_i) - f^{\min} + \epsilon}{\sum_{j=1}^{n}(f(X_j) - f^{\min}) + \epsilon} \quad (1)$$

$$s_i = M_e \cdot \frac{f^{\max} - f(X_i) + \epsilon}{\sum_{j=1}^{n}(f^{\max} - f(X_j)) + \epsilon} \quad (2)$$

where $\hat{A}$ and M_e are two control parameters, n is the population size, $f^{\max}$ and $f^{\min}$ are respectively the maximum and minimum objective values among the n fireworks, and ϵ is a small constant to avoid division by zero.

To avoid overwhelming effects of splendid fireworks, the number of sparks is bounded as follows:

$$s_i = \begin{cases} s^{\min} & \text{if } s_i < s^{\min} \\ s^{\max} & \text{else if } s_i > s^{\max} \\ round(s_i) & \text{else} \end{cases} \quad (3)$$

In EFWA [5], for each dimension k of the problem, the explosion amplitude A_k is also bounded as follows to avoid too small explosion amplitude:

$$A_k = \begin{cases} A_k^{\min} & \text{if } A_k < A_k^{\min} \\ A_k & \text{else} \end{cases} \quad (4)$$

And it is suggested to use a nonlinear decreasing $A_k^{\min}$ in the EFWA as follows:

$$A_k^{\min} = A^{\mathrm{U}} + \frac{A^{\mathrm{U}} - A^{\mathrm{L}}}{t^{\max}} \cdot \sqrt{(2t^{\max} - t)t} \quad (5)$$

where A^{U} and A^{L} are respectively the upper and lower limits of the explosion amplitude, t is the current number of generations (or function evaluations), and $t^{\max}$ is the maximum number of generations (or function evaluations).

For a D-dimensional problem, each spark X_j of X_i is obtained by randomly selecting z dimensions ($z < D$), and at each dimension k add a displacement to the original location as follows:

$$X_{j,k} = X_{i,k} + A_k \cdot rand(-1, 1) \quad (6)$$

where $rand$ is a function generating a random number uniformly distributed in a given range.

To keep the diversity, another type of Gaussian explosion is performed on a small number M_g of randomly selected fireworks. FWA and EFWA respectively use the following equations for get the position of a Gaussian spark at each dimension k:

$$X_{j,k} = X_{i,k} \cdot Gaussian(1, 1) \quad (7)$$

$$X_{j,k} = X_{i,k} + (X_{b,k} - X_{i,k}) \cdot Gaussian(0, 1) \quad (8)$$

where $Gaussian(\mu, \sigma)$ generates a Gaussian random value with mean μ and standard deviation σ, and X_b is the best individual found so far.

In both types of explosion, if any dimension of a spark is out of the search space, it is reset to a random position with uniform distribution in the search space.

At each generation, among the current sparks and fireworks, the best location is always chosen to the next generation. FWA selects other $(n - 1)$ fireworks with probabilities proportional to their distances to other individuals, but EFWA suggests to select the $(n - 1)$ fireworks randomly. Algorithm 1 presents the main procedure of EFWA.

Algorithm 1 The enhanced fireworks algorithm (EFWA).

Randomly initialize a population P of n fireworks;
while (stop criterion is not met) **do**
  let R be the empty set of sparks;
  for each firework $X_i \in P$ **do**
    calculate s_i according to Equations (2) and (3);
    calculate A_i according to Equation (1);
    for $j = 1$ to s_i **do**
      produce a spark $X_j = X_i$;
      for $k = 1$ to D **do**
        if $rand(0, 1) < 0.5$ **then**
          set $X_{j,k}$ according to Equations (4) and (6);
      $R = R \cup \{X_j\}$;
  randomly select a set Q of M_g fireworks;
  for each firework $X_i \in Q$ **do**
    produce a spark $X_j = X_i$;
    for $k = 1$ to D **do**
      if $rand(0, 1) < 0.5$ **then**
        Set X_j^k according to Equation (8);
    $R = R \cup \{X_j\}$;
  $R = R \cup P$;
  select the best and the other $n - 1$ individuals for P;
  update $A_k^{\min}$ according to Equation (5);
return the best individual found so far.

B. Biogeography-Based Optimization

Borrowing ideas from biogeographic evolution over space and time, BBO [12] is another population-based heuristic to optimization problems. In BBO, each solution in the population is analogous to a "habitats" or "islands", the solution components are analogous to a set of suitability index variables (SIVs), and the fitness of the solution is analogous to the species richness or habitat suitability index (HSI) of the island. The method mainly works on the principle of immigration and emigration of the species from one island to another, and therefore evolves the islands to find better solutions to the problem. BBO has proven itself a competitive method to other well-known heuristics on a wide set of problems (e.g., [12], [14]–[17]).

A distinct feature of BBO is its migration operator, which indicates that high HSI islands have a high species emigration

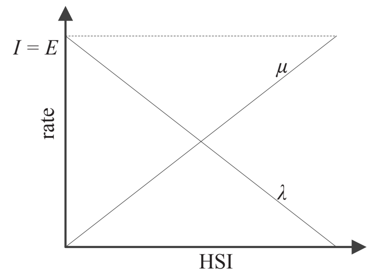


Fig. 2. A linear migration model.

rate μ and low HSI islands have a high species immigration rate λ. The migration rates are functions of the HSI value or fitness of the islands. Fig. 2 presents a simple linear migration model, in which λ_i and μ_i of each island X_i are calculated as follows (but there are also other nonlinear migration models can be used [12], [18]):

$$\lambda_i = I(\frac{f_i - f^{\min}}{f^{\max} - f^{\min}}) \tag{9}$$

$$\mu_i = E(\frac{f^{\max} - f_i}{f^{\max} - f^{\min}}) \tag{10}$$

where I and E are respectively the maximum possible immigration rate and emigration rate, which are typically both set to 1.

At each time, the migration operator migrates an SIV from an emigrating island to an immigrating island, which are probabilistically selected according to the emigration and immigration rates of the islands. Algorithm 2 presents the basic procedure of a probably migration operation on an island X_i.

Algorithm 2 The migration operation in BBO.

for $k = 1$ to D **do**
  if $rand() < \lambda_i$ **then**
    Select an emigrating island X_j with probability $\propto \mu_j$;
    $X_{i,k} \leftarrow X_{j,k}$;

III. A Hybrid Biogeography-Based Optimization and Fireworks Algorithm

For a high-dimensional optimization problem, the fitness value of a solution is co-determined by its component values of all dimensions. A solution that has discovered the region corresponding to the global optimum in some dimensions may have a low fitness value because of the poor quality in the other dimensions. Thus, some well-known population-based evolutionary algorithms, including differential evolution (DE) [19], comprehensive learning PSO [20], fully informed PSO [21], enable the individuals to make the utmost use of the beneficial information in the population and thus perform a very effective search.

In the original FWA, the individuals in the population never directly interacts with each other. EFWA makes a slight improvement by employing Eq. (8) to let some individuals learn from the best individual found so far. On the other hand, FWA uses a distance-based metric for selecting individuals in less crowded regions to the next generation so as to keep diversity. But such a selection operator is computational expensive, and thus EFWA turns to a random selection operator.

In the hybrid algorithm, we employ a diversification strategy that integrates the BBO's migration mechanism to FWA. In fact, the migration operator of BBO and the normal explosion operator of FWA both have their advantages and disadvantages:

- The migration operator contributes greatly to the information sharing between different individuals by making low HSI islands probably learning from high HSI ones. It is also computational cheap (which requires only one function evaluation at each time, while the explosion requires s_i evaluations).
- The normal explosion operator provides a good balance between exploration and exploitation. In particular, when a high quality firework is nearby the global optimum, the explosion enables an intensive local search around the optimum.

To combine their advantages while reducing their disadvantages as much as possible, we introduce a migration probability, denoted by ρ, to the hybrid algorithm. Each firework X_i has a probability of ρ to apply the migration operator, and a probability of $(1 - \rho)$ to explode.

Since the migration operator helps to enhance the information sharing and increase the solution diversity, and the Gaussian explosion also utilizes the information of the global best, we do not use the elitism method that always put the best known individual to the new population (note that the global best is always recorded by the algorithm). Algorithm 3 presents an overview of the hybrid BBO_FWA.

Algorithm 3 The hybrid BBO_FWA.

Randomly initialize a population P of n fireworks;
while (stop criterion is not met) **do**
  for each firework $X_i \in P$ **do**
    if $rand(0, 1) < \rho$ **then**
      use Algorithm 2 to perform migration on X_i;
    else
      use Lines 5-12 of Algorithm 1 to produce sparks;
  for $j = 1$ **to** M_g **do**
    select a random firework X_i;
    use Lines 15-18 of Algorithm 1 to produce a spark;
  add the new individuals to P;
  randomly select n individuals for P;
  update $A_k^{\min}$ and the migration rates;
return the best individual found so far.

In general, for problems with complex objective functions, we prefer to set a larger value of the probability ρ which allows a smaller number of function evaluations to reduce the computational burden. A larger ρ can also enhance the solution diversity and thus improve the exploration ability for multimodal functions. In contrast, a smaller ρ is more suitable for those functions whose optima are often located in very narrow or sharp ridges, since the normal explosion operator can diverse the search along different directions and thus decrease the chance of skipping the optima. Empirically, the value of ρ can range from 0.5 to 0.8 to achieve an obvious performance improvement over both BBO and FWA/EFWA. In the next section we will evaluate the influence of different values of ρ to the algorithm performance on a set of benchmark functions.

IV. Experiments

A. Experimental Setup

Yao et al. [22] provided a set 23 well-known test functions for global optimization, from which we choose the first 13 high-dimensional functions as the benchmark problems, a brief introduction of which is given in Table I. In this paper, we set the dimensions $D = 30$ for all the functions; For those functions having non-zero optimal values, we also simply add some constants to the expression such that their optimal values all become 0. The maximum number of function evaluations (NFE) are set to 300'000 for f_3, f_4, and f_5, and 150'000 for the other 10 functions.

TABLE I. A summary of the benchmark functions used in the paper.

ID	Function	Search range	X^*
f_1	Sphere	$[-100, 100]^D$	0^D
f_2	Schwefel 2.22	$[-10, 10]^D$	0^D
f_3	Schwefel 1.2	$[-100, 100]^D$	0^D
f_4	Schwefel 2.21	$[-100, 100]^D$	0^D
f_5	Rosenbrock	$[-2.048, 2.048]^D$	1^D
f_6	Step	$[-100, 100]^D$	0^D
f_7	Quartic	$[-1.28, 1.28]^D$	0^D
f_8	Schwefel	$[-500, 500]^D$	420.9687^D
f_9	Rastrigin	$[-5.12, 5.12]^D$	0^D
f_{10}	Ackley	$[-32.768, 32.768]^D$	0^D
f_{11}	Griewank	$[-600, 600]^D$	0^D
f_{12}	Penalized1	$[-50, 50]^D$	1^D
f_{13}	Penalized2	$[-50, 50]^D$	1^D

We compare our hybrid BBO_FWA with the basic BBO and EFWA on the benchmark problems. For BBO, we set $I = E = 1$ and $n = 50$. For EFWA, we set $\widehat{A} = 40$, $M_e = 50$, $M_g = 5$, $s_i^{\max} = 40$, $s_i^{\max} = 2$, $A^{\mathrm{U}} = 0.02(X_k^{\max} - X_k^{\min})$, $A^{\mathrm{L}} = 0.001(X_k^{\max} - X_k^{\min})$, and $n = 5$, as suggested in [4] and [5]. BBO_FWA inherits the parameter settings of BBO and EFWA, except it uses a population size $n = 10$. As we will see in the next subsection, for different problems, the best value of p often varies. Nevertheless, for the sake of fairness, we will use a general setting of $p = 0.68$ in the comparative experiments, rather than fine-tuning the value for each problem. The experiment environment is a computer of Intel Core i5-2430M processor and 4GB memory.

B. Evaluation on the Migration Probability

On each benchmark problem, we first test the performance of BBO_FWA with p ranges from 0 to 1 with an interval of 0.05. For each given value of p, the algorithm is run for 60 times, and the mean best function error value is recorded. The results reveal that, on most of the problems, BBO_FWA obtains the minimum mean errors with p in the range [0.6,0.75].

Therefore, we re-test BBO_FWA with p ranges from 0.6 to 0.75 with an interval of 0.01, and record the mean best values obtained over the 60 runs. Combining the results, we present the best value of p with which BBO_FWA obtains the minimum mean error $f^{\min}$ on each problem in Table II. Note that for f_6, any value of p in the range of [0.3,0.95] makes the algorithm obtain the global optimum.

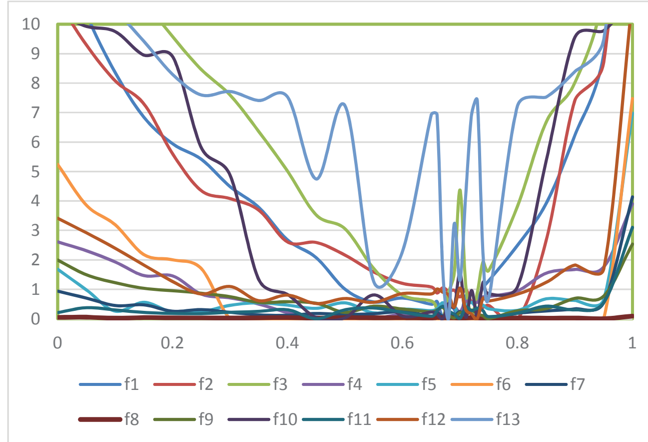


Fig. 3. The variation of BBO_FWA performance with the value of p.

TABLE II. The best value of p for each benchmark problem.

Function	f_1	f_2	f_3	f_4	f_5	f_6	
p^*	0.7	0.8	0.67	0.55	0.7	0.3-0.95	
Function	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}
p^*	0.75	0.95	0.75	0.67	0.45	0.73	0.68

The curves in Fig. 3 illustrates the variation of the mean bests with p, where the y-axis values are set as $\log(f/f^{\min})$, i.e., the natural logarithm of the ratio of the current mean error to the minimum mean error.

As we can see, the relation between the algorithm performance and the value of p is generally non-linear and non-monotonic. First of all, the setting of $p = 0$ (i.e., the only use of explosion) or $p = 1$ (i.e., the only use of migration) always leads to the worst results. Thus, the combination of two operators is always preferred. Roughly, we suggest using a migration probability in the range of [0.6,0.75] for most unknown problems. However, in order to achieve more precise results, fine-tuning of the parameter p is needed.

C. Comparison with BBO and EFWA

On each test problem, we respectively run three algorithms for 60 times with different random seeds, and compute mean best values averaged over the 60 runs. Table I presents results of the comparative experiments. In columns 1-3, the upper part of each cell corresponds to mean best and the lower part corresponds to standard deviation, and the bold values indicate the minimum mean bests among the three algorithms. We also conduct paired t-tests between BBO_FWA and the other two methods. In columns 2 and 3 of Table III, $^+$ indicates that BBO_FWA has significant performance improvement over EFWA and BBO (at 95% confidence level), and $^-$ indicates vice versa.

As we can see from the results, the general performance of BBO_FWA is much better than EFWA and BBO. In particular, BBO_FWA has statistically significant improvement over EFWA on all the problems, which shows that the integration of migration operator can effectively improve the performance of the normal explosion operator of EFWA. Among the 13 test problems, BBO_FWA obtains the minimum mean bests on 11 problems, and BBO has the minimum mean bests only on f_7 and f_8. Moreover, BBO only has significant improvement over BBO_FWA on f_7, and on f_8 there is no statistically significant difference between BBO and BBO_FWA. On the contrary, BBO_FWA has significant improvement over BBO on 10 test problems. This also shows that the integrated explosion/migration operator of BBO_FWA can perform more effective search than the single migration operator of BBO.

Fig 4(a)−(m) respectively present the convergence curves of the algorithms on the 13 test problems, from which we can see that, BBO_FWA converges faster than the other two algorithms on most of the problems (including $f_1 - f_3$, $f_5 - f_7$, f_9, and $f_{11} - f_{13}$). BBO and EFWA respectively converge faster on f_4 and f_{10} at early stages, but are both overtaken by BBO_FWA by later stages. In particular, the convergence processes of EFWA are left far behind BBO_FWA on complex multimodal problems such as f_6, f_{12} and f_{13} because it is easy to be trapped by local optima. In fact, on these functions EFWA can sometimes get similar results to BBO_FWA, but it also occasionally obtains very large error values, which greatly degrades the average performance over the 60 runs. The BBO_FWA's advantages in convergence speed demonstrate that the integration of migration to FWA can improve the solution diversity to a great extent, and thus effectively avoid premature convergence.

On the other hand, on many problems the convergence curves of BBO and BBO_FWA have similar shapes, but in general BBO_FWA converges faster and reaches better results. This also demonstrates that the combination of normal explosion operator can improve the exploration ability of the migration operator, resulting in much more precise optima. In summary, the integrated explosion/migration operator of BBO_FWA can achieve a better balance between solution diversification and intensification, and thus exhibit much better performance than both BBO and EFWA.

V. Conclusions

FWA is a metaheuristic method inspired by the phenomenon of fireworks explosion, and has received much interest in recent years. FWA has drawbacks of high computational cost and lacking of information sharing among the population. In this paper, we propose a hybrid algorithm BBO_FWA, which integrates the migration operator of BBO with the explosion operator of FWA based on a migration probability, and thus effectively increases the solution diversity without harming the exploitation ability of FWA. Comparative Experiments show that BBO_FWA outperforms both FWA and BBO on a set of well-known benchmark functions.

In this paper, BBO_FWA uses a fixed migration probability p which is easy to implement. However, as indicated by the numerical experiments, the parameter value needs to be fine-tuned to obtain the best results on different problems. We are currently studying a self-adaptive strategy, which can dynamically adjust the probability according to the state of the search. Ongoing work also includes extending the hybrid algorithm for constrained optimization and multiobjective optimization problems.

TABLE III. The experiment results of the three algorithms.

Problem	EFWA	BBO	BBO_FWA
f_1	2.56E+01 $^+$	3.49E+00 $^+$	**1.92E-13**
	(7.56E+00)	(1.53E+00)	(5.49E-13)
f_2	5.03E+00 $^+$	6.94E-01 $^+$	**3.95E-10**
	(2.74E+00)	(1.15E-01)	(2.63E-10)
f_3	1.09E+03 $^+$	1.28E+01 $^+$	**1.62E-15**
	(5.08E+02)	(6.84E+00)	(1.05E-14)
f_4	7.22E-01 $^+$	2.91E+00 $^+$	**3.66E-02**
	(7.42E-01)	(4.30E-01)	(7.65E-02)
f_5	1.24E+05 $^+$	3.53E+02 $^+$	**5.54E+01**
	(9.06E+04)	(1.06E+03)	(8.62E+01)
f_6	1.13E+03 $^+$	3.17E+00 $^+$	**0.00E+00**
	(3.24E+02)	(1.97E+00)	(0.00E+00)
f_7	1.75E-01 $^+$	**5.72E-03**	7.23E-03
	(7.14E-02)	(4.07E-03)	(4.51E-03)
f_8	1.00E+04 $^+$	**8.92E+00** $^-$	9.19E+03
	(9.55E+02)	(3.34E+00)	(9.22E+02)
f_9	1.11E+02 $^+$	1.70E+00	**1.63E+00**
	(2.31E+01)	(5.88E-01)	(9.64E-01)
f_{10}	4.89E+00 $^+$	9.04E-01 $^+$	**3.09E-10**
	(1.01E+00)	(2.42E-01)	(2.78E-10)
f_{11}	2.31E-02 $^+$	1.00E+00 $^+$	**1.47E-02**
	(1.26E-02)	(4.02E-02)	(1.54E-02)
f_{12}	2.52E+02 $^+$	9.14E-02 $^+$	**2.51E-02**
	(5.89E+02)	(4.74E-02)	(7.79E-02)
f_{13}	2.09E+04 $^+$	5.00E-01 $^+$	**9.16E-04**
	(2.64E+04)	(1.91E-01)	(3.06E-03)

References

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1975.

[2] S. Kirkpatrick, D. G. Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[3] J. O. Kephart, "Learning from nature," *Science*, vol. 331, pp. 682–683, 2011.

[4] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in *Advances in Swarm Intelligence*, ser. Lect. Note. Comput. Sci., Y. Tan, Y. Shi, and K. Tan, Eds. Springer, 2010, vol. 6145, pp. 355–364.

[5] S. Zheng, A. Janecek, and Y. Tan, "Enhanced fireworks algorithm," in *IEEE Congress on Evolutionary Computation*, 2013, pp. 2069–2077.

[6] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *IEEE International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948.

[7] Y. Tan and Z. Xiao, "Clonal particle swarm optimization and its applications," in *IEEE Congress on Evolutionary Computation*, 2007, pp. 2303–2309.

[8] A. Janecek and Y. Tan, "Iterative improvement of the multiplicative update nmf algorithm using nature-inspired optimization," in *2011 Seventh International Conference on Natural Computation*, vol. 3, 2011, pp. 1668–1672.

[9] H. Gao and M. Diao, "Cultural firework algorithm and its application for digital filters design," *International Journal of Modelling, Identification and Control*, vol. 14, no. 4, pp. 324–331, 2011.

[10] W. He, G. Mi, and Y. Tan, "Parameter optimization of local-concentration model for spam detection by using fireworks algorithm," in *Advances in Swarm Intelligence*, ser. Lecture Notes in Computer Science, Y. Tan, Y. Shi, and H. Mo, Eds. Springer Berlin Heidelberg, 2013, vol. 7928, pp. 439–450.

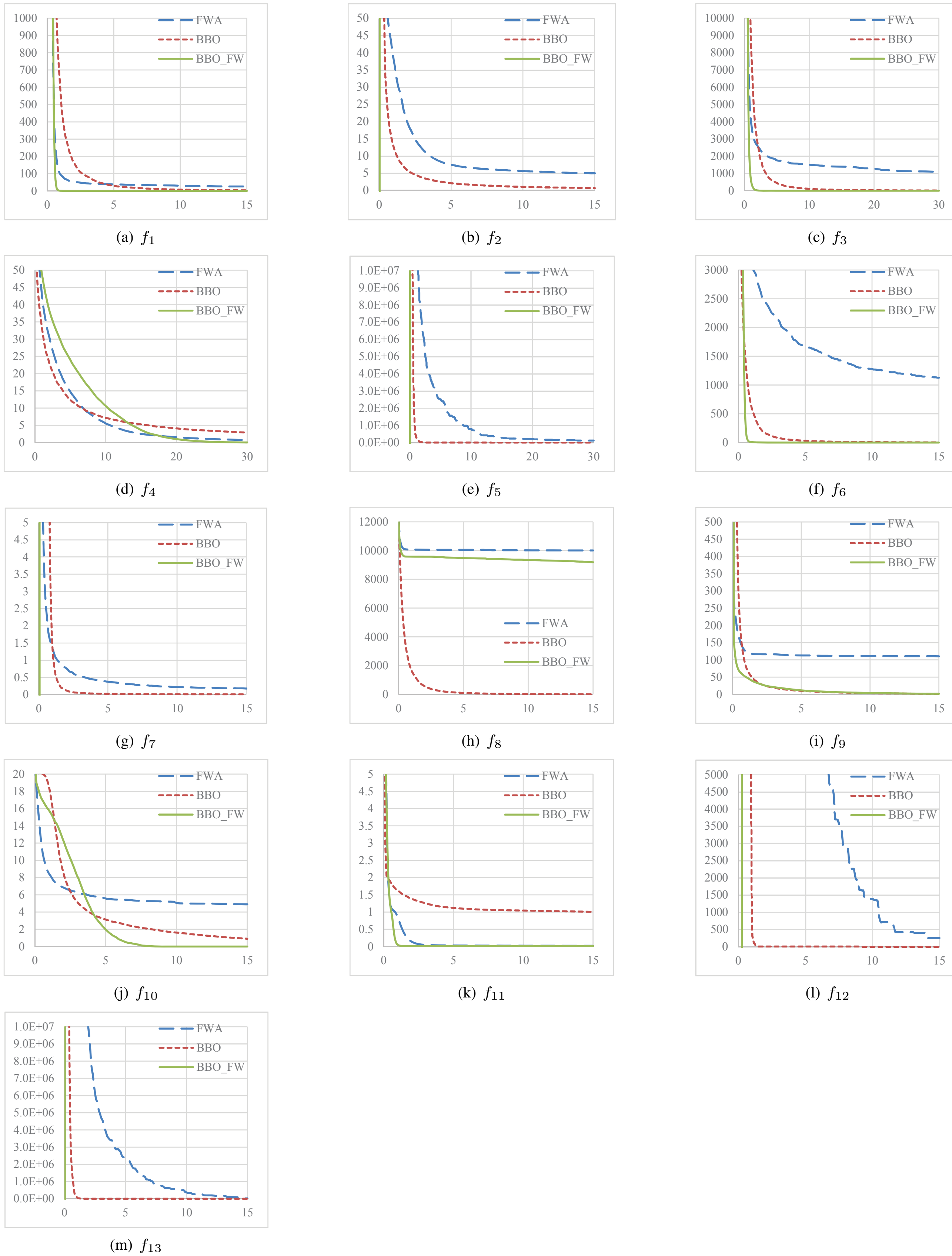


Fig. 4. The algorithms' convergence curves on the benchmark problems, where the x-axis values are NFE (10^4) and the y-axis values are mean best errors.

[11] Y.-J. Zheng, Q. Song, and S.-Y. Chen, "Multiobjective fireworks optimization for variable-rate fertilization in oil crop production," *Appl. Soft Comput.*, vol. 13, no. 11, pp. 4253–4263, 2013.

[12] D. Simon, "Biogeography-based optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 6, pp. 702–713, 2008.

[13] H. Ma, "An analysis of the equilibrium of migration models for biogeography-based optimization," *Inform. Sci.*, vol. 180, no. 18, pp. 3444–3464, 2010.

[14] A. Bhattacharya and P. Chattopadhyay, "Biogeography-based optimization for different economic load dispatch problems," *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1064–1077, 2010.

[15] U. Singh, H. Kumar, and T. Kamal, "Design of yagi-uda antenna using biogeography based optimization," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 10, pp. 3375–3379, 2010.

[16] I. Boussaïd, A. Chatterjee, P. Siarry, and M. Ahmed-Nacer, "Biogeography-based optimization for constrained optimization problems," *Comput. Oper. Res.*, vol. 39, no. 12, pp. 3293–3304, 2012.

[17] Y.-J. Zheng, H.-F. Ling, H.-H. Shi, H.-S. Chen, and S.-Y. Chen, "Emergency railway wagon scheduling by hybrid biogeography-based optimization," *Comput. Oper. Res.*, vol. 43, no. 3, pp. 1–8, 2014.

[18] H. Ma and D. Simon, "Blended biogeography-based optimization for constrained optimization," *Engin. Appl. Artif. Intell.*, vol. 24, no. 3, pp. 517–525, 2011.

[19] R. Storn and K. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[20] J. J. Liang, A. K. Qin, P. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, 2006.

[21] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 204–210, 2004.

[22] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, vol. 3, no. 2, pp. 82–102, 1999.