Comparisons of Firefly Algorithm with Chaotic Maps

Shoubao Su[1], Yu Su[2], Mingjuan Xu[3*]

[1] *School of Computer Engineering, Jinling Institute of Technology, Nanjing 211169, China;*

[2] *School of Information & Software Engineering, University of Electronic Science &Technology of China, Chengdu 610054, China;*

[3] *School of Information Engineering, West Anhui University, Lu'an 237012, P.R. China*

Received 1 November 2014, www.cmnt.lv

Abstract

Firefly Algorithm (FA) is one of the new bio-inspired algorithm driven by the simulation of the flashing behavior of fireflies. To deal with the problems of low accuracy and local convergence in standard FA, the chaos theory is introduced into the evolutionary process of FA. Since chaotic mapping has certainty, ergodicity and stochastic property, by initializing the population of fireflies and replacing the constant value of absorption coefficient with four chaotic maps, the proposed FA increases its convergence rate and resulting precision. Comparisons experimentally show that convergence quality and accuracy are improved, which testify that modified FA with chaos is valid and feasible.

Keywords: Firefly algorithm (FA), Chaotic map, biological computing, ergodicity and stochastic property.

1 Introduction

Metaheuristic algorithms are optimization algorithms which attempt to better the quality of solution members iteratively with some random characters. Majority of these algorithms are inspired by biological behavior. Unlike deterministic solution methods, metaheuristic algorithms are not affected by the behavior of the optimization problem[1]. So this property makes the algorithm to be used widely in different fields[2]. The firefly algorithm has become an increasingly important method of Swarm Intelligence that has been applied in almost all areas of optimazation, as well as engineering practice. Many problems from various areas have been successfully solved using the firefly algorithm and its variants. The FA is based on the idealized behavior of the flashing features of fireflies. Preliminary studies show that the FA can perform superiorly, compared with genetic algorithm and particle swarm optimization, and it is applicable for mixed variable and engineering optimization[3].

Recently, the new idea of applying chaotic systems to stochastic processes in optimization algorithms has been noticed by many researchers. In random-based and stochastic optimization algorithms, the role of randomness can be played by a chaotic dynamics[6-11]. In Ref. [8] different chaotic maps are utilized to tune the attractive movement of the fireflies in the algorithm and Ref. [12] introduces a chaos-enhanced firefly algorithm with automatic parameter tuning. In Ref. [2], researchers introduce a modified FA approach combined with chaotic sequences applied to reliability-redundancy optimization. Experimental studies assert that the benefits of using chaotic signals instead of random signals are often evident although it is not mathematically proved yet [13].

In this paper, we use four chaotic maps to initialize the population of fireflies and replace the constant value of absorption coefficient γ. Simulation is also done on sixteen benchmark functions. From the simulation result, it is shown that the modified firefly algorithm with chaos outperforms the standard firefly algorithm. The structure of this paper is organized as follows. In Section 2, firefly algorithm and chaotic map will be presented. The details of the proposed algorithm with chaos are explained in Section 3, and the simulation results will be presented In Section 4. Finally, conclusions are demonstrated in Section 5.

2 Firefly algorithm and chaotic map

2.1 FIREFLY ALGORITHM

The firefly algorithm is inspired by the flashing behavior of fireflies. According to Yang[10], there are three assumptions in firefly algorithm:

1) All fireflies are unisexual and every firefly attracts attracted to every other firefly;
2) Attractiveness is proportional to a firefly's brightness. For any two fireflies, the less bright one will be attracted by the brighter one, and the brightness will reduce as their distance increases;
3) If there are no fireflies brighter than a given firefly, it will move randomly.

As we know the light intensity *I(r)* is inversely proportional to the distance *r* from the light source. Therefore when light passes through a medium with light absorption coefficient of γ, *I(r)* varies with distance *r* as given below:

$$I(r) = I_0 e^{-\gamma r^2} \quad , \tag{1}$$

where I_0 is the intensity at the point of source. Because computationally computing $1/(1+\gamma r^2)$ is easier than $e^{-\gamma r^2}$. So *I(r)* can be calculated as follows[8]:

$$I(r) = \frac{I_0}{\gamma r^2} \quad . \tag{2}$$

* ***Corresponding author's*** e-mail: xmj8217@wxc.edu.cn (M.J. Xu).

Each firefly has its distinctive attractiveness β can be defined as below:

$$\beta(r) = \beta_0 e^{-\gamma r^2}, \tag{3}$$

Similarly it can be defined as follows:

$$\beta(r) = \frac{\beta_0}{1+\gamma r^2}, \tag{4}$$

where β_0 is the attractiveness at r =0.

The firefly located at x_i movement is attracted to another more brighter firefly located at x_j is determined by equation (5):

$$x_{i+1} = x_i + \beta_0 e^{-\gamma r^2}(x_j - x_i) + \alpha\varepsilon \ . \tag{5}$$

The second item is attributing to the attraction, and the third item is randomization with $\alpha(0\le\alpha\le1)$ and ε. For most practical problems, we can use a constant value of α=0.2. Here ε is a vector of random variables being derived from a Gaussian distribution. In Ref. [10] Levy distribution was used instead of Gaussian one.

2.2 CHAOTIC MAP

Chaos is a stochastic motion mapped by the deterministic equation and it is different from the phenomenon of iregularity and disorder. Chaos has a fine internal structure. It has three characters: random, ergodic and regularity. Ergodic property can search all states by its formulas within certain range. So chaos became a available strategy to avoid being trapped in local optima and improve the quality of searching global optimum[8]. To utilize this advantage, this paper initializes the population of fireflies and replaces the constant value of absorption coefficient with chaotic maps. Four well-known maps as follows.

1) Logistic map

$$x_{i+1} = \mu x_i (1 - x_i), \tag{6}$$

here μ is a control parameter. When $\mu = 4, 0 \le \mu_0 \le 1$, logistic is totally in a chaotic state [11]. In this paper, μ assigns 4.

2) Tent map. The tent map is very similar to the logistic map. This map is defined by the following equation[7]

$$x_{i+1} = \begin{cases} \dfrac{x_i}{0.7} & x_i < 0.7 \\ \dfrac{10}{3}(1 - x_i) & x_i \ge 0.7 \end{cases}, \tag{7}$$

3) Iterative map. The iterative chaotic map with infinite collapses is expressed by equation[8]

$$x_{i+1} = \sin\left(\frac{a\pi}{x_i}\right), \tag{8}$$

where $a \in (0,1)$ is a suitable parameter.

4) Gauss map. The Gauss map (also known as Gaussian map or mouse map) given by the Gaussian function [9]

$$x_{i+1} = \exp(-\alpha x_i^2). \tag{9}$$

3 Propsed firefly algorithm

When firefly algorithm is used to optimize the multi-peak function, it can be easily trapped in the local minima, which leads to slow convergence speed. Moreover, it is difficult to obtain an accurate result without the use of a good searching method. Chaos is a general nonlinear phenomenon in nature which has characteristics of randomness, ergodicity, and regularity because of its exquisite internal structure. This section we present a modified FA with chaos theory to improve the standard FA's convergence quality and precision.

3.1 DEFECTS OF FIREFLY ALGORITHM

Analyzing defects of firefly algorithm in its search process as follows:

1) Initialization process is random. Although random initialization can ensure the initial fireflies distributed homogeneous in the solution space, the quality of solutions is uncertain, because a part of fireflies far away the global optimum. If the initial fireflies are not only distributed homogeneous but also high-quality, it will help to better the mass of fireflies, and prevent algorithm to be prematurely stuck in local optima to some extent.

2) The light absorption coefficient of γ is a constant value. The value of γ determines the attractiveness with all the fireflies. In general $\gamma\in[0,10]$ could be suggested[15], it is more convenient. However, to use fixed value for all the problems is not rational. In fact the absorption coefficient should be varied with the iteration in searching space. Therefore, we proposed here to tune γ with chaotic maps and not use the constant value.

3.2 CHAOTIC FIREFLY ALGORITHM

In stochastic searching optimization algorithms, the methods utilizing chaotic variables instead of random variables are called chaotic optimization algorithm[16]. In these algorithms, due to the nonrepeatability and ergodicity of chaos, it can achieve overall searches at higher speeds than stochastic searches that depend on probabilities[17]. So in this paper we try to use chaotic map to initialize the population of fireflies and tune the absorption coefficient.

```
// Chaotic Firefly Algorithm
Begin
Objective function  f(x),x = (x1.....xd)^T
Generate an initial chaotic population of fireflies
xi,i = 1,2,...n
Formulate light intensity I so that it is associated with f(x)
   While (t < MaxGeneration)
Define absorption coefficient γ with chaos
   For i = 1: n (n fireflies)
      for j = 1: n (n fireflies)
         if (Ij > Ii),
             move firefly i towards j
         end if
         Vary attractiveness with distance r via exp( −γr^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
```

FIGURE 1 Chaotic Firefly Algorithm Pseudo code

On the other hand, from equation (5), it is easy to see that there exist two limiting cases when γ is small and large. When γ trends towards zero, the attractiveness become constant. That is to say, a firefly can be seen by all other fireflies. When γ tends to very large, then the attractiveness decreases remarkably, and all fireflies are short-sightedness. This means all fireflies move nearly casually, which equivalently a stochastic search technique. This may be decrease the precision of the firefly algorithm. So, the proposed chaotic firefly algorithm in this paper, tries to use four famous chaotic maps to initialize the population of fireflies and to tune γ instead of constant values in the equation (5). The equation (5) is modified to

$$x_{i+1} = x_i + \beta_0 e^{-\gamma(t)r^2}(x_j - x_i) + \alpha\varepsilon \ . \tag{10}$$

In this context, $\gamma\ (t)$ is adopted with four mentioned chaotic maps. The pseudo code of chaotic firefly algorithm can be given in Figure 1.

4 Simulation results

The proposed algorithm were tested on sixteen benchmark functions which be given as Table 1. All simulations are run in Matlab 2010b with 2GB RAM. The algorithms used for comparison are 1) Standard firefly algorithm, 2) Proposed algorithm with four chaotic maps. The algorithms were executed with 500 generations and 30 population sizes, $\beta_0 = 1$ as suggested in Ref. [18].

Table 2 shows the best solution, the worst solution, the mean of the solutions and the standard deviation of 100 runs. Mean best fitness and standard deviation are considered to measure the scalability. It is considerable that all of the best solutions are exactly equal the exact solution of the problem. Logistic and Gauss map have better solutions than other maps according to the best solution.

Stronger local search ability of the proposed algorithm with chaos is proven by the improved results of the given functions. Global searching capacity is also represented by the betterment of the results. Due to space limitations, we give nine comparisons of convergence processes of the proposed algorithm with chaos and standard FA in the above benchmark functions (FIGURES 2-9). It can be found out that the proposed FA outperformed standard FA and with the same parameter settings (the algorithms were executed with 100 generations and 30 population sizes).

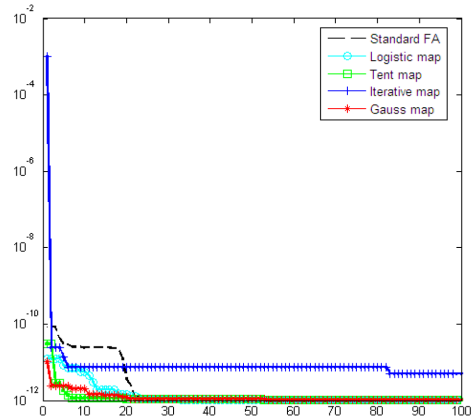

FIGURE 2 Comparison of curve graph for f_1

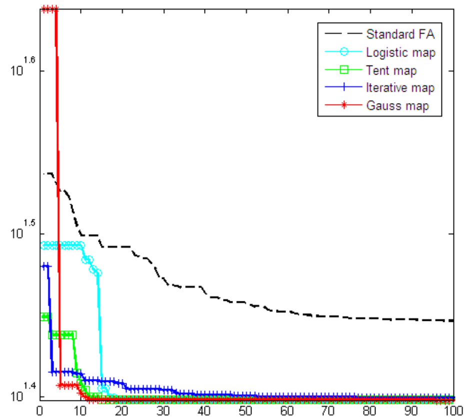

FIGURE 3 Comparison of curve graph for f2

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	$\min f(x) = (\lvert x_1\rvert - 5)^2 + (\lvert x_2\rvert + 5)^2$	$x_i \in [-10,10]$
f_3	$\min f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2)$	$x_i \in [-1,1]$
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	$\min f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1x_2 - 5)^2$	$x_i \in [-10,10]$
f_7	$\min f(x) = 4x_1^2 + 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_1^2 + 4x_1^4$	$x_i \in [-5,5]$
f_8	$\min f(x) = \frac{\sin(\sqrt{x_1^2 + x_2^2} - 0.5)}{1 + 0.001(x_1^2 + x_2^2)^2 - 0.5}$	$x_i \in [-10,10]$

f_9	$\min\ f(x) = \{1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_1+6x_1x_2+3x_1^2)\}$ $\{30+(2x_1-3x_2)^2(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)\}$	$x_i\in[-2,2]$
f_{10}	$\max\ f(x)=660-(x_1^2+x_2-11)^2-(x_1+x_2^2-7)^2$	$x_i\in[-6,6]$
f_{11}	$\max\ f_{15}(x)=\cos x_1\cos x_2 e^{-(x_1-\pi)^2-(x_2-\pi)^2}$	$x_i\in[-20,20]$
f_{12}	$\max\ f(x) = \ln\{[(\sin(\cos x_1+\cos x_2)^2)^2-(\cos(\sin x_1+\sin x_2)^2)^2+x_1]^2\}$ $-0.1((x_1-1)2+(x_2-1)^2)$	$x_i\in[-10,10]$
f_{13}	$\max\ f(x)=-\left[\sum_{i=1}^{5} i\cos(i+1)x_1+i\right]\left[\sum_{i=1}^{5} i\cos((i\cos(i+1)x_2+i)\right]$	$x_i\in[-10,10]$
f_{14}	$\min\ f(x) = \sum_{i=1}^{n} x_i^2$	$x_i \in [-2, 2]$
f_{15}	$\min\ f(x) = \sum_{i=1}^{n} (x_i^2-10\cos(2\pi x_i)+10)$	$x_i \in [-10, 10]$

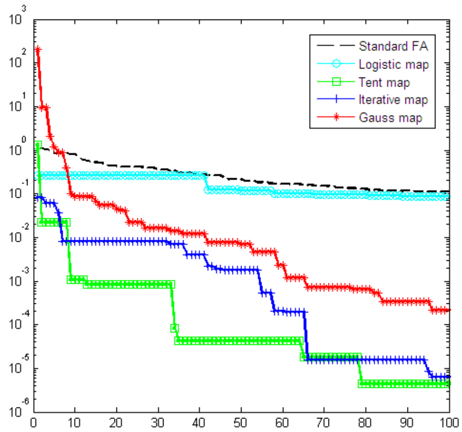

FIGURE 4 Comparison of curve graph for f_4

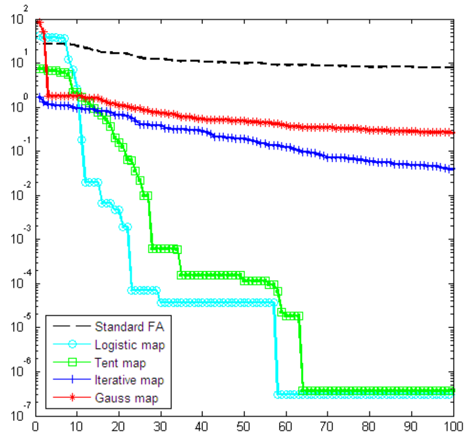

FIGURE 6 Comparison of curve graph for f_6

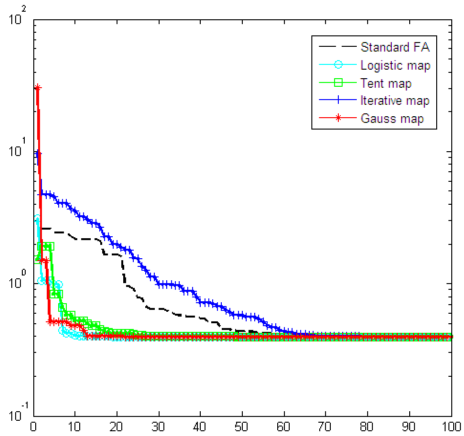

FIGURE 5 Comparison of curve graph for f_5

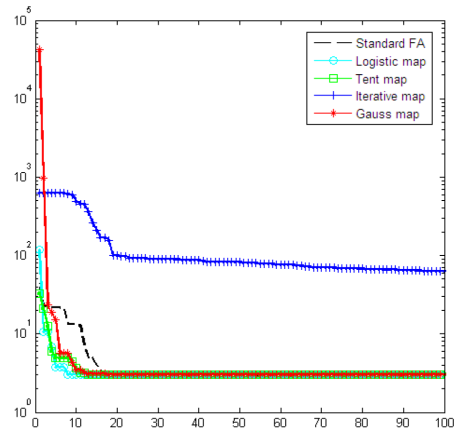

FIGURE 7 Comparison of curve graph for f_9

TABLE 2 Comparison of Standard FA and Chaotic FA

Functions	Optimization Method	Best solution	Worst solution	Medium of solutions	Standard Deviation
f_1	Standard FA	1.0176E-12	6.4556E-11	2.4690E-12	7.6288E-12
	Logistic map	1.0176E-12	1.8155E-11	2.1521E-12	3.0949E-12
	Tent map	1.0176E-12	1.0491E-10	4.8634E-12	1.2216E-11
	Iterative map	1.0851E-12	4.0348E-04	6.2328E-05	1.0876E-04
	Gauss map	1.0176E-12	1.0329E-11	1.7976E-12	1.6285E-12

f_2	Standard FA	2.5000E+01	4.4996E+01	2.6933E+01	3.3282E+00
	Logistic map	2.5000E+01	2.9694E+01	2.5211E+01	6.2300E-01
	Tent map	2.5000E+01	2.5862E+01	2.5039E+01	1.2829E-01
	Iterative map	2.5000E+01	5.7063E+01	3.5034E+01	9.4611E+00
	Gauss map	2.5000E+01	2.7954E+01	2.5247E+01	5.2753E-01
f_3	Standard FA	-2.0000E+00	-2.0000E+00	-2.0000E+00	4.3668E-15
	Logistic map	-2.0000E+00	-2.0000E+00	-2.0000E+00	3.2105E-15
	Tent map	-2.0000E+00	-1.8789E+00	-1.9891E+00	3.4831E-02
	Iterative map	-2.0000E+00	-1.3945E+00	-1.8232E+00	9.0131E-02
	Gauss map	-2.0000E+00	-1.3945E+00	-1.9201E+00	9.7702E-02
f_4	Standard FA	4.4876E-18	1.2602E+00	4.2904E-02	1.3422E-01
	Logistic map	1.0928E-18	6.8244E-01	3.8561E-02	9.5235E-02
	Tent map	6.7522E-18	2.3144E+00	9.7590E-02	3.3412E-01
	Iterative map	1.5032E-18	3.8095E+00	4.6297E-02	3.8240E-01
	Gauss map	4.2402E-18	1.7009E+00	5.9070E-02	2.0013E-01
f_5	Standard FA	3.9789E-01	1.6091E+00	4.2213E-01	1.3534E-01
	Logistic map	3.9789E-01	7.0256E-01	4.0129E-01	3.0604E-02
	Tent map	3.9789E-01	5.4383E+00	5.1643E-01	5.6304E-01
	Iterative map	3.9789E-01	2.3977E+01	1.7885E+00	3.0466E+00
	Gauss map	3.9789E-01	1.7454E+00	4.6988E-01	2.1218E-01
f_6	Standard FA	3.4693E-18	3.2596E+01	4.4325E+00	6.3051E+00
	Logistic map	2.3766E-19	2.6489E+00	8.0357E-02	3.8548E-01
	Tent map	4.5401E-19	1.8151E+00	3.6728E-02	1.8924E-01
	Iterative map	2.5026E-18	1.6343E+01	2.5490E+00	3.4054E+00
	Gauss map	6.1576E-19	9.2337E+00	2.0756E-01	1.0640E+00
f_7	Standard FA	-1.0316E+00	-2.1546E-01	-1.0234E+00	8.1615E-02
	Logistic map	-1.0316E+00	-1.0316E+00	-1.0316E+00	1.5621E-15
	Tent map	-1.0316E+00	-1.0316E+00	-1.0316E+00	1.5621E-15
	Iterative map	-1.0316E+00	-7.4975E-02	-9.8644E-01	1.8559E-01
	Gauss map	-1.0316E+00	-2.1546E-01	-9.7421E-01	2.0923E-01
f_8	Standard FA	-1.4480E+00	-1.4476E+00	-1.4480E+00	4.0389E-05
	Logistic map	-1.4480E+00	-1.4480E+00	-1.4480E+00	2.9011E-15
	Tent map	-9.5885E-01	-8.8878E-01	-8.9286E-01	1.4984E-02
	Iterative map	-9.5885E-01	-8.8877E-01	-8.9051E-01	1.0355E-02
	Gauss map	-9.5885E-01	-8.8878E-01	-8.9439E-01	1.9106E-02
f_9	Standard FA	3.0000E+00	5.3589E+01	9.1263E+00	1.2362E+01
	Logistic map	3.0000E+00	3.0000E+01	6.7289E+00	9.2646E+00
	Tent map	3.0000E+00	3.0000E+01	5.1600E+00	7.3618E+00
	Iterative map	3.0000E+00	8.4000E+02	2.9189E+02	3.6549E+02
	Gauss map	3.0000E+00	8.4000E+02	1.8838E+01	8.4071E+01
f_{10}	Standard FA	6.6000E+02	6.6000E+02	6.6000E+02	0.0000E+00
	Logistic map	6.6000E+02	6.6000E+02	6.6000E+02	0.0000E+00
	Tent map	6.6000E+02	6.6000E+02	6.6000E+02	0.0000E+00
	Iterative map	6.6000E+02	6.6000E+02	6.6000E+02	2.9406E-04
	Gauss map	6.6000E+02	6.4976E+02	6.5984E+02	1.1679E+00
f_{11}	Standard FA	1.0000E+00	9.3803E-04	2.0951E-01	2.9770E-01
	Logistic map	1.0000E+00	1.3566E-01	8.4126E-01	2.8523E-01
	Tent map	1.0000E+00	3.5662E-26	3.4516E-01	4.5263E-01
	Iterative map	1.0000E+00	1.1820E-203	2.1324E-01	3.9122E-01
	Gauss map	1.0000E+00	0.0000E+00	1.1002E-01	3.1446E-01
f_{12}	Standard FA	2.2051E+00	2.0156E+00	2.1694E+00	5.4808E-02
	Logistic map	2.2051E+00	2.2051E+00	2.2051E+00	4.5454E-16
	Tent map	2.2051E+00	1.3026E+00	1.8712E+00	2.7416E-01

	Iterative map	2.2051E+00	1.6587E+00	1.9531E+00	2.6351E-01
	Gauss map	2.2051E+00	-3.7112E+00	8.5499E-01	1.1649E+00
f_{13}	Standard FA	1.8673E+02	5.4405E+01	1.5826E+02	3.9709E+01
	Logistic map	1.8673E+02	2.3137E+01	1.4147E+02	5.1459E+01
	Tent map	1.8673E+02	5.2553E+01	1.6440E+02	3.7499E+01
	Iterative map	1.8673E+02	1.2965E+01	8.6621E+01	5.6661E+01
	Gauss map	1.8673E+02	1.6286E+01	9.1016E+01	6.1511E+01
f_{14}	Standard FA	6.4453E-07	1.7977E-04	1.0845E-05	2.4858E-05
	Logistic map	5.0530E-07	1.7721E-04	8.2073E-06	2.1639E-05
	Tent map	4.1150E-07	1.0832E-04	9.1679E-06	1.8123E-05
	Iterative map	3.8522E-07	3.1273E-04	1.4320E-05	3.7946E-05
	Gauss map	5.1035E-07	4.2150E-04	1.0884E-05	4.2908E-05
f_{15}	Standard FA	6.9673E+00	5.9701E+01	1.8765E+01	7.5423E+00
	Logistic map	6.9697E+00	5.2739E+01	1.9154E+01	8.0053E+00

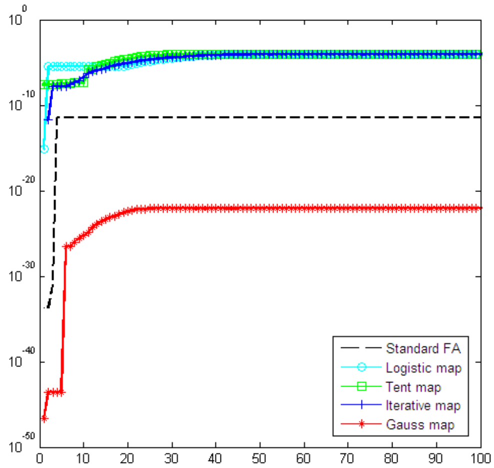

FIGURE 8 Comparison of curve graph for f_{11}

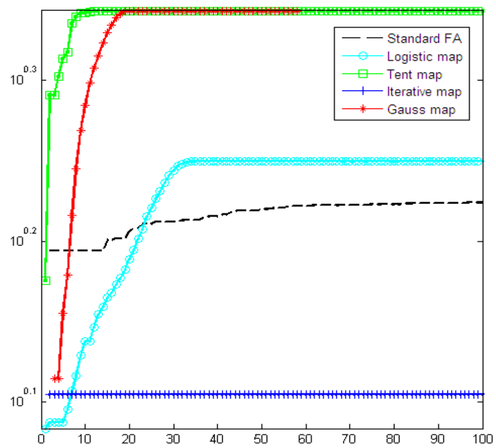

FIGURE 9 Comparison of curve graph for f_{12}

5 Conclusion

This paper proposed modified firefly algorithm with chaos. We use four chaotic maps to initialize the population of fireflies and replace the constant value of absorption coefficient. Simulation is also done on sixteen benchmark functions. From the simulation result, it is shown that the modified firefly algorithm with chaos can increase the quality of initial solutions, avoid being in local optima in a certain extent. Our methods enhance the convergence speed and improve the precision of the solution.

Acknowledgements

This work is supported by the National Natural Science Fundation of China (Nos.61375121, 61075049), the Provincial Project of Natural Scientific Research Fund from the Bureau of Education of Anhui Province, China (No. KJ2011Z401), Anhui Provincial Science Funds for Excellent Youths of Universities (No.2011SQRL150), and sponsored by JIT Scientific Research Program for Introducing Talents (Prof. S.B. Su).

References

[1] Faulin J 2012 Interfaces 42(2), pp 414-415
[2] Coelho LD, Bernert DLD, Mariani VC 2011 A Chaotic Firefly Algorithm Applied to Reliability-Redundancy Optimization: Proc IEEE Congress Evolut Comput(CEC)(New Orleans, Louisiana, USA 5-8 June 2011). pp 517-521
[3] Gandomi H, Yang XS, Alavi AH 2011, Computers and Structures 89(9), 2325-2336
[4] Fister I, Yang XS2013 Swarm and Evolutionary Computation, 13(1), 34–46
[5] Yang XS 2009 Firefly Algorithms for Multimodal Optimization: Proc International Symposium on Stochastic Algorithms, Foundations and Applications (SAGA)(Sapporo, Japan, October 26-28, 2009), Lect Notes Comput Sci,vol.5792, ed Watanabe O, Zeugmann T: Springer Berlin Heidelberg pp169-178
[6] Li MW, Kang HG, Zhou PF, Hong WC 2013 Journal of Systems Engineering and Electronics 24(2), pp 324-334
[7] Gandomi H, Yang XS, Talatahari S, Alavi AH. 2013 Communications in Nonlinear Science and Numerical Simulation 18(1), pp 89-98.
[8] Man J, Nigam MJ 2012 Application of Exponential Atmosphere Concept in Improving Firefly Algorithm: Proc Int Conf Computing Communication & Networking Technologies (Icccnt)(Coimbatore, Tamilnadu, India, 26-28 July 2012), pp 873-882
[9] Huang K, Zhou YQ 2012 A Novel Chaos Glowworm Swarm Optimization Algorithm for Optimization Functions: Bio-Inspired Computing and Applications, vol.6840, pp 426-434
[10] Yang XS 2011 International Journal of Swarm Intelligence Research 2(4), pp 1-11
[11] Yu SH, Su SB, Lu QP, Li H 2014 International Journal of Computer Mathematics 91(5), pp1-7

[12] Yang XS 2010 International Journal of Bio-Inspired Computation 2(1), pp 78-84
[13] Chahkandi V, Yaghoobi M, Veisi G 2013 Nonlinear Dynamics 73(3), pp 475-484
[14] Coelho LdS, Mariani VC 2008 Expert Systems with Applications, 34(10), pp 1905-1913
[15] Lukasik S, Zak S 2009 Firefly Algorithm for Continuous Constrained Optimization Tasks: Proc Computational Collective Intelligence (Wroclaw, Poland, October 5-7, 2009), vol.5796, Lect Notes Comput Sci, ed Kowalczyk R: Springer Berlin Heidelberg pp.97-106

Authors

Shoubao Su

Current position, grades: Ph.D., Professor, CCF senior member. A vice-dean of School of Computer Engineering at Jinling Institute of Technology (JIT), China.
University studies: Ph.D. and M.S. degrees in computer application technology both from Anhui University, China.
Scientific interest: applied swarm intelligent computing to the embedded control optimization and cloud computing.
Experience: research fellow in Nanjing Institutes of Software with studies in big-data mining, where he has served as principal investigators for several projects, including the National Natural Science Foundation of China (NSFC).

Yu Su

Current position, grades: undergraduate student majoring in software engineering (embedded systems), School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, China.
Scientific interest: Swarm intelligence and complex network computing.
Experience: Innovational practices, and has participated in the practical studies of several engineering projects both theoretical analysis and experimental design in weighted networking

Mingjuan Xu

Current position, grades: Lecturer in the School of Information Engineering, West Anhui University, Lu'an, China.
University studies: B.A. degree and the M.A. degree both in computer science from Anhui University of Technology in 2003, and Anhui University in 2009
Scientific interest: Intelligent computing, data mining, and bioinformatics.
Publications: Published several research articles in peer-reviewed journals and international conferences.