

An Adaptive Fruit Fly Optimization Algorithm Based on Velocity Variable

Mindi Lu¹, Yongquan Zhou², Qifang Luo² and Kang Huang¹

¹*School of Computer and Electronic Information, Guangxi University, Nanning, 530004, China*

²*College of Information Science and Engineering, Guangxi University for Nationalities, Nanning, 530004, China*
E-mail:yongquanzhou@126.com

Abstract

In view of the problems of easily relapsing into local extremum and low convergence accuracy of fruit fly optimization algorithm (FOA), this paper proposes a adaptive fruit fly optimization algorithm based on velocity variable (VFOA). The idea of this algorithm is based on the flight characteristics of fruit fly, using particle swarm optimization (PSO) concept of particle velocity, based on fruit fly optimization algorithm, improved the convergence speed of fruit fly optimization algorithm by adding the particle velocity variable parameter. Finally, simulation comparison experiment tests are conducted on 13 benchmark functions, test results show that adaptive fruit fly optimization algorithm based on velocity variable VFOA compared to swarm intelligence algorithms of FOA, PSO, CS, and so on, the convergence speed and accuracy are improved obviously.

Keywords: *Fruit fly optimization algorithm, adaptive, velocity, particle swarm optimization*

1. Introduction

Swarm intelligence optimization algorithm is a new optimization method. In recent years, draw more and more people's attention, is becoming a hot research field and direction of artificial intelligence, computer science and technology and so on. Mechanism of swarm intelligence optimization algorithm is derived from it simulates all kinds of animal's group behavior, using information exchange and cooperation among individuals in groups to achieve the optimization objective. Compared with other types of optimization method, it is simple and high efficiency. Such as Genetic Algorithm(GA) [1], Particle Swarm Optimization(PSO) [2], Firefly Algorithm(FA) [3], Group Search Optimizer(GSO) [4], Cuckoo Search(CS) [5], Harmony Search(HS) [6], Bat-Inspired Algorithm(BA) [7], Animal Migration Optimization(AMO) [8], artificial bee colony(ABC) [9] and so on. The swarm intelligence optimization algorithm as a new heuristic algorithm based on swarm intelligence, has been widely applied in many fields of engineering technology, military technology, network communication, finance, automatic control, resource allocation, economic management.

Fruit Fly Optimization Algorithm (FOA) [10] is proposed by Taiwan scholar Wen-Tsao Pan in 2011, it is a kind of new evolutionary algorithm for global optimization based on feeding behavior of fruit fly. Compared to other intelligent optimization algorithms, Fruit Fly Optimization Algorithm is simple to understand, its adjustment parameters are less, its convergence speed is fast, and it's easy to implement. This algorithm gradually arouses scholar's close attentions at domestic and foreign. Fruit Fly Optimization Algorithm has been successfully applied to solve the function optimization, generalized regression neural

network parameter optimization, grey neural network parameter optimization and so on. But the algorithm still has some shortcomings, such as easily trapped into local optimal value and low accuracy.

In view of the problems of easily relapsing into local extremum and low convergence accuracy of fruit fly optimization algorithm (FOA), this paper proposes an adaptive fruit fly optimization algorithm based on velocity variable (VFOA). The idea of this algorithm is based on the flight characteristics of fruit fly, using particle swarm optimization (PSO) concept of particle velocity, based on fruit fly optimization algorithm, improved the convergence speed of fruit fly optimization algorithm by adding the particle velocity variable parameter. Finally, simulation comparison experiment tests are conducted on 13 benchmark functions, test results show that adaptive fruit fly optimization algorithm based on velocity variable VFOA compared to swarm intelligence algorithms of FOA, PSO, CS, and so on, the convergence speed and accuracy are improved obviously.

2. Fruit Fly Optimization Algorithm

The Fruit Fly Optimization Algorithm (FOA) is a new method for finding global optimization based on the food finding behavior of the fruit fly. The fruit fly itself is superior to other species in sensing and perception, especially in smell and vision. The smell organs of fruit flies can find all kinds of scents floating in the air, it can even smell food source from 40 km away. Then, after it gets close to the food location, it can also use its sensitive vision to find food and the company's flocking location, and fly towards that direction too. The mathematical model for the description of the Fruit fly optimization algorithm can be summarized as follows:

Step 1. Random initial fruit fly swarm location:

$$Init X_axis$$

$$Init Y_axis$$

Step 2. Give the random direction and distance for the search of food using smell by an individual fruit fly:

$$X_i = X_axis + Random\ Valve$$

$$Y_i = Y_axis + Random\ Valve$$

Step 3. Since the food location cannot be known, the distance to the origin is thus estimated first ($Dist$), then the smell concentration judgment value (S) is calculated, and this value is the reciprocal of distance:

$$Dist_i = \sqrt{X_i^2 + Y_i^2}$$

$$S_i = 1 / Dist_i$$

Step 4. Substitutes smell concentration judgment value (S) into smell concentration judgments function (or called *Fitness function*) so as to find the smell concentration ($Smelli$) of the individual location of the fruit fly:

$$Smelli = Function(S_i)$$

Step 5. Find out the fruit fly with maximal smell concentration (finding the maximal value) among the fruit fly swarm:

$$[bestSmell\ bestIndex] = \max(Smell)$$

Step 6. Keep the best smell concentration value and x , y coordinate, and at this moment, the fruit fly swarm will use vision to fly towards that location:

$$Smellbest = bestSmell$$

$$X_axis = X(bestIndex)$$

$$Y_axis = Y(bestIndex)$$

Step 7. Enter iterative optimization to repeat the implementation of step 2 to step 5, then judge if the smell concentration is superior to the previous iterative smell concentration, if so, implement step 6.

3. Adaptive Fruit Fly Optimization Algorithm Based on Velocity Variable

The FOA is easily trapped into local optimal value, it reduce the convergence speed and accuracy, and appear premature convergence. According to the characteristics and behavior characteristics of fruit fly, which not only has a keen sense of smell and the developed vision, also has a very good ability to fly, and fly with a certain speed, according to the characteristics of fruit fly, adding velocity variable of PSO, based on fruit fly optimization algorithm.

The fruit fly optimization algorithm is based on two-dimensional space design, The change of velocity and position is according to the following equation change:

$$V_{ix} = w * V_{ix} + c_1 rand() (P_{ix} - X_i) + c_2 rand() (G_x - X_i) \quad (1)$$

$$V_{iy} = w * V_{iy} + c_1 rand() (P_{iy} - Y_i) + c_2 rand() (G_y - Y_i) \quad (2)$$

$$X_i = X_i + V_{ix} \quad (3)$$

$$Y_i = Y_i + V_{iy} \quad (4)$$

Where w is inertia weight, c_1 and c_2 are Acceleration constant, $rand()$ is random function.

P_{ix} is i -th fruit fly experienced the best position on the x direction, P_{iy} is i -th fruit fly experienced the best position on the y direction. G_x is all fruit fly experienced the best position on the x direction, G_y is all fruit fly experienced the best position on the y direction.

The steps of the proposed VFOA can be depicted as follow:

Step 1. Random initial fruit fly swarm location:

Init X _ axis

Init Y _ axis

Step 2. Give the random direction and distance for the search of food using smell by an individual fruit fly:

$$X_i = X_axis + Random\ Valve$$

$$Y_i = Y_axis + Random\ Valve$$

Step 3. Implement FOA step 3 to step 5.

Step 4. Keep the best smell concentration value and x , y coordinate, and at this moment, the fruit fly swarm will use vision to fly towards that location:

$$Smellbest = bestSmell$$

$$X_axis = X (bestIndex)$$

$$Y_axis = Y (bestIndex)$$

Step 5. Using the formula (1) to formula (4) update velocities and locations.

Step 6. Implement FOA step 3 to step 5, then judge if the smell concentration is superior to the previous iterative smell concentration, if so, implement step 4.

Step 7. If the iterations attain to the maximum number, the search is stopped, If not jump to step 5 to continue the search.

4. Experimental Results and Analysis

In order to verify the performance of VFOA, this paper selects 13 standards benchmark functions for performance testing, as shown in Table 1. 13 functions are 30 dimensional functions, Among these functions, functions $f_1 \sim f_9$ are unimodal high-dimensional functions. Functions $f_{10} \sim f_{13}$ are Multimodal high-dimensional functions.

Table 1. Benchmark Functions Based in our Experimental Study

Test functions	D	Range	Optimum
$f_1 = \sum_{i=1}^n x_i^2$	30	[-100, 100]	0
$f_2 = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10,10]	0
$f_3 = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	30	[-100, 100]	0
$f_4 = \max \{ x_i , 1 \leq i \leq n \}$	30	[-100, 100]	0
$f_5 = \sum_{i=1}^n (\lfloor x_i + 0.5 \rfloor)^2$	30	[-100, 100]	0
$f_6 = \sum_{i=1}^n ix_i^4 + random[0,1)$	30	[-1.28,1.28]	0
$f_7(x) = -(\prod_{i=1}^n \cos(x_i))e^{-(\sum_{i=1}^n (x_i - \pi)^2)}$	30	[-100, 100]	-1
$f_8(x) = \sum_{i=1}^n x_i ^{i+1}$	30	[-1,1]	0
$f_9(x) = -\sum_{i=1}^{n-1} (\exp(-(x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1})) / 8) \cos(4\sqrt{x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}})$	30	[-5,5]	-n+1
$f_{10}(x) = 1 + \sum_{i=1}^n (\sin(x_i))^2 - 0.1 \prod_{i=1}^n (\exp(-x_i^2))$	30	[-10,10]	0.9
$f_{11} = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5.12,5.12]	0
$f_{12}(x) = 20 + e - 20e^{-\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}} - e^{-\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)}$	30	[-32,32]	0
$f_{13}(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos(x_i / \sqrt{i}) + 1$	30	[-600,600]	0

4.1. Experimental Set Up

All algorithms are coded in MATLAB R2012b, and experiments are performed on an Intel core 2 Duo processor T6600 with 2.0 GB of memory.

4.2. Experiment Parameters Setting

The values of c_1 , c_2 , w have a certain influence on VFOA, After many experiments, This paper recommends c_1 and c_2 take the same value between the 2.6~3.0, The value of w is 0.76. Specific parameters of each function as shown in table 2. The population is 30, the iteration number is 600.

Table 2. Parameters Setting

Functions	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}
$C_1 = C_2$	3	3	3	3	3	2.7	2.6	3	2.6	2.6	2.7	2.7	2.7
w	0.76	0.76	0.76	0.76	0.76	0.76	0.76	0.76	0.76	0.76	0.76	0.76	0.76

4.3. Algorithm Comparison

In order to test the performance of the VFOA, use simulation experiment comparison respectively with FOA, PSO, CS, GSO, GSA, ABC and AMO algorithm. Each algorithm independently operates 20 times to $f_1 \sim f_{13}$ functions, the mean and variance of test results as shown in Table 3.

The setting values of algorithm control parameters of the mentioned algorithms are given below:

PSO: According to the given reference value from literature [2], $w=0.6$, $c_1 = c_2 = 2$;

CS: According to the given reference value from literature [5], $\beta = 1.5$, $\rho_0 = 1.5$, the population size is 50 because of this algorithm has two phases;

GSO: According to the given reference value from literature [4];

GSA: According to the given reference value from literature [12], $G_0 = 100$, $\alpha = 20$, κ_0 is set to NP and is decreased linearly to 1. the population size is 100;

ABC: According to the given reference value from literature [9], limit=50D, the population size is 50 because of this algorithm has two phases;

AMO: According to the given reference value from literature [8], the population size is 50 because of this algorithm has two phases;

FOA: The population is 30.

Table 3. Test Result of each Algorithm for each Function

Functions Algorithm		PSO	CS	GSO	GSA	ABC	AMO	FOA	VFOA
f_1	Mean	3.33E-10	5.66E-06	1.95E-08	3.37E-18	7.61E-16	8.65E-40	0.0993	0
	StdDev	7.04E-10	2.86E-06	1.16E-08	8.09E-19	1.73E-16	1.04E-39	0.2392	0
f_2	Mean	6.66E-11	0.002	3.70E-05	8.92E-09	1.42E-15	8.23E-32	0.5047	0
	StdDev	9.26E-11	8.10E-04	8.62E-05	1.33E-09	5.53E-16	3.41E-32	1.272	0
f_3	Mean	2.9847	0.0014	5.7829	0.1126	2.40E+03	8.89E-04	4.37E+03	0
	StdDev	2.2778	6.10E-04	3.6813	0.1266	656.96	8.73E-04	1.05E+04	0
f_4	Mean	7.9997	3.2388	0.1078	9.93E-10	18.5227	2.86E-05	0.0015	0
	StdDev	2.5351	0.6644	3.99E-02	1.19E-10	4.2477	2.35E-05	1.06E-05	0
f_5	Mean	3.69E-10	5.43E-06	1.60E-02	3.34E-18	3.09E-20	0	0	0
	StdDev	6.37E-10	2.24E-06	0.1333	5.68E-19	4.01E-20	0	0	0
f_6	Mean	0.0135	0.0096	7.37E-02	0.0039	0.0324	0.0017	0.0664	6.49E-05
	StdDev	0.0041	0.0028	9.25E-02	0.0021	0.0059	4.71E-04	0.2439	7.85E-05
f_7	Mean	0	-1.27E-58	0	0	0	0	-0.4408	-1
	StdDev	0	1.64E-58	0	0	0	0	0.5066	0
f_8	Mean	5.79E-10	1.28E-06	1.85E-23	9.42E-18	1.01E-13	1.68E-175	2.27E-06	0
	StdDev	3.89E-09	7.55E-07	4.13E-23	1.40E-17	1.57E-13	2.56E-175	3.44E-08	0
f_9	Mean	-1.1894	-21.44106	-28.822	-20.4055	-24.9727	-13.4105	-28.9383	-29
	StdDev	1.736	1.4247	0.3981	1.2491	0.9152	1.0715	0.2143	0
f_{10}	Mean	1.002	0.9833	1.001	1	1	1.219	0.9922	0.9
	StdDev	0.1351	0.0445	2.00E-04	5.93E-17	1.87E-06	0.0515	0.2285	3.40E-16
f_{11}	Mean	18.2675	51.2202	1.0179	7.2831	1.64E-07	0	21.4936	0
	StdDev	4.7965	8.1069	0.9509	1.8991	3.61E-07	0	45.0126	0
f_{12}	Mean	3.87E-06	2.375	2.65E-05	1.47E-09	1.19E-09	4.44E-15	0.2521	8.88E-16
	StdDev	2.86E-06	1.1238	3.08E-05	1.44E-10	5.01E-10	0	0.5765	0
f_{13}	Mean	0.0168	4.49E-05	3.07E-02	0	6.92E-13	0	0.0065	0
	StdDev	0.0205	8.96E-05	3.08E-02	0	1.05E-12	0	0.0181	0

The results can be seen from Table 2, VFOA obtained solution is better than FOA and other algorithms to unimodal functions $f_1 \sim f_9$, VFOA can find the optimal solution from all

function except function f_6 , and the solution from function f_6 is better than other algorithms. To the Multimodal functions $f_{10} \sim f_{13}$, VFOA can find the optimal solution except function f_{12} , and the solution from function f_{12} is better than other algorithms. The variance of VFOA reached 0 except function f_6 and f_{10} , It is very strong stability, and FOA often falls into local optimal solution, it can be seen from here, VFOA improves the deficiency.

Figure 1~Figure 5 are fitness function curves for f_1 , f_4 , f_8 , f_{11} , f_{12} (note: in order to display and observe the fitness function curves, this paper takes the logarithm base 10). To the unimodal functions f_1 , f_4 , f_8 , accuracy in VFOA is significantly higher than FOA and other algorithms, reaches theoretically optimal value, and convergence speed in VFOA is very fast. To the multimodal function f_{11} , VFOA and AMO are better than other algorithms, reach theoretically optimal value, but convergence speed in VFOA is obviously faster than in AMO algorithm. To the multimodal function f_{12} , accuracy in VFOA is also significantly higher than other algorithms, and convergence speed is very fast.

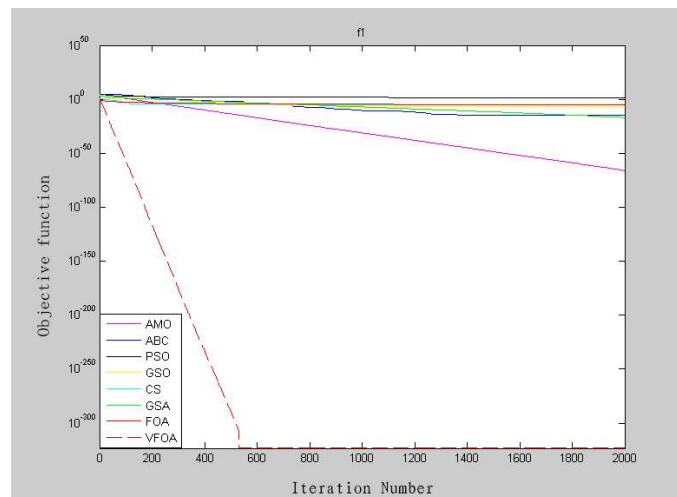


Figure 1. Fitness Function Curves for the Function f_1

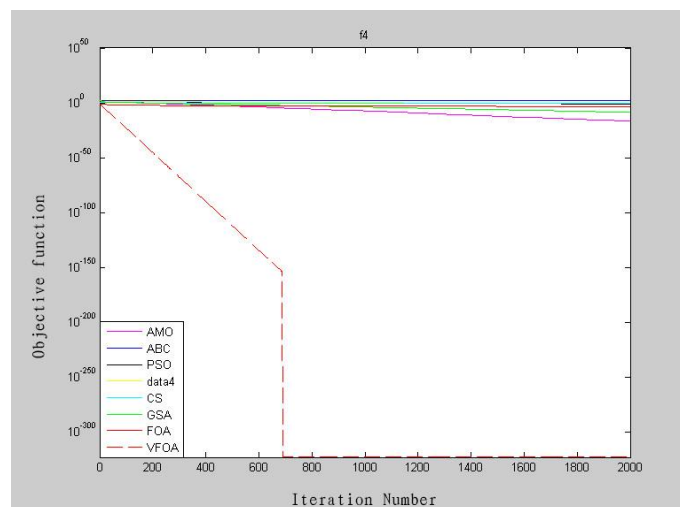


Figure 2. Fitness Function Curves for the Function f_4

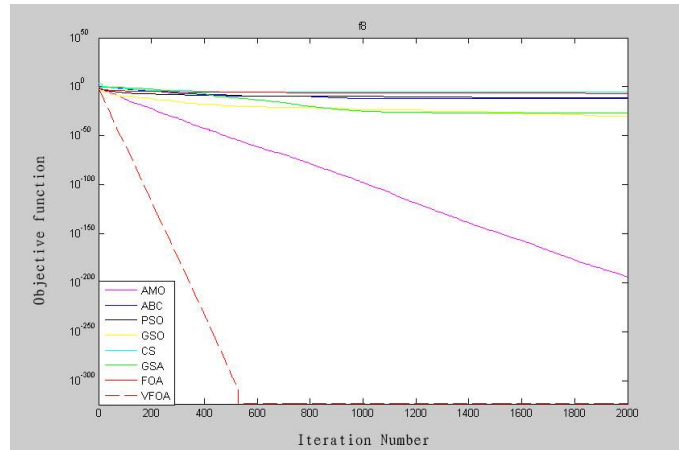


Figure 3. Fitness Function Curves for the Function f_8

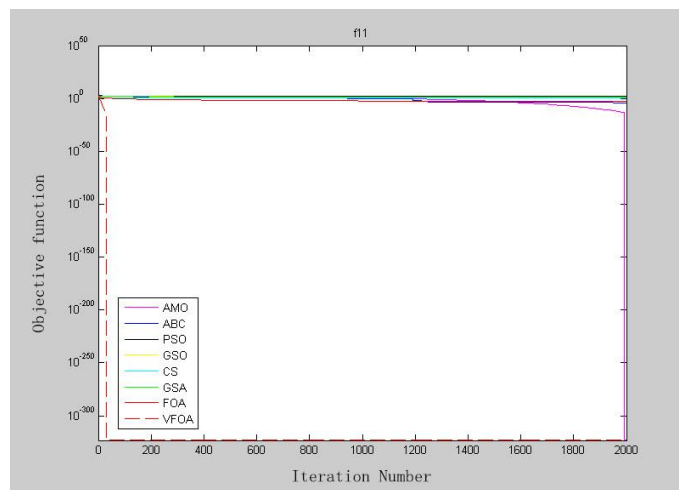


Figure 4. Fitness Function Curves for the Function f_{11}

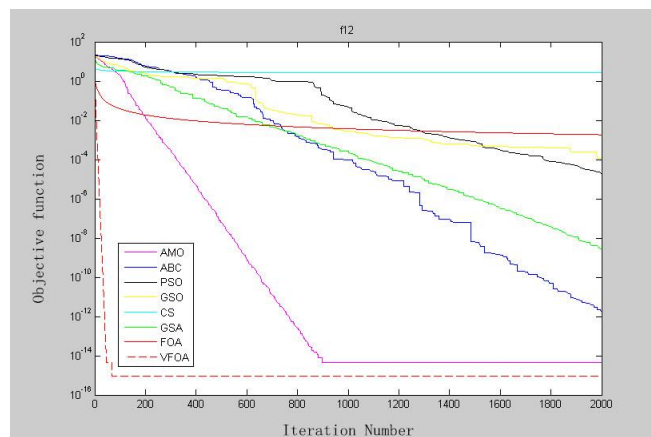


Figure 5. Fitness Function Curves for the Function f_{12}

4.4. The Higher Dimensional Function Test

Compared to the unimodal functions, multimodal functions are more difficult to find the optimal value, so we use multimodal functions $f_{10} \sim f_{13}$ to test VFOA, Test dimension respectively is 1000 D, 10000 D and 100000 D. Comparative object are

which can reach the optimal value or whose best value closes to the best value of VFOA in each function. So VFOA compared to FOA in function f_{10} , compared to AMO in function f_{11} and f_{12} , compared to AMO and GSA in function f_{13} . if algorithm get poor result in the 1000 D, we will not continue to higher dimension. Test results show in Table 4 which give best value and run time of each algorithm.

Table 4. Test Results

Functions	D	VFOA	Time(s)	AMO	Time(s)	GSA	Time(s)	FOA	Time(s)
f_{10}	1000	0.9	6.53598	—	—	—	—	0.9025	5.21885
	10000	0.9	114.398	—	—	—	—	0.9248	85.1154
	100000	0.9	1009.62	—	—	—	—	2.3e+03	1802.08
f_{11}	1000	0	4.79682	2.2e-08	1063.31	—	—	—	—
	10000	0	101.672	—	—	—	—	—	—
	100000	0	1009.48	—	—	—	—	—	—
f_{12}	1000	8.9e-16	5.60644	3.8593	1096.48	—	—	—	—
	10000	8.9e-16	118.681	—	—	—	—	—	—
	100000	8.9e-16	996.297	—	—	—	—	—	—
f_{13}	1000	0	6.08831	126.231	1073.73	4.3e+03	385.180	—	—
	10000	0	115.283	—	—	—	—	—	—
	100000	0	1219.81	—	—	—	—	—	—

It can be seen from Table 4, to the multimodal functions, test result of VFOA not only has high convergence accuracy, but also run fast in the 1000 D, 10000 D, 100000 D. To the function f_{10} , The FOA's running time is slightly shorter than the VFOA, but the convergence accuracy is obviously not as good as VFOA. To the functions $f_{11} \sim f_{13}$, The convergence accuracy and running time in VFOA are obviously better than in other algorithm.

5. Conclusions

Aiming at the phenomenon of easily relapsing into local extremum and low convergence accuracy of fruit fly optimization algorithm, this paper proposes an adaptive fruit fly optimization algorithm based on velocity variable. 13 benchmark functions' contrast test results show that VFOA not only improved deficiency of easily falling into local minimum of FOA, has better global search ability and local optimization ability, greatly improves the convergence accuracy and convergence speed of FOA, and can find the theoretically optimal value in up to 100000 D, greatly improves the performance of FOA.

Acknowledgments

This work is supported by National Science Foundation of China under Grants No.61165015, 61463007. Key Project of Guangxi Science Foundation under Grant No. 2012GXNSFDA053028, Key Project of Guangxi High School Science Foundation under Grant No. 20121ZD008, and the Foundation of Guangxi University for Nationalities research project under Grant No. 2012MDZD037.

References

- [1] J. H. Holland, "Adaptation in Natural and Artificial Systems", Cambridge, USA: MIT Press, (1992).
- [2] J. Kennedy and R. Eberhart, "Particle swarm optimization. Proc Int Conf Neural Netw., vol. 4, (1995), pp. 1942–1948.
- [3] Y. Xinshe, "Multi-objective Firefly Algorithm for Continuous Optimization", Engineering with Computers, vol. 29, no. 2, (2013), pp. 175-184.
- [4] S. He, Q. H. Wu and J. R. Saunders, "Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior", IEEE Trans. Evol. Computation, vol. 13, no. 5, (2009) October, pp. 973-990.
- [5] X. S. Yang and S. Deb, "Cuckoo search via Levy flights, in: world congress on nature & biologically inspired computing (NaBIC2009)", IEEE Publication, USA, (2009), pp. 210–214.
- [6] Z. W. Geem, J. H. Kim and G. V. Loganathan, "A New Heuristic Optimization Algorithm: Harmony Search", Simulation, vol. 76, no. 2, (2001), pp. 60-68.
- [7] X .S. Yang, "A nature-Inspired metaheuristic bat-inspired algorithms", studies in computational intelligence, vol. 284, (2010), pp. 65-74.
- [8] X. Li, J. Zhang and M. Yin, "Animal Migration Optimization: an optimization algorithm inspired by animal migration behavior", Neural Computing and Applications, (2013).
- [9] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm", J Global Optim., vol. 39, no. 3, (2007), pp. 459–471.
- [10] W.-T. Pan, "A new fruit fly optimization algorithm: Taking the financial distress model as an example", Knowledge-Based Systems, vol. 26, (2012), pp. 69-74.
- [11] W.-T. Pan, "fruit fly optimization algorithm", Taipei: Tsang Hai Book Publishing Co., (2011), pp. 10-12.
- [12] R. Esmat, N.-P. Hossein and S. Saryazdi, "GSA: A Gravitational Search Algorithm", Inf Sci., vol. 179, (2009), pp. 2232–2248.

Authors



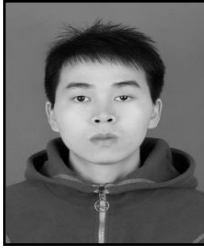
Mindi Lu, M.S., received his B.S degree from College of computer Science and Technology, Guangxi University for Nationalities, Nanning, China, in 2012. He is currently research interest is in computation intelligence, swarm intelligence algorithm.



Yongquan Zhou, Ph.D & Prof. He received the MS degree in computer science from Lanzhou University, Lanzhou, China, in 1993 and the Ph.D degree in computation intelligence from the Xiandian University, Xi'an, China, in 2006. He is currently a professor in Guangxi University for Nationalities. His research interests include computation intelligence, neural networks, and intelligence information processing et al. He has published 1 book, and more than 150 research papers in journals.



Qifang Luo, Associate Prof. received his B.S degree from Guangxi University of School of Computer and Electronis Information, Guangxi, China, in 1993. She is currently research interest is in computation intelligence, neural networks..



Kang Huang, M.S., received his B.S degree from College of computer Science and Technology, Guangxi University for Nationalities, Nanning, China, in 2013. He is currently research interest is in computation intelligence and applications.