Improved Firefly Algorithms Including Slow Fireflies

Masaki Takeuchi
Department of Electrical and Electronic Engineering,
Tokushima University
2-1 Minami-Josanjima,
Tokushima 770–8506, Japan
E-mail: masaki@ee.tokushima-u.ac.jp

Haruna Matsushita
Department of Electronics and Information Engineering,
Kagawa University
2217-20 Hayashi-cho,
Takamatsu, Kagawa, 761-0396, Japan
E-mail: haruna@eng.kagawa-u.ac.jp

Yoko Uwate and Yoshifumi Nishio
Department of Electrical and Electronic Engineering,
Tokushima University
2-1 Minami-Josanjima,
Tokushima 770–8506, Japan
E-mail: {uwate, nishio}@ee.tokushima-u.ac.jp

***Abstract*—Animals having variation among individuals have a greater chance of surviving than those without variation. Therefore, we have proposed a improved Firefly Algorithm existing two type fireflies. In this study, we propose two new Firefly Algorithm existing fireflies flying slow. Numerical experiments indicate that our proposed Firefly Algorithms are superior to the conventional Firefly Algorithm.**

I. Introduction

The solution of optimization problems has recently become increasingly important. Most optimization problems are nonlinear with many constraints. Consequently, optimization algorithms must be efficient to find optimal solutions. Stochastic algorithms, one category of optimization algorithms, are efficient optimization algorithms. Stochastic algorithms have a deterministic component and a random component. Almost all algorithms having only a deterministic component are local search algorithms, for which there is a risk of being trapped at local optima. However, the random component of stochastic algorithms makes it possible to escape from such local optima.

One type of stochastic algorithm is swarm intelligence algorithms, which are based on the behavior of animals and insects. Representative examples are particle swarm optimization (PSO) [1], ant colony optimization (ACO), and the firefly algorithm (FA) [2]–[4].

In the conventional FA, all fireflies are unisex. However, in the real world, there are males and females. Animals having variation among individuals have a greater chance of surviving than those without variation. In the case of solving optimization problems, we also consider that variation among individuals will lead to a verity of solutions. These solutions may include the global optimal solution. Therefore, we have proposed a new FA that distinguishes the sex of fireflies [5]. In our proposed method, the movements of males and females are defined from their physical differences. Therefore, the movements of males and females are different. We investigate the features of our proposed methods using well-known test functions introduced at Congress on Evolutionary Computation(CEC) 2013. Numerical experiments indicate that our proposed methods are more efficient algorithms than the conventional FA.

This study is organized as follows: first, we explain the conventional Firefly Algorithm in Section II. Next, we describe in detail of our proposed methods in Section III. Followed by, we show numerical experiments. Finally, we conclude in this study and discuss future work.

II. The Conventional Firefly Algorithm (FA)

Firefly Algorithm (FA) has been developed by Yang, and it was based on the idealized behavior of the flashing characteristics of fireflies. The conventional FA is idealized these flashing characteristics as the following three rules

- all fireflies are unisex so that one firefly is attracted to other fireflies regardless of their sex;
- Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less brighter one will move towards the brighter one. The attractiveness is proportional to the brightness and they both decrease as their distance increases. If no one is brighter than a particular firefly, it moves randomly;
- The brightness or light intensity of a firefly is affected or determined by the landscape of the objective function to be optimized.

Attractiveness of firefly β is defined by

$$\beta = (\beta_0 - \beta_{min})e^{-\gamma r_{ij}^2} + \beta_{min}, \tag{1}$$

$$\gamma = \frac{1}{\sqrt{L}}, \tag{2}$$

$$L = \frac{|X_{max} - X_{min}|}{2}, \tag{3}$$

where γ is the light absorption coefficient, β_{min} is the minimum value of β, β_0 is the attractiveness at $r_{ij} = 0$, and r_{ij} is the distance between any two fireflies i and j at $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$. L means the average scale for the problem. The movement of the firefly i is attracted to another more attractive firefly j, and is determined by

$$\boldsymbol{x_i} = \boldsymbol{x_i} + \beta(\boldsymbol{x_j} - \boldsymbol{x_i}) + \alpha\boldsymbol{\epsilon_i}, \tag{4}$$

$$\boldsymbol{\epsilon_i} = (random - 0.5)L, \tag{5}$$

Algorithm 1 Firefly Algorithm

Objective function $f(x)$, $x = (x_1, ..., x_d)^T$
Initialize a population of fireflies $x_i (i = 1, 2, ..., n)$
Define light absorption coefficient γ
while $t < MaxGeneration$ **do**
  for $i = 1$ to n, all n fireflies **do**
    for $j = 1$ to n, all n fireflies **do**
      Light intensity I_i at x_i is determined by $f(x_i)$
      if $I_i > I_j$ **then**
        Move firefly i towards j in all d dimensions
      end if
      Attractiveness varies with distance r via $exp[-\gamma r]$
      Evaluate new solutions and update light intensity
    end for$j$
  end for$i$
  Rank the fireflies and find the current best
end while
Postprocess results and visualization

where $\boldsymbol{x_i}$ is the position vector of firefly i, $random$ is a uniform random number distributed in [0, 1] and $\alpha(t)$ is the randomization parameter. The parameter $\alpha(t)$ is defined by

$$\alpha(t) = \alpha(0) \left(\frac{10^{-4}}{0.9} \right)^{t/t_{max}}, \quad (6)$$

where t is the number of iteration.

Algorithm 1 shows pseudo code of the conventional FA for minimum optimization problems.

III. Our Proposed FAs

One of the rules of the conventional FA is all fireflies are unisex. However males and females exist in the real world. Therefore we distinguish sex of fireflies, that is, there are two swarms in our proposed method. The movement of female is modeled from the physical differences. In the real world, females are bigger than males. Thus in our proposed method females move slower than males.

In this study, we propose two patterns as female movement. The female movements $\boldsymbol{x}$ are determined with parameters V by

$$\boldsymbol{x_i} = \boldsymbol{x_i} + (\beta(\boldsymbol{x_j} - \boldsymbol{x_i}) + \alpha \boldsymbol{\epsilon_i})/V, \quad (7)$$

$$\boldsymbol{x_i} = \boldsymbol{x_i} + \beta(\boldsymbol{x_j} - \boldsymbol{x_i})/V + \alpha \boldsymbol{\epsilon_i}. \quad (8)$$

Pattern 1 is Eq. (7) and pattern 2 is Eq. (8). Males move the same as fireflies of the conventional FA.

IV. Numerical Experiments

We compare our two proposed methods to the conventional FA using the 28 CEC'13 Test Functions [6](see Table.I).

These functions are composed of 5 unimodal functions, 15 basic multimodal functions and 8 composition functions. The graphic form of the unimodal function is very simple because there are only one minimum. The graphic form of the basic multimodal function is more complex than the unimodal function because there are many local minima and local maxima. The composition functions are combined with some unimodal function and some multimodal function. The optimal solutions $\boldsymbol{x}^*$ of these benchmark functions is shifted from 0, and the global optima $f(\boldsymbol{x}^*)$ are not equal to 0. The search range of these functions is $[-100, 100]^D$, and the dimension N is 30. The total number of fireflies is also 30. We define $\alpha(0) = 0.5$, $\beta_0 = 1.0$, $\beta_{min} = 0.2$. Each parameter is decided by reference to [2]. Each numerical experiment is run 50 times. In each test functions, the maximum number of iterations t_{max} is 1500.

TABLE I
CEC'13 Test Functions

No.	Name	$f(\boldsymbol{x}^*)$
Unimodal Functions		
1	Sphere function	−1400
2	Rotated High Conditioned Elliptic Function	−1300
3	Rotated Bent Cigar Function	−1200
4	Rotated Discus Function	−1100
5	Different Powers Function	−1000
Basic Multimodal Functions		
6	Rotated Rosenbrock's Function	−900
7	Rotated Schaffers F7 Function	−800
8	Rotated Ackley's Function	−700
9	Rotated Weierstrass Function	−600
10	Rotated Griewank's Function	−500
11	Rastrigin's Function	−400
12	Rotated Rastrigin's Function	−300
13	Non-Continuous Rotated Rastrigin's Function	−200
14	Schwefel's Function	−100
15	Rotated Schwefel's Function	100
16	Rotated Katsuura Function	200
17	Lunacek Bi Rastrigin Function	300
18	Rotated Lunacek Bi Rastrigin Function	400
19	Expanded Griewank's plus Rosenbrock's Function	500
20	Rotated Expanded Scaffer's F6 Function	600
Composition Functions		
21	Composition Function 1 (n=5, Rotated)	700
22	Composition Function 2 (n=3, Unrotated)	800
23	Composition Function 3 (n=3, Rotated)	900
24	Composition Function 4 (n=3, Rotated)	1000
25	Composition Function 5 (n=3, Rotated)	1100
26	Composition Function 6 (n=5, Rotated)	1200
27	Composition Function 7 (n=5, Rotated)	1300
28	Composition Function 8 (n=5, Rotated)	1400

We compare the number that our proposed FA wins the conventional FA in the comparison of average error value, when the parameter $V = 2$ and female percentage is 50%(see Table II).

Table II shows our proposed methods are superior to the conventional FA. When three algorithms are compared, pattern 1 of our proposed methods is the most efficient algorithm. However when we compare two our proposed methods to the conventional FA, the number that pattern 2 of our proposed methods obtain is larger and the sum ranks that pattern 2 of our proposed methods obtain is fewer.

In the case of comparing the numbers each algorithm obtains the best result for the unimodal functions, pattern 1 of our proposed methods is the most efficient algorithm of three algorithms, the conventional FA is the second efficient algorithm. We focus on the minimum value then pattern 1

TABLE II
NUMERICAL EXPERIMENTS

f		FA	our proposed method	
			pattern 1	pattern 2
f_1	avg	1.47×10^{-4}	$\mathbf{2.73 \times 10^{-5}}$	1.05×10^{-4}
	min	1.06×10^{-4}	1.62×10^{-5}	6.20×10^{-5}
	max	2.14×10^{-4}	3.90×10^{-5}	1.55×10^{-4}
	std	2.60×10^{-5}	4.89×10^{-6}	1.61×10^{-5}
f_2	avg	2.82×10^{6}	$\mathbf{2.37 \times 10^{6}}$	3.93×10^{6}
	min	8.44×10^{5}	4.84×10^{5}	5.82×10^{5}
	max	6.54×10^{6}	7.08×10^{6}	1.07×10^{7}
	std	1.45×10^{6}	1.14×10^{6}	2.41×10^{6}
f_3	avg	$\mathbf{7.08 \times 10^{4}}$	1.21×10^{6}	6.59×10^{5}
	min	6.67×10^{2}	3.68×10^{1}	1.41×10^{2}
	max	1.64×10^{6}	1.78×10^{7}	1.88×10^{7}
	std	2.61×10^{5}	3.68×10^{6}	3.13×10^{6}
f_4	avg	1.93×10^{5}	$\mathbf{1.30 \times 10^{5}}$	2.16×10^{5}
	min	9.46×10^{4}	7.27×10^{4}	1.19×10^{5}
	max	3.60×10^{5}	1.90×10^{5}	3.38×10^{5}
	std	5.32×10^{4}	2.82×10^{4}	5.62×10^{4}
f_5	avg	6.05×10^{-3}	$\mathbf{4.03 \times 10^{-3}}$	5.34×10^{-3}
	min	4.65×10^{-3}	2.79×10^{-3}	3.36×10^{-3}
	max	7.74×10^{-3}	5.24×10^{-3}	7.15×10^{-3}
	std	8.06×10^{-4}	5.14×10^{-4}	6.74×10^{-4}
f_6	avg	$\mathbf{2.30 \times 10^{1}}$	2.49×10^{1}	2.33×10^{1}
	min	2.11×10^{1}	2.23×10^{1}	2.14×10^{1}
	max	2.83×10^{1}	2.66×10^{1}	2.57×10^{1}
	std	1.11×10^{0}	9.13×10^{-1}	8.41×10^{-1}
f_7	avg	6.24×10^{-1}	7.81×10^{-1}	$\mathbf{6.00 \times 10^{-1}}$
	min	1.47×10^{-1}	3.10×10^{-2}	4.64×10^{-2}
	max	6.77×10^{0}	3.06×10^{0}	3.21×10^{0}
	std	9.64×10^{-1}	7.11×10^{-1}	7.53×10^{-1}
f_8	avg	2.14×10^{1}	$\mathbf{2.14 \times 10^{1}}$	2.14×10^{1}
	min	2.12×10^{1}	2.10×10^{1}	2.11×10^{1}
	max	2.16×10^{1}	2.15×10^{1}	2.16×10^{1}
	std	8.47×10^{-2}	9.36×10^{-2}	9.22×10^{-2}
f_9	avg	6.07×10^{0}	7.69×10^{0}	$\mathbf{5.53 \times 10^{0}}$
	min	1.56×10^{0}	3.40×10^{0}	2.01×10^{0}
	max	8.80×10^{0}	1.25×10^{1}	9.76×10^{0}
	std	1.67×10^{0}	1.65×10^{0}	1.63×10^{0}
f_{10}	avg	1.07×10^{-1}	7.37×10^{-3}	$\mathbf{6.73 \times 10^{-3}}$
	min	1.45×10^{-3}	4.09×10^{-4}	1.40×10^{-3}
	max	3.27×10^{-2}	3.00×10^{-2}	2.15×10^{-2}
	std	9.32×10^{-3}	6.81×10^{-3}	6.14×10^{-3}
f_{11}	avg	2.89×10^{1}	$\mathbf{2.44 \times 10^{1}}$	2.48×10^{1}
	min	1.49×10^{1}	1.29×10^{1}	1.39×10^{1}
	max	4.58×10^{1}	4.08×10^{1}	4.18×10^{1}
	std	7.71×10^{0}	6.29×10^{0}	6.10×10^{0}
f_{12}	avg	2.86×10^{1}	$\mathbf{2.17 \times 10^{1}}$	2.56×10^{1}
	min	1.39×10^{1}	9.95×10^{0}	9.95×10^{0}
	max	4.68×10^{1}	3.68×10^{1}	5.17×10^{1}
	std	8.02×10^{0}	5.71×10^{0}	8.70×10^{0}
f_{13}	avg	6.14×10^{1}	$\mathbf{5.24 \times 10^{1}}$	5.26×10^{1}
	min	2.39×10^{1}	1.92×10^{1}	1.62×10^{1}
	max	1.14×10^{2}	9.53×10^{1}	1.13×10^{2}
	std	2.16×10^{1}	1.89×10^{1}	2.13×10^{1}
f_{14}	avg	1.49×10^{3}	1.54×10^{3}	$\mathbf{1.38 \times 10^{3}}$
	min	5.91×10^{2}	7.85×10^{2}	4.06×10^{2}
	max	2.48×10^{3}	2.69×10^{3}	2.74×10^{3}
	std	4.24×10^{2}	4.50×10^{2}	4.91×10^{2}
f_{15}	avg	1.51×10^{3}	1.52×10^{3}	$\mathbf{1.43 \times 10^{3}}$
	min	7.22×10^{2}	7.65×10^{2}	4.71×10^{2}
	max	2.86×10^{3}	2.35×10^{3}	2.40×10^{3}
	std	4.17×10^{2}	4.02×10^{2}	4.17×10^{2}
f_{16}	avg	9.57×10^{-2}	$\mathbf{3.17 \times 10^{-2}}$	4.36×10^{-2}
	min	1.69×10^{-2}	6.88×10^{-3}	1.25×10^{-2}
	max	2.58×10^{-1}	8.80×10^{-2}	1.01×10^{-1}
	std	4.61×10^{-2}	1.90×10^{-2}	2.17×10^{-2}
f_{17}	avg	6.13×10^{1}	$\mathbf{5.39 \times 10^{1}}$	5.63×10^{1}
	min	4.49×10^{1}	4.46×10^{1}	4.58×10^{1}
	max	7.86×10^{1}	6.30×10^{1}	9.27×10^{1}
	std	8.65×10^{0}	4.72×10^{0}	8.14×10^{0}
f_{18}	avg	6.07×10^{1}	5.47×10^{1}	$\mathbf{5.52 \times 10^{1}}$
	min	4.68×10^{1}	4.40×10^{1}	4.28×10^{1}
	max	8.16×10^{1}	7.77×10^{1}	6.74×10^{1}
	std	7.67×10^{0}	7.15×10^{0}	5.80×10^{0}
f_{19}	avg	3.35×10^{0}	$\mathbf{3.19 \times 10^{0}}$	3.30×10^{0}
	min	2.23×10^{0}	2.28×10^{0}	2.24×10^{0}
	max	5.10×10^{0}	4.73×10^{0}	5.17×10^{0}
	std	6.60×10^{-1}	6.52×10^{-1}	5.77×10^{-1}
f_{20}	avg	$\mathbf{1.46 \times 10^{1}}$	1.48×10^{1}	1.49×10^{1}
	min	1.16×10^{1}	1.17×10^{1}	1.23×10^{1}
	max	1.50×10^{1}	1.50×10^{1}	1.50×10^{1}
	std	9.88×10^{-1}	7.62×10^{-1}	5.45×10^{-1}
f_{21}	avg	3.19×10^{2}	3.29×10^{2}	$\mathbf{3.17 \times 10^{2}}$
	min	2.00×10^{2}	1.00×10^{2}	2.00×10^{2}
	max	4.44×10^{2}	4.44×10^{2}	4.44×10^{2}
	std	7.87×10^{1}	7.54×10^{1}	8.39×10^{1}
f_{22}	avg	1.54×10^{3}	1.62×10^{3}	$\mathbf{1.41 \times 10^{3}}$
	min	6.91×10^{2}	4.87×10^{2}	6.42×10^{2}
	max	2.72×10^{3}	2.95×10^{3}	3.86×10^{3}
	std	4.71×10^{2}	4.90×10^{2}	5.48×10^{2}
f_{23}	avg	1.70×10^{3}	1.85×10^{3}	$\mathbf{1.61 \times 10^{3}}$
	min	6.12×10^{2}	4.17×10^{2}	7.73×10^{2}
	max	2.65×10^{3}	3.96×10^{3}	2.60×10^{3}
	std	4.71×10^{2}	6.14×10^{2}	4.63×10^{2}
f_{24}	avg	2.13×10^{2}	$\mathbf{2.10 \times 10^{2}}$	2.16×10^{2}
	min	2.01×10^{2}	2.00×10^{2}	2.00×10^{2}
	max	2.28×10^{2}	2.33×10^{2}	2.31×10^{2}
	std	8.16×10^{0}	1.07×10^{1}	7.05×10^{0}
f_{25}	avg	2.19×10^{2}	2.27×10^{2}	$\mathbf{2.18 \times 10^{2}}$
	min	2.10×10^{2}	2.00×10^{2}	2.10×10^{2}
	max	2.40×10^{2}	2.49×10^{2}	2.28×10^{2}
	std	5.51×10^{0}	1.04×10^{1}	4.24×10^{0}
f_{26}	avg	2.95×10^{2}	$\mathbf{2.93 \times 10^{2}}$	2.97×10^{2}
	min	2.00×10^{2}	2.00×10^{2}	2.00×10^{2}
	max	3.26×10^{2}	3.34×10^{2}	3.21×10^{2}
	std	3.88×10^{1}	4.43×10^{1}	3.61×10^{1}
f_{27}	avg	$\mathbf{3.90 \times 10^{2}}$	4.34×10^{2}	4.12×10^{2}
	min	3.13×10^{2}	3.05×10^{2}	3.07×10^{2}
	max	5.33×10^{2}	6.27×10^{2}	5.59×10^{2}
	std	7.38×10^{1}	1.07×10^{2}	8.01×10^{1}
f_{28}	avg	3.88×10^{2}	3.36×10^{2}	$\mathbf{3.28 \times 10^{2}}$
	min	1.00×10^{2}	1.00×10^{2}	1.00×10^{2}
	max	1.36×10^{3}	1.31×10^{3}	1.31×10^{3}
	std	3.21×10^{2}	1.97×10^{2}	2.05×10^{2}
f		FA	pattern 1	pattern 2
			our proposed method	

		pattern 1	pattern 2
best solution	4	13	11
sum ranks	54	54	50
more than the conventional FA		16	20
compare our proposed mothods		13	16

of our proposed methods obtains the smallest value in all unimodal functions. Therefore pattern 1 of our proposed methods is the earliest convergence algorithm of three algorithms.

In the case of comparing the numbers each algorithm obtains the best result for the basic multimodal functions, pattern 1 of our proposed methods is the most efficient algorithm of three algorithms, pattern 2 of our proposed methods is the second efficient algorithm. Sum rank of pattern 1 of our proposed methods for basic multimodal functions is 28. Sum rank of pattern 2 of our proposed methods for basic multimodal functions is 25. Therefore we cannot say that pattern 1 of our proposed methods easy to escape local optima.

In the case of comparing the numbers each algorithm obtains the best result for the composition functions, pattern 2 of our proposed methods is the most efficient algorithm of three algorithms, pattern 1 of our proposed methods is the second efficient algorithm. In addition, pattern 2 of our proposed methods the smallest sum rank for basic multimodal functions and composition functions. Therefore pattern 2 of our proposed methods is the most efficient algorithm of three algorithms.

V. Conclusion

In this study, we have proposed two new FAs existing fireflies flying slow. We applied these two FAs to CEC'13 test functions. Numerical experiments indicate that our proposed methods are more efficient algorithms than the conventional FA. Pattern 1 of our proposed methods is the earliest convergence algorithm of three algorithms. Pattern 2 of our proposed methods is the most efficient algorithm of three algorithms.

In the future work, we try to improve these two FAs, compare other improved algorithms and apply actual optimization problems.

Acknowledgment

This work was partly supported by JSPS Grant-in-Aid for Challenging Exploratory Research 26540127.

References

[1] J. Kennedy, R. Eberhart, *"Particle Swarm Optimization"*, Proceedings of the IEEE international conference on neural networks, pp. 19421948, 1995.
[2] Yang, X.S., *"Nature-Inspired Metaheuristic Algorithms Second Edition"*, Luniver Press, 2010.
[3] Yang, X.S., *"Firefly Algorithms for multimodal Optimization"*, Int. J. Bio-Inspired Computation, Vol. 2, No. 2, pp.78-84, 2010.
[4] H. Matsushita, *"Firefly Algorithm with Dynamically Changing Connections"*, Proceedings of International Symposium on Nonlinear Theory and its Application, pp.906-909, 2014.
[5] S. Lukasik, S. Zak, *"Firefly Algorithm for Continuous Constrained Optimization Tasks"*, Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems, Vol. 5796 of the series Lecture Notes in Computer Science, pp.97-106, 2009.
[6] J.J. Liang, B.Y. Qu, P.N. Suganthan, and A.G. Hernandez-Diaz, *"Problem Definitions and Evaluation Criteria for the CEC* 2013 *Special Session on Real-Parameter Optimization"*, Technical Report 201212, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, January 2013.