

## Research Article

# Self-Adaptive Step Firefly Algorithm

Shuhao Yu,<sup>1,2</sup> Shanlin Yang,<sup>1</sup> and Shoubao Su<sup>2</sup>

<sup>1</sup> Institute of Computer Network Systems, Hefei University of Technology, Hefei 230009, China

<sup>2</sup> School of Information Engineering, West Anhui University, Lu'an 237012, China

Correspondence should be addressed to Shuhao Yu; yush@wxc.edu.cn

Received 1 July 2013; Accepted 17 September 2013

Academic Editor: Sabri Arik

Copyright © 2013 Shuhao Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the standard firefly algorithm, each firefly has the same step settings and its values decrease from iteration to iteration. Therefore, it may fall into the local optimum. Furthermore, the decreasing of step is restrained by the maximum of iteration, which has an influence on the convergence speed and precision. In order to avoid falling into the local optimum and reduce the impact of the maximum of iteration, a self-adaptive step firefly algorithm is proposed in the paper. Its core idea is setting the step of each firefly varying with the iteration, according to each firefly's historical information and current situation. Experiments are made to show the performance of our approach compared with the standard FA, based on sixteen standard testing benchmark functions. The results reveal that our method can prevent the premature convergence and improve the convergence speed and accurateness.

## 1. Introduction

Firefly algorithm (FA) is inspired by biochemical and social aspects of real fireflies [1]. It could handle multimodal problems of combinational and numerical optimization more naturally and efficiently [2–5]. Owing to its few parameters to adjust, easy to understand, realize, and compute, it was applied to various fields, such as codebook of vector quantization [6], in-line spring-mass systems [7], mixed variable structural optimization [8], nonlinear grayscale image enhancement [9], travelling salesman problems [10], continuously cast steel slabs [11], promoting products online [12], nonconvex economic dispatch problems [13], chiller loading for energy conservation [14], stock market price forecasting [15], and multiple objectives optimization [16].

Despite these advantages, the FA is also a metaheuristic algorithm; the standard FA can easily get trapped in the local optima when solving complex multimodal problems. These weaknesses have restricted wider applications of the FA. Therefore, avoiding the local optima and accelerating convergence speed have become the two most important and appealing goals in the FA research. To overcome these disadvantages, many researchers have proposed a variety of modifications to the original FA [17–19].

Compared with other evolutionary algorithms, such as Genetic Algorithm and Simulated Annealing, standard FA has the following problem: it is not rational that each firefly uses the same step or the linear step just depends on maximum iteration not related to experience of fireflies, which may impact on the balance between the global and local search. Based on the above problem, a self-adaptive step firefly algorithm (SASFA) is proposed in the paper, which considers the historical information and current situation of each firefly.

The rest of this paper is organized as follows. Section 2 shows a brief review of the updating process of the standard FA and analyzes some problems about the linear step approach. In Section 3, a novel approach is proposed to set the step of each firefly self-adaptively. In Section 4, experimental settings and results compared with the two algorithms are presented. Finally, we make the conclusions in Section 5.

## 2. Firefly Algorithm Concepts

Firefly algorithm is based on the idealized behavior of the flashing feature of fireflies. The following three rules are idealized for the basic formulation of FA: (1) all fireflies are unisex so that fireflies will attract each other regardless of

```

Begin
Objective function  $f(x), x = (x_1, \dots, x_d)^T$ 
Generate initial population of  $n$  fireflies  $x_i, i = 1, 2, \dots, n$ 
Formulate light intensity  $I$  so that it is associated with  $f(x)$ 
While ( $t < \text{MaxGeneration}$ )
  Define absorption coefficient  $\gamma$ 
  for  $i = 1 : n$  ( $n$  fireflies)
    for  $j = 1 : n$  ( $n$  fireflies)
      if ( $I_j > I_i$ ),
        move firefly  $i$  towards  $j$ 
      end if
      Vary attractiveness with distance  $r$  via  $\exp(-\gamma r^2)$ 
      Evaluate new solutions and update light intensity
    end for  $j$ 
  end for  $i$ 
  Rank the fireflies and find the current best
end while
Post-processing the results and visualization
end

```

PSEUDOCODE 1: Pseudocode of the FA.

their sex; (2) attractiveness is proportional to their brightness, which decreases as distance increases between two flies; (3) the brightness of a firefly is determined by the objective function [1].

**2.1. Standard Firefly Algorithm.** In the FA, there are two critical issues: the formulation of the attractiveness and the variation of light intensity. We can always suppose that a firefly's attractiveness is determined by its light intensity or brightness, which in turn is associated with the encoded objective function [5]. The light intensity  $I(r)$  varies with the distance  $r$  exponentially and monotonically. That is,

$$I = I_0 e^{-\gamma r}, \quad (1)$$

where  $I_0$  is the initial light intensity and  $\gamma$  is the light absorption coefficient. As a firefly's attractiveness is proportional to the light intensity seen by neighbor fireflies, we can now define the attractiveness  $\beta$  of a firefly by

$$\beta = \beta_0 e^{-\gamma r^2}, \quad (2)$$

where  $\beta_0$  is the attractiveness at  $r = 0$ . The movement of a firefly  $i$  is attracted to another more attractive firefly  $j$ , which is determined by

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \epsilon. \quad (3)$$

The third term is the randomization with the step  $\alpha$ , being drawn from a Gaussian distribution. In [2], the Levy distribution was used instead of the Gaussian one. Based on comments in [1] the implementation of standard FA was adopted  $\beta_0 = 1$  and  $\alpha$  generated with uniform

distribution in range  $[0, 1]$ . The procedure for implementing the FA can be summarized as the pseudocode revealed in Pseudocode 1.

### 2.2. Problems of Setting Step in Standard Firefly Algorithm.

In standard FA, the third term of (3) is the randomization with step  $\alpha$ . It generates with uniform distribution in range  $[0, 1]$ . In general, the method of setting step is static or linear; step just depends on maximum generation, which cannot be adaptive to different fireflies. Firstly, each firefly has the same step settings, and its value decreases from iteration to iteration. Consequently, it may trap in the local optimum, resulting in premature convergence. Secondly, a large step can make the firefly skip the best optimal solution when it is in the neighborhood of the firefly, during the early stage of the search. Therefore, this may reduce the search performance.

Finally, we can see from (3), that it is helpful for firefly to explore new search space with a large step, but it is not useful to the convergence of global optimum. If step has a small value, the result is contrary. For this reason, the step  $\alpha$  has a great effect on the exploration and convergence of the algorithm. The value of  $\alpha$  decreases slowly with the large number of iterations but decreases rapidly with the small number of iterations. To take into account the above issues, a new method based on self-adaptive step is introduced which considers the respective experience of each firefly.

## 3. Proposed Firefly Algorithm

When we make a decision, two important messages are usually integrated, which is known by Boyd and Richerson, on the exploration of human decision-making process. One

TABLE 1: Benchmark functions.

Functions	Formulations	Limits
$f_1$	$\min f(x) = e^{-(x_1-4)^2-(x_2-4)^2} + e^{-(x_1+4)^2-(x_2-4)^2} + 2e^{-x_1^2-(x_2+4)^2} + 2e^{-x_1^2-x_2^2}$	$x_i \in [-5, 5]$
$f_2$	$\min f(x) = ( x_1  - 5)^2 + ( x_2  + 5)^2$	$x_i \in [-10, 10]$
$f_3$	$\min f(x) = 100(x_2 - x_1)^2 + (6.4(x_2 - 0.5)^2 - x_1 - 0.6)^2$	$x_i \in [-5, 5]$
$f_4$	$\min f(x) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$	$x_i \in [-2.048, 2.048]$
$f_5$	$\min f(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	$x_i \in [-5, 15]$
$f_6$	$\min f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2)$	$x_i \in [-1, 1]$
$f_7$	$\min f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1x_2 - 5)^2$	$x_i \in [-10, 10]$
$f_8$	$\min f(x) = 4x_1^2 + 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_1^2 + 4x_1^4$	$x_i \in [-5, 5]$
$f_9$	$\min f(x) = \{1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_1^2)\}$ $\{30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\}$	$x_i \in [-2, 2]$
$f_{10}$	$\min f(x) = \frac{\sin(\sqrt{x_1^2 + x_2^2} - 0.5)}{1 + 0.001(x_1^2 + x_2^2)^2} - 0.5$	$x_i \in [-10, 10]$
$f_{11}$	$\max f(x) = -(20 + (x^2 - 10 \cos(2\pi x_1)) + (x_2^2 - 10 \cos(2\pi x_2)))$	$x_i \in [-2.048, 2.048]$
$f_{12}$	$\max f(x) = \cos x_1 \cos x_2 e^{-(x_1-\pi)^2-(x_2-\pi)^2}$	$x_i \in [-20, 20]$
$f_{13}$	$\max f(x) = 280 - \frac{(x_1^4 - 16x_1^2 + 5x_1)}{2} - \frac{(x_2^4 - 16x_2^2 + 5x_2)}{2}$	$x_i \in [-5, 5]$
$f_{14}$	$\max f(x) = \ln\{[(\sin(\cos x_1 + \cos x_2))^2 - (\cos(\sin x_1 + \sin x_2))^2 + x_1]^2\} - 0.1((x_1 - 1)^2 + (x_2 - 1)^2)$	$x_i \in [-10, 10]$
$f_{15}$	$\max f(x) = -\left[\sum_{i=1}^5 i \cos(i+1)x_1 + i\right] \left[\sum_{i=1}^5 i \cos(i \cos(i+1)x_2 + i)\right]$	$x_i \in [-10, 10]$
$f_{16}$	$\max f(x) = 660 - (x_1^2 + x_2 - 11)^2 - (x_1 + x_2^2 - 7)^2$	$x_i \in [-6, 6]$

is the experience of themselves and their neighbors. The other is the current situation. Inspired by this idea, we use it to set the step of each firefly to guide its search in the search space [20].

**3.1. Self-Adaptive Step Settings.** To solve the problems mentioned in Section 2, step settings should vary with the different problems, fireflies, and environment. Based on these considerations, a method is presented that sets the step of each firefly according to its respective experience and current situation. The step of firefly near the optimal solution should be set small. Similarly, the step of firefly far away from the optimal solution should be set large. Fireflies between the above two are used to balance the global search and local search. Therefore, the step of firefly should also be concerned with its historical information and current situation. In this paper, firefly's history information includes its past two iteration's optimum value. Based on the comments mentioned above and many experiments, the step  $\alpha$  of each firefly is calculated by (4) and (5), respectively [21]. Consider

$$h_i(t) = \frac{1}{\sqrt{(f_{pi}(t-1) - f_{pi}(t-2))^2 + 1}}, \quad (4)$$

$$\alpha_i(t+1) = 1 - \frac{1}{\sqrt{(f_{best}(t) - f_i(t))^2 + h_i(t)^2 + 1}}, \quad (5)$$

where  $h_i(t)$  is the past two iterations' history information of  $i$ th firefly.  $f_{pi}$  is the fitness value of the best solution of  $i$ th firefly.  $f_{best}$  is the fitness value of the best solution of population heretofore found, and  $f_i$  is the fitness value of  $i$ th firefly, which reflects the current information. The next iteration step of each firefly is self-adaptive, determined by the difference between its current fitness value and the best fitness value of the population. So the step of each firefly can vary with the iteration, and the step of each firefly is also altered at the same iteration. Obviously, the step of each firefly is different for various problems, because different fitness functions are used.

**3.2. Procedure of Self-Adaptive Step Firefly Algorithm.** The implementation procedure of our proposed self-adaptive step firefly algorithm (SASFA) can be described as follows.

*Step 1.* Generate the initial population of fireflies,  $\{x_1, x_2, \dots, x_n\}$ .

*Step 2.* Compute intensity for each firefly member,  $\{I_1, I_2, \dots, I_n\}$ .

*Step 3.* Update the step of each firefly. The step is calculated by (4) and (5).

*Step 4.* Move each firefly  $i$  towards other brighter fireflies. The position of each firefly is updated by (3).

TABLE 2: Comparison of standard FA and SASFA.

Functions	Optimization method	Best solution	Worst solution	Medium of solutions	Standard deviation
$f_1$	Standard FA	1.017552E - 12	9.648893E - 11	3.065379E - 12	9.930156E - 12
	SASFA	1.017595E - 12	1.043827E - 12	1.026483E - 12	5.723552E - 15
$f_2$	Standard FA	2.500005E + 01	4.339859E + 01	2.647969E + 01	2.582632E + 00
	SASFA	2.500019E + 01	2.501609E + 01	2.500482E + 01	3.358280E - 03
$f_3$	Standard FA	1.345107E - 12	5.080286E + 01	5.080286E - 01	5.080286E + 00
	SASFA	1.640501E - 14	5.474220E - 03	1.074990E - 03	1.152379E - 03
$f_4$	Standard FA	1.085659E - 13	1.288528E + 00	6.863843E - 02	1.597118E - 01
	SASFA	4.991661E - 08	4.744487E - 04	2.186426E - 05	5.391553E - 05
$f_5$	Standard FA	3.978874E - 01	2.024074E + 00	4.245216E - 01	1.715188E - 01
	SASFA	3.978882E - 01	3.980451E - 01	3.979238E - 01	3.732067E - 05
$f_6$	Standard FA	-2.000000E + 00	-1.757801E + 00	-1.979413E + 00	4.885083E - 02
	SASFA	-2.000000E + 00	-1.859492E + 00	-1.996340E + 00	1.833774E - 02
$f_7$	Standard FA	1.975868E - 13	4.346346E + 01	5.078575E + 00	8.627606E + 00
	SASFA	5.542967E - 14	8.963713E - 04	9.258293E - 05	1.227339E - 04
$f_8$	Standard FA	-1.031628E + 00	-9.940826E - 01	-1.031253E + 00	3.754586E - 03
	SASFA	-1.031628E + 00	-1.031575E + 00	-1.031615E + 00	1.269080E - 05
$f_9$	Standard FA	3.000000E + 00	8.400000E + 01	8.065019E + 00	1.289350E + 01
	SASFA	3.000000E + 00	3.017131E + 00	3.001386E + 00	2.768176E - 03
$f_{10}$	Standard FA	7.283618E - 13	1.674303E - 02	8.079304E - 03	3.942527E - 03
	SASFA	9.985832E - 08	9.716095E - 03	7.439098E - 03	4.036846E - 03
$f_{11}$	Standard FA	-9.949591E - 01	-9.949591E - 01	-9.949591E - 01	1.673724E - 15
	SASFA	-2.844390E - 04	-2.844390E - 04	-2.844390E - 04	2.179328E - 19
$f_{12}$	Standard FA	1.000000E + 00	2.326942E - 42	8.127414E - 02	2.306928E - 01
	SASFA	1.000000E + 00	7.680531E - 05	7.687526E - 01	4.105102E - 01
$f_{13}$	Standard FA	3.583323E + 02	3.441956E + 02	3.581890E + 02	1.413611E + 00
	SASFA	3.583323E + 02	3.583305E + 02	3.583320E + 02	4.197847E - 04
$f_{14}$	Standard FA	2.205095E + 00	1.200545E + 00	1.820786E + 00	2.814797E - 01
	SASFA	2.205094E + 00	1.776040E + 00	2.195012E + 00	6.151606E - 02
$f_{15}$	Standard FA	1.867309E + 02	4.712473E + 01	1.628826E + 02	4.008809E + 01
	SASFA	1.867308E + 02	1.834636E + 02	1.863022E + 02	5.283146E - 01
$f_{16}$	Standard FA	6.600000E + 02	6.600000E + 02	6.600000E + 02	1.771468E - 11
	SASFA	6.600000E + 02	6.599976E + 02	6.599997E + 02	4.307035E - 04

Step 5. Update the solution set.

Step 6. Terminate if a termination criterion is fulfilled; otherwise go to Step 2.

#### 4. Experiments and Results

The proposed SASFA and standard FA are tested on sixteen benchmark functions which are given in Table 1. All simulations are run in Matlab 2010b with 2 GB of random access memory. To eliminate stochastic discrepancy, in each case study, it adopted 100 independent runs for each of the optimization methods, involving 100 different initial trial solutions for each optimization method. The number of

fireflies was 30, and the maximum iteration number was 300.

Table 2 shows the best solution, the worst solution, the medium of solutions, and the standard deviation of 100 runs. Medium of solutions and standard deviation of 100 runs are considered to measure the scalability. Table 2 reveals the striking potential of the SASFA in obtaining the optimal solution with high precision, and the solutions of SASFA are better than the solutions found by standard FA.

Figures 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, and 16 give the comparison of convergence processes of the SASFA and standard FA in the above sixteen benchmark functions. It can be found out that SASFA outperformed standard FA with the same parameter settings. Convergence curves depict

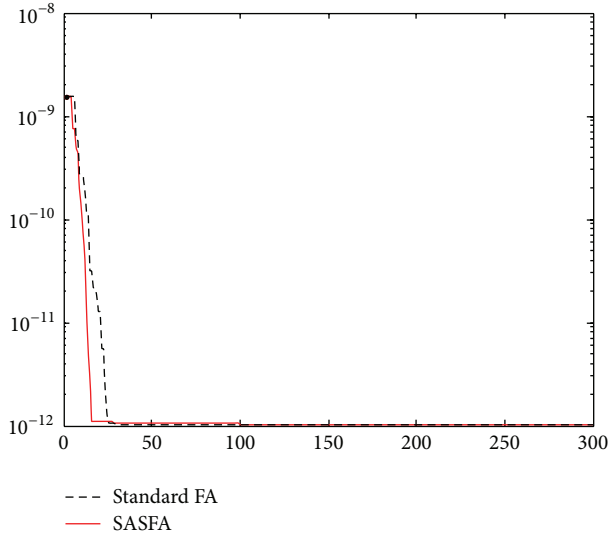


FIGURE 1: Comparison of curve graph for  $f_1$ .

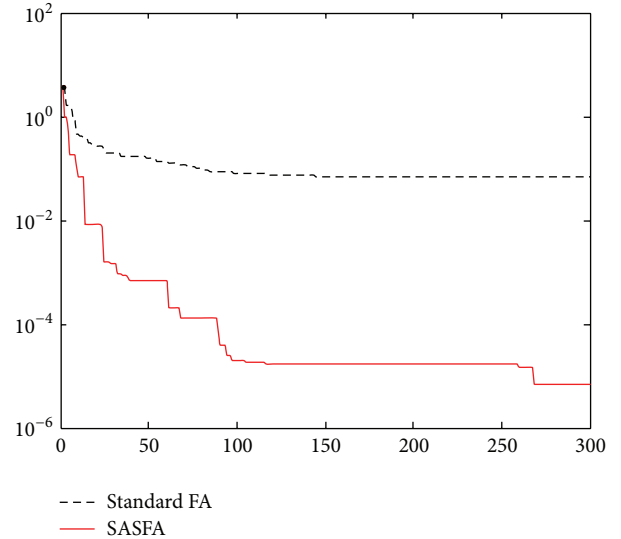


FIGURE 4: Comparison of curve graph for  $f_4$ .

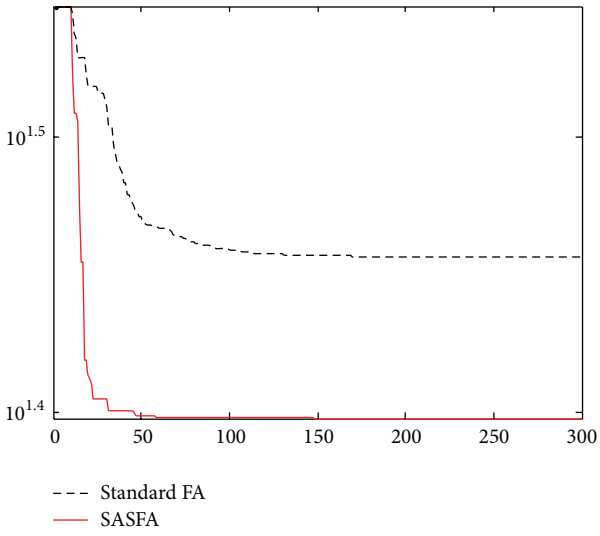


FIGURE 2: Comparison of curve graph for  $f_2$ .

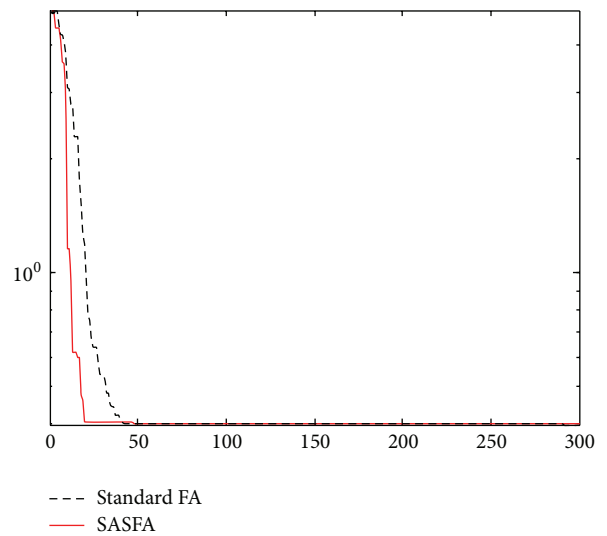


FIGURE 5: Comparison of curve graph for  $f_5$ .

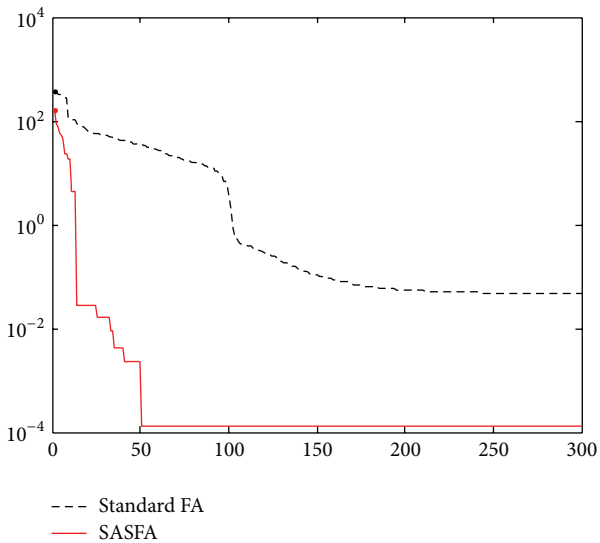


FIGURE 3: Comparison of curve graph for  $f_3$ .

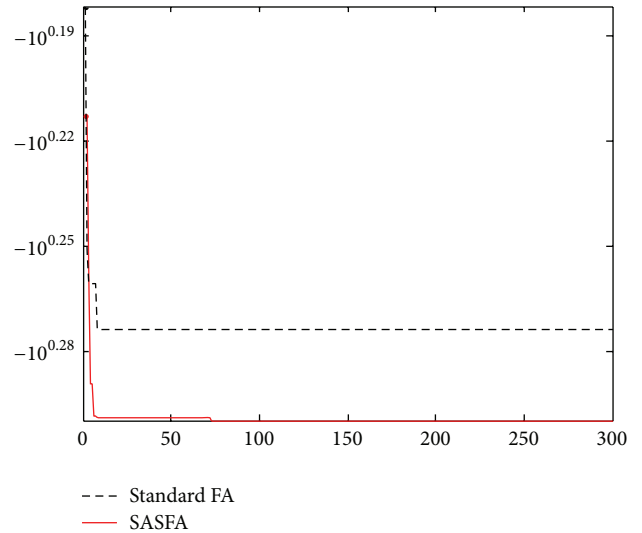


FIGURE 6: Comparison of curve graph for  $f_6$ .

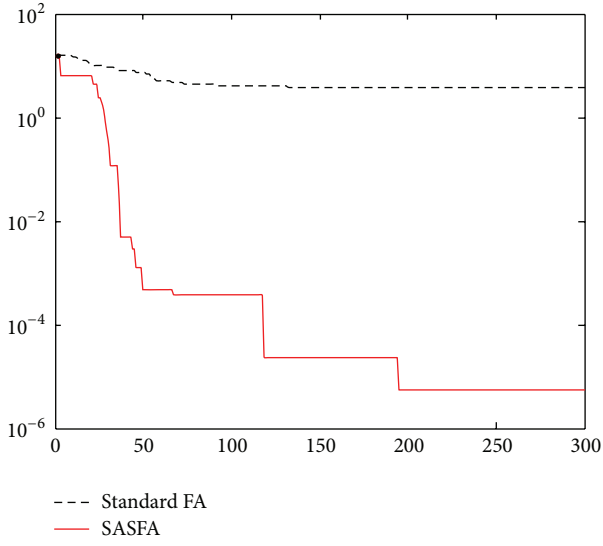


FIGURE 7: Comparison of curve graph for  $f_7$ .

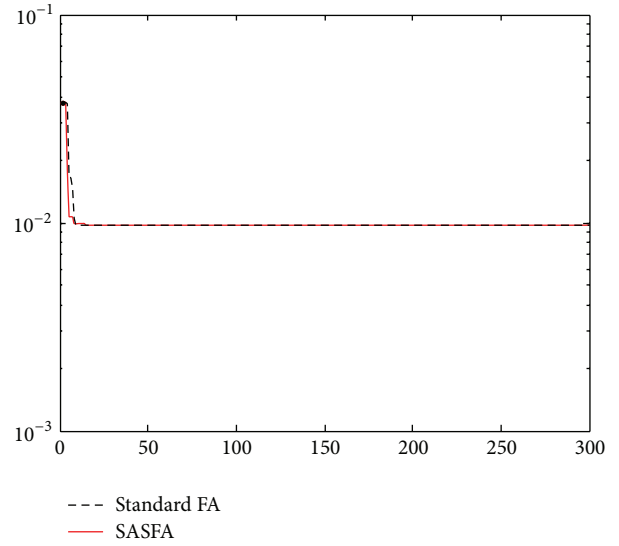


FIGURE 10: Comparison of curve graph for  $f_{10}$ .

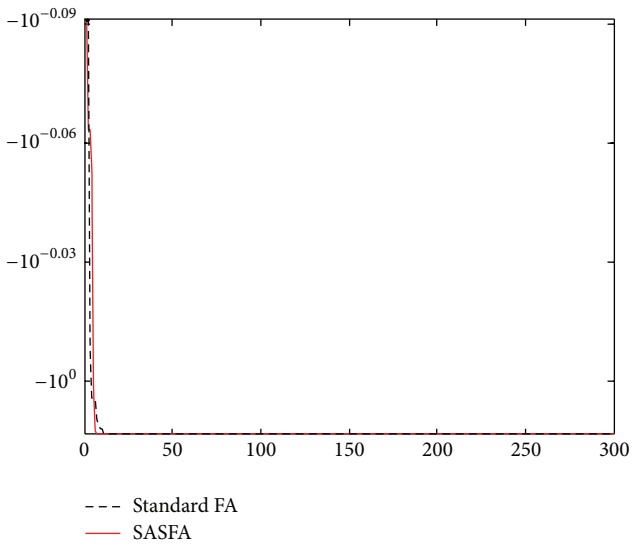


FIGURE 8: Comparison of curve graph for  $f_8$ .

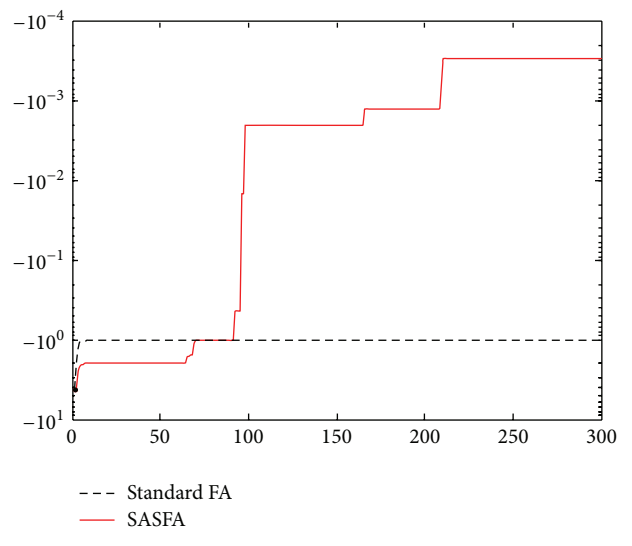


FIGURE 11: Comparison of curve graph for  $f_{11}$ .

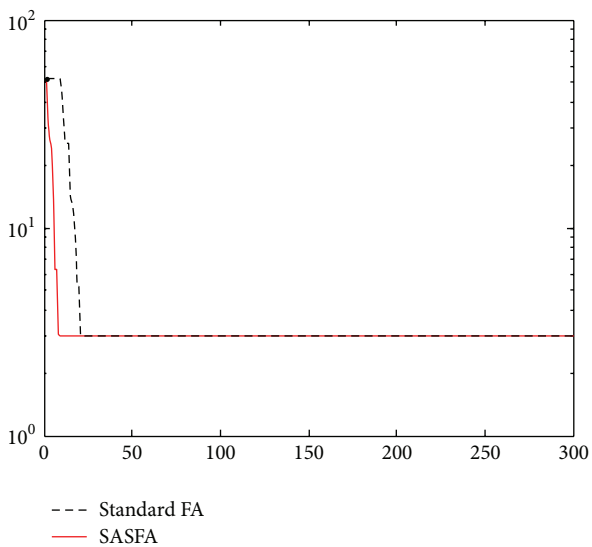


FIGURE 9: Comparison of curve graph for  $f_9$ .

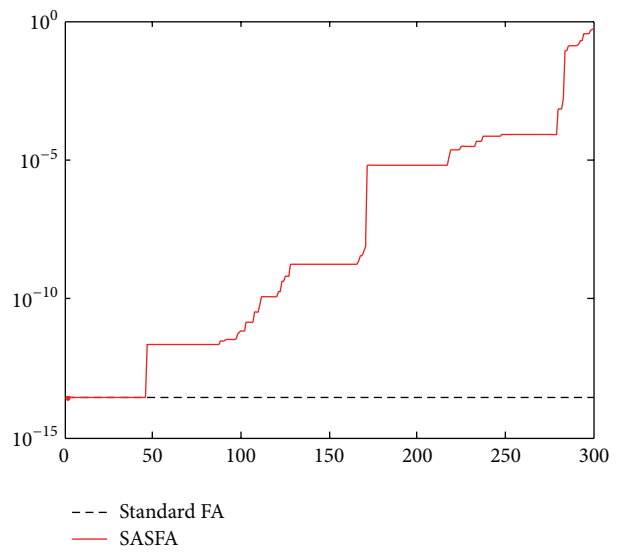


FIGURE 12: Comparison of curve graph for  $f_{12}$ .

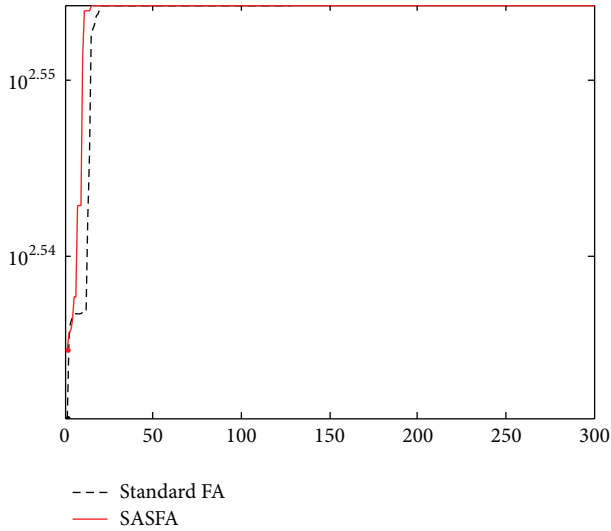


FIGURE 13: Comparison of curve graph for  $f_{13}$ .

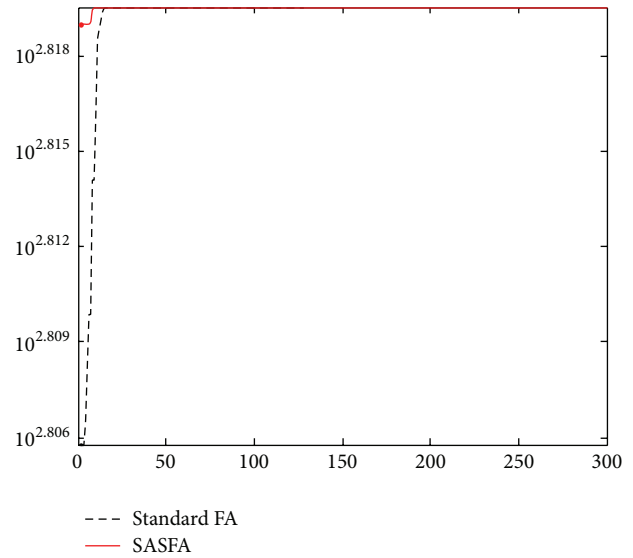


FIGURE 16: Comparison of curve graph for  $f_{16}$ .

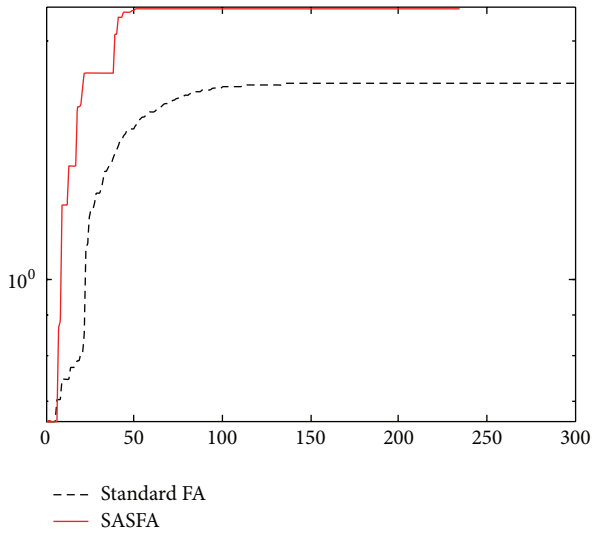


FIGURE 14: Comparison of curve graph for  $f_{14}$ .

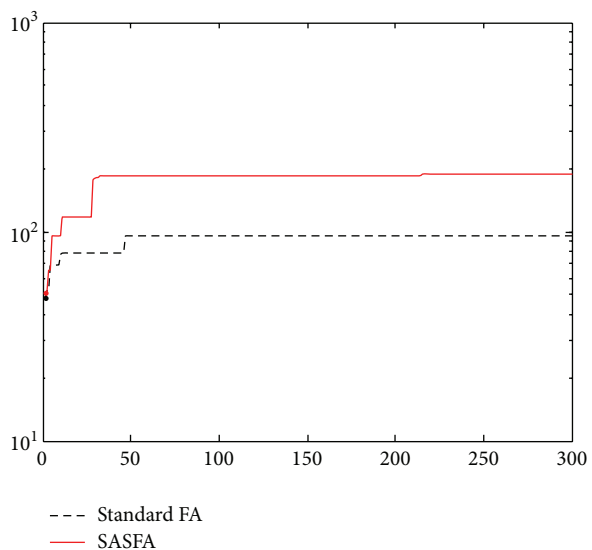


FIGURE 15: Comparison of curve graph for  $f_{15}$ .

the provision trade-off between exploration and exploitation of the SASFA during the search process.

### 5. Conclusions and Further Research

In the paper, we have proposed a self-adaptive step firefly algorithm which considers the current situation and historical information of each firefly. Simulation results demonstrated the potential of the proposed algorithm. Considering more iteration's information of the algorithm could be an exciting direction in future.

### Acknowledgments

This research is financially supported by the National Natural Science Foundation of China (NSFC) for Professor Shanlin Yang (no. 71071045) and Professor Shoubao Su (no. 61075049) and the Universities natural science foundation of anhui province (nos. KJ2011A268 and KJ2012Z429). The authors of the paper express great acknowledgment for these supports.

### References

- [1] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, Frome, UK, 2008.
- [2] X.-S. Yang, "Firefly algorithm, lévy flights and global optimization," in *Research and Development in Intelligent Systems*, vol. 26, pp. 209–218, Springer, Berlin, Germany, 2010.
- [3] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *Stochastic Algorithms: Foundations and Applications*, vol. 5792 of *Lecture Notes in Computer Science*, pp. 169–178, Springer, Berlin, Germany, 2009.
- [4] S. Łukasik and S. Zak, "Firefly algorithm for continuous constrained optimization tasks," in *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems*, vol. 5796 of *Lecture Notes in Computer Science*, pp. 97–106, Springer, Heidelberg, Germany, 2009.

- [5] X. S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *International Journal of Bio-Inspired Computation*, vol. 2, no. 2, pp. 78–84, 2010.
- [6] M.-H. Horng and T.-W. Jiang, "The codebook design of image vector quantization based on the firefly algorithm," in *Computational Collective Intelligence. Technologies and Applications*, vol. 6423 of *Lecture Notes in Computer Science*, pp. 438–447, Springer, Heidelberg, Germany, 2010.
- [7] R. Dutta, R. Ganguli, and V. Mani, "Exploring isospectral spring-mass systems with firefly algorithm," *Proceedings of the Royal Society A*, vol. 467, no. 2135, pp. 3222–3240, 2011.
- [8] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using firefly algorithm," *Computers and Structures*, vol. 89, no. 23–24, pp. 2325–2336, 2011.
- [9] T. Hassanzadeh, H. Vojodi, and F. Mahmoudi, "Non-linear grayscale image enhancement based on firefly algorithm," in *Swarm, Evolutionary, and Memetic Computing*, vol. 7077 of *Lecture Notes in Computer Science*, pp. 174–181, Springer, Heidelberg, Germany, 2011.
- [10] G. K. Jati and Suyanto, "Evolutionary discrete firefly algorithm for travelling salesman problem," in *Adaptive and Intelligent Systems*, vol. 6943 of *Lecture Notes in Computer Science*, pp. 393–403, Springer, Heidelberg, Germany, 2011.
- [11] T. Mauder, C. Sandera, J. Stetina, and M. Seda, "Optimization of the quality of continuously cast steel slabs using the firefly algorithm," *Materiali in Tehnologije*, vol. 45, no. 4, pp. 347–350, 2011.
- [12] H. Banati and M. Bajaj, "Promoting products online using firefly algorithm," in *Proceedings of the 12th International Conference on Intelligent Systems Design and Applications (Isda)*, pp. 580–585, 2012.
- [13] X.-S. Yang, S. S. S. Hosseini, and A. H. Gandomi, "Firefly Algorithm for solving non-convex economic dispatch problems with valve loading effect," *Applied Soft Computing Journal*, vol. 12, no. 3, pp. 1180–1186, 2012.
- [14] L. D. Coelho and V. C. Mariani, "Improved firefly algorithm approach applied to chiller loading for energy conservation," *Energ Buildings*, vol. 59, pp. 273–278, 2013.
- [15] A. Kazem, E. Sharifi, F. K. Hussain, M. Saberi, and O. K. Hussain, "Support vector regression with chaos-based firefly algorithm for stock market price forecasting," *Applied Soft Computing*, vol. 13, no. 2, pp. 947–958, 2013.
- [16] X. S. Yang, "Multiobjective firefly algorithm for continuous optimization," *Engineering with Computers*, vol. 29, no. 2, pp. 175–184, 2013.
- [17] Y. F. Tian, W. M. Gao, and S. Yan, "An improved inertia weight firefly optimization algorithm and application," in *International Conference on Control Engineering and Communication Technology (ICCECT '12)*, pp. 64–68, 2012.
- [18] S. L. Tilahun and H. C. Ong, "Modified firefly algorithm," *Journal of Applied Mathematics*, vol. 2012, Article ID 467631, 12 pages, 2012.
- [19] A. H. Gandomi, X.-S. Yang, S. Talatahari, and A. H. Alavi, "Firefly algorithm with chaos," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 89–98, 2013.
- [20] R. Boyd and P. J. Richerson, *Culture and the Evolutionary Process*, University of Chicago Press, 1988.
- [21] H. T. Geng, Y. H. Huang, J. Gao, and H. F. Zhu, "A self-guided particle swarm optimization with independent dynamic inertia weights setting on each particle," *Applied Mathematics & Information Sciences*, vol. 7, no. 2, pp. 545–552, 2013.