

An improved fruit fly optimization algorithm based on selecting evolutionary direction intelligently *

Wu Lei[†]

*College of mechanical and electronic engineering, China University of Petroleum, No. 66, Changjiang West Road, Huangdao District, Qingdao, Shangdong Province, China
E-mail: wuleiupc@163.com*

Xiao Wensheng

College of mechanical and electronic engineering, China University of Petroleum, No. 66, Changjiang West Road, Huangdao District, Qingdao, Shangdong Province, China

Zhang Liang

College of mechanical and electronic engineering, China University of Petroleum, No. 66, Changjiang West Road, Huangdao District, Qingdao, Shangdong Province, China

Liu Qi

College of mechanical and electronic engineering, China University of Petroleum, No. 66, Changjiang West Road, Huangdao District, Qingdao, Shangdong Province, China

Wang Jingli

College of mechanical and electronic engineering, China University of Petroleum, No. 66, Changjiang West Road, Huangdao District, Qingdao, Shangdong Province, China

Received 19 July 2015

Accepted 1 December 2015

Abstract

As a novel global optimization algorithm, the fruit fly optimization algorithm FOA has been successfully applied in a variety of mathematic and engineering fields. For the purpose of accelerating the convergence speed and overcoming the shortcomings of FOA, an improved fruit fly optimization called SEDI-FOA was proposed in this paper. In the proposed SEDI-FOA, more fruit flies would fly in the search direction that was best for finding the optimal solution, or at least in a direction close to the optimal direction. Experiments were conducted on a set of 12 benchmark functions, and the results showed that SEDI-FOA performed better than other several improved FOA and frequently-used intelligence algorithms, especially in the areas of accelerating convergence and global search ability and efficiency.

Keywords: fruit fly optimization algorithm; intelligent selection; the best search direction; benchmark functions; experimental simulation

* Corresponding author: Wu Lei
Email address: wuleiupc@163.com

1. Introduction

Optimization problems are used extensively in science, engineering and business. Over the last few decades, a number of meta-heuristic optimizations have been developed. These optimizations are based on biotic factors [1] [2]. Fruit fly optimization algorithm (FOA) is a new novel optimization algorithm presented by the scholar Pan [1][3], which is inspired by the food searching behavior of fruit-fly, and has the advantage of being easy to understand, much simpler and more robust compared with the complicated optimization methods proposed by past scholars [1][3]. The fruit fly has obvious advantages over other creatures in olfactory and visual sensory perception; its olfactory senses can perceive extremely faint smells of food floating in the air, even can smell a food source accurately up to 40 kilometers away. The fruit fly's keen vision can confirm the food's location, as well as the location of other fruit flies that have gathered close to the food.

FOA has some advantages such as being easily understandable and having simple calculations [4]. Therefore, the algorithm can be widely used in science and engineering fields [4], such as in solving the steelmaking casting problem [5], continuous mathematical function optimization problems [6], GRNN parameters optimization [7], semiconductor final testing scheduling problem [8], web auction logistics service [9], design of the PID controller [10], power load forecasting [11], multidimensional knapsack problem optimization [12] and analyzing swarms of mini autonomous surface vehicles [13].

2. Literature review

To improve the search efficiency and global search ability, researchers designed several improved FOAs [2]. Yuan et.al [14] proposed a novel CFOA (chaotic-enhanced fruit fly optimization algorithm), which employs chaotic sequence to enhance the global optimization capacity of original FOA. Wang et.al [15] introduced AM-FOA (adaptive mutation fruit fly optimization algorithm). When algorithm trap in local optimum, AM-FOA selected a corresponding number of fruit flies from the population and makes mutation and then updated the global optimum. Niu et.al [4] proposed an improved fruit fly optimization algorithm DFOA (differential evolution FOA) by modifying the expression of the smell concentration judgment value

and by introducing a differential vector to replace the stochastic search. Wang et.al [2] introduced an improved FOA (IFOA) in which a new method of maintaining the population diversity was developed to enhance the exploration ability and fruit flies with better fitness values use vision to fly toward a new location, and the others fly randomly in initial search space based on swarm collaboration. Shan [16] has proposed LGMS-FOA (FOA based on linear generation mechanism of candidate solution), in which fruit fly individuals are generated by a linear mechanism. Pan [6] had put forward IFFO (improved fruit fly optimization). In IFFO, a new control parameter is introduced to tune the search scope around its swarm location adaptively and a new solution generating method is developed to enhance accuracy and convergence rate of the algorithm. But in the later stage of IFFO, while the adjacent generation optimal solution remains unchanged, it is not rational to reduce the iteration step value continually. Yuan [17] presented MFOA (multi-swarm FOA). In the MFOA approach, several sub-swarms moved independently in the search space with the aim of simultaneously exploring global optimal at the same time, and local behavior between sub-swarms are also considered [17].

But there is a common disadvantage of all these FOAs, the search direction of fruit fly is random. Obviously, it is not suit for the demand of efficiency. For the purpose of improving the search efficiency, a mechanism called "selecting the evolutionary direction intelligently" was introduced. And then, an improved fruit fly optimization algorithm called SEDI-FOA based on selecting the evolutionary direction intelligently was presented. The best search direction is fixed after evaluating the optimization results of the current group, so more fruit flies would fly in the best search direction or in a nearby direction to find the optimal solution. In SEDI-FOA the search direction of population individuals is selected according a certain mechanism which be introduced in this paper. But in other FOAs, the search direction and distance of population individuals are random. The effectiveness of SEDI-FOA is verified by a simulation experiment.

The rest of the paper is organized as follows. Section 3 describes original fruit fly optimization FOA and the improved fruit fly optimization algorithm SEDI-FOA. Simulation results and comparisons are introduced in Section 4. Section 5 covers conclusions.

3. Fruit fly optimization algorithm

3.1 Original fruit fly optimization FOA

The main idea behind the FOA technique is based upon the drosophila's biological behavior [13]: (1) The fly flies with levy flight motion; (2) It smells the potential location (attractiveness); (3) It would then taste. Parameters in the original fruit fly optimization FOA mainly include: the initial position of the group (X, Y) , the population size S , the iteration number M , and the iteration step value R .

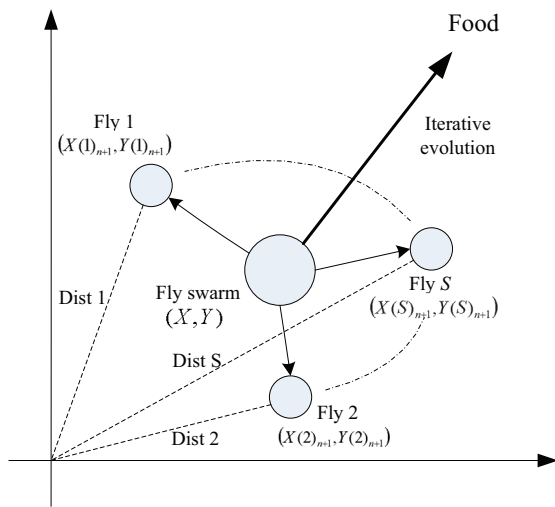


Fig.1. Iterative process of FOA

According to fruit fly foraging behavior, the iterative optimization process of fruit fly optimization algorithm can be summarized in the following steps [17](a case study of solving two-dimensional function extremum):

Step1 : Random initial fruit fly swarm location (X, Y) as shown in Fig. 1;

Step2 : Individual searches for food in random directions and distances;

$$\begin{cases} X(i)_n = X + R \cdot rand() \\ Y(i)_n = Y + R \cdot rand() \end{cases}, \quad i = 1, 2, \dots, S \quad (1)$$

$(X(i)_n, Y(i)_n)$ is the position of i -th fruit fly individual among S fruit flies.

Step3 : Since the food location cannot be known, estimate the distance $D(i)_n$ between the coordinate origin and the individuals;

$$D(i)_n = \sqrt{X(i)_n^2 + Y(i)_n^2} \quad (2)$$

The closer the location is to the origin location, the smaller the density of the food. Then, the smell concentration judgment value $S(i)_n$ is calculated, and this value is the reciprocal of the distance.

$$S(i)_n = 1 / D(i)_n \quad (3)$$

Step4 : Substitute the smell concentration judgment value $S(i)_n$ into a smell concentration judgment function F (sometimes called the Fitness function) so as to find the smell concentration $Smell(i)$ of the individual location of the fruit fly.

$$Smell(i) = F(S(i)_n) \quad (4)$$

Step5 : Identify the fruit fly with the maximal smell concentration (finding the maximal value) among the fruit fly swarm.

$$[bestSmell \ bestIndex] = \max(Smell) \quad (5)$$

Step6 : Reserve the best smell concentration value and the individual coordinate with the highest density of fruit fly individuals $(X(best)_n, Y(best)_n)$ of the n -th generation swarm. Other individuals fly toward the highest density individual using their keen vision.

$$\begin{cases} Smellbest = bestSmell \\ X = X(best)_n \\ Y = Y(best)_n \end{cases} \quad (6)$$

Step7 : Apply iterative optimization to repeat the implementation of Step 2-Step 5, and then judge if the smell concentration is superior to the previous iterative smell concentration; if so, implement Step 6. When the number of iterations is less than the maximum iterative number M , the loop terminates.

3.2 Improve fruit fly optimization algorithm SEDI-FOA

In the original FOA, the smell concentration of fruit fly individuals is judged by the distance from the fruit fly individual to the virtual food source. As we know, the distance value cannot be negative, so the original FOA cannot solve for the extremum when the extremum is in the negative domain. Therefore, in this paper, the smell concentration $Smell(i)$ is evaluated based on the decision variable $X = (x_1, x_2, \dots, x_n)$ directly as $Smell(i) = F(X(i))$ [17].

In the original FOA, the fruit fly individual search direction and distance are random. However, in the actual biological world, the swarms have intelligence and can intelligently evaluate and remember individual

search behavior. Given the biological swarm's intelligence, the individual direction should not be randomly selected in FOA. This paper presents an intelligent decision method for the FOA individual optimization direction. This method evaluates the optimization effect of the fruit fly's contemporary generation, and then the optimal search direction is determined in order to ensure that more fruit fly generations move in the optimal search direction or in the nearby direction when finding the optimal solution.

The initial position of the fruit fly swarm is random (X, Y) ; the coordinate of the highest taste density of individual is $(X(best)_n, Y(best)_n)$ after n generations of optimization. According to the FOA optimization rules, the n -th generation of a fruit fly group will converge $(X(best)_n, Y(best)_n)$, and there are:

$$\begin{cases} X_n = X(best)_n \\ Y_n = Y(best)_n \end{cases} \quad (7)$$

(X_n, Y_n) is the position coordinate of n -th generation swarm's converged position.

Define the optimal search direction \vec{F}_n of the n -th generation swarm (as shown in figure 2):

$$\vec{F}_n = [(X(best)_n - X), (Y(best)_n - Y)] \quad n = 1, 2, \dots, M \quad (8)$$

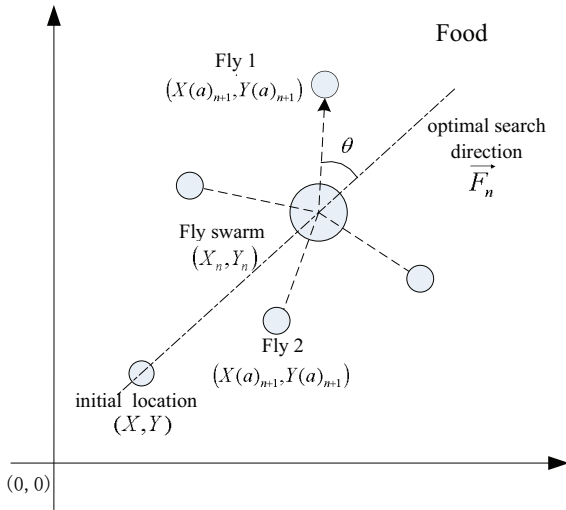


Fig.2. Optimization process of SEDI-FOA

\vec{F}_n , which is called the optimal search direction of the n -th generation swarm, is the flight direction that can enable the swarm to reach the optimal solution in the shortest time. In the evolution of the $(n+1)$ -th

generation swarm, more fruit fly individuals should fly toward \vec{F}_n or in the nearby direction.

Assume that the position coordinates of one fruit fly of the $(n+1)$ -th generation swarm is $(X(a)_{n+1}, Y(a)_{n+1})$, $a = 1, 2, \dots, S$. Define the angle θ between the flight direction and the optimal search direction \vec{F}_n as follows:

$$\cos \theta = \frac{(X(a)_{n+1} - X_n)(X_n - X) + (Y(a)_{n+1} - Y_n)(Y_n - Y)}{\sqrt{(X(a)_{n+1} - X_n)^2 + (Y(a)_{n+1} - Y_n)^2} \sqrt{(X_n - X)^2 + (Y_n - Y)^2}} \quad (9)$$

As we know, $\theta \in [0, \pi]$ distributed symmetrically on two sides of \vec{F}_n .

In the process of the $(n+1)$ -th generation optimization, the swarm flies around a center at coordinates (X_n, Y_n) . The mechanism of intelligent selection search direction

prescribes that the nearer the flight direction is to \vec{F}_n (meaning that θ is closer to 0), the greater probability of the direction being selected. Likewise, the farther the flight direction is from \vec{F}_n (meaning that θ is closer to π), the smaller the probability of the direction being selected.

Define the selection function of the individual flight direction $g(\theta)$ as follows:

$$g(\theta) = -\frac{C}{\pi} * \theta + C \quad (10)$$

C is a given constant. As we know from equation (10), when $\theta = 0$, $g(\theta) = C$. This means that individuals of the $(n+1)$ -th generation swarm fly toward \vec{F}_n in the maximum probability C . Likewise, when $\theta = \pi$, $g(\theta) = 0$; this means that individuals of $(n+1)$ -th generation swarm fly against \vec{F}_n in the probability 0.

Obviously, the value of C must insure that the probability of being selected of anyone in swarm is 100%. In the other words, any individual of fruit fly swarm can be selected and then flies toward a direction.

So, we can conclude:

$$2 \cdot \int_0^\pi g(\theta) d\theta = 2 \cdot \int_0^\pi \left(-\frac{C}{\pi} \cdot \theta + C \right) d\theta = 1 \quad (11)$$

It is easy to calculate that:

$$C = \frac{1}{\pi} \quad (12)$$

The optimization flowchart of SEDI-FOA is found in Fig. 3.

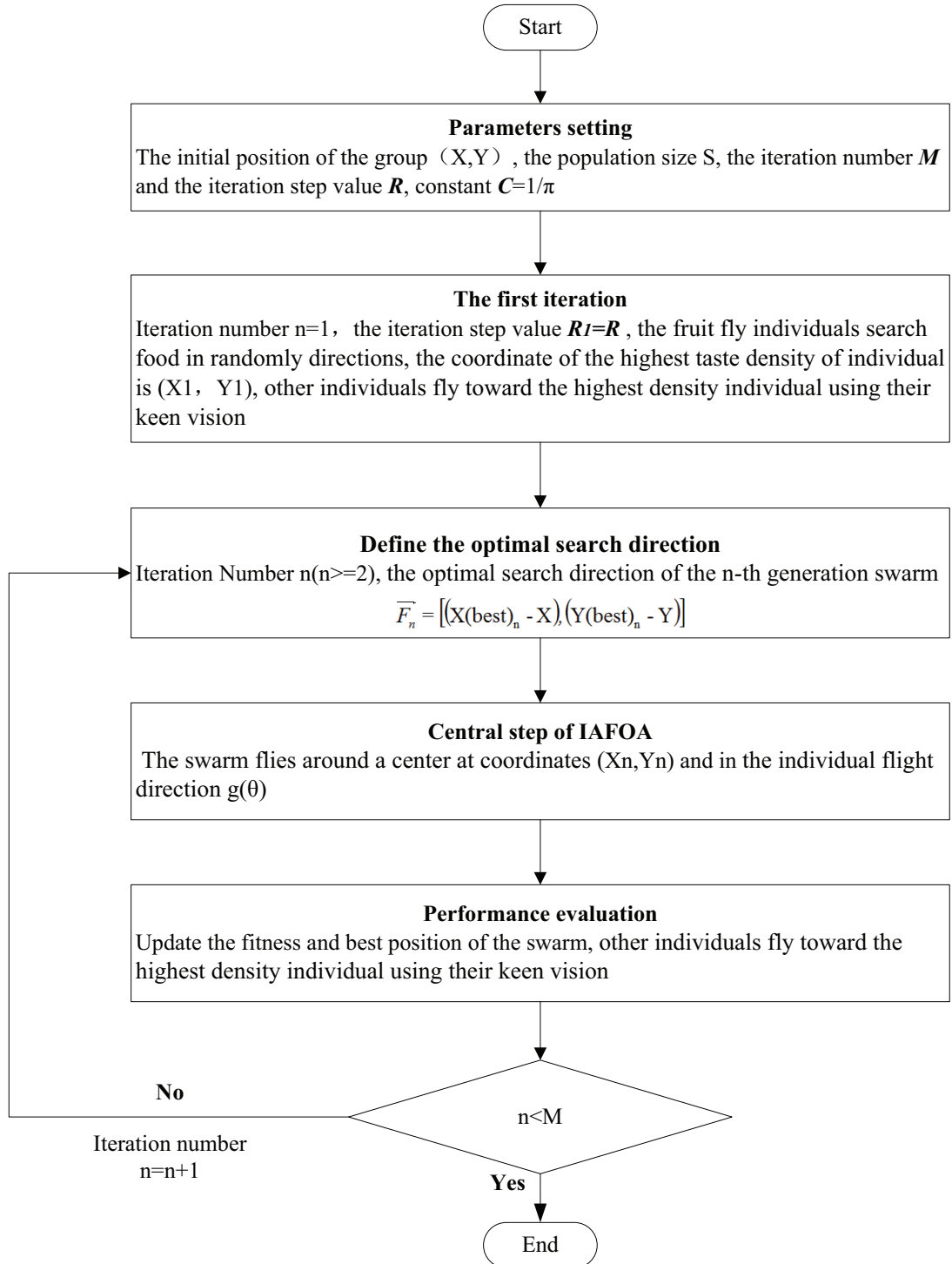


Fig.3. optimization flowchart of SEDI-FOA

4. Simulation results and comparisons

To test the performance of the proposed algorithm SEDI-FOA, we conducted experiments on 12

benchmark functions [4]. The details of these 12 functions in list in Table 1.

Table1. List of benchmark functions

Test functions	Functional expression	Region of search	Extreme value	Coordinates of extreme value point
f_1	$f(x_1, x_2) = 1 - \exp(x_1^2 + x_2^2)$	[-10,10]	0	[0,0]
f_2	$f(x_1, x_2) = 0.5 - \frac{\sin^2(\sqrt{x_1^2 + x_2^2} - 0.5)}{(1 + 0.001(x_1^2 + x_2^2))^2}$	[-20,20]	1	[0,0]
f_3	$f(x_1, x_2) = \cos(x_1)\cos(x_2)e^{-(x_1-\pi)^2-(x_2-\pi)^2}$	[-30,30]	1	$[\pi, \pi]$
f_4	$f(x) = \sum_{i=1}^3 x_i^2$	[-5.12,5.12]	0	[0,0,0]
f_5	$f(x) = \sum_{i=1}^3 (x_i - 10\cos(2\pi x_i) + 10)$	[-5,5]	0	[0,0,0]
f_6	$f(x) = \sum_{i=1}^5 x_i + \prod_{i=1}^5 x_i $	[-10,10]	0	[0,0,0,0,0]
f_7	$f(x) = 1 - \cos(2\pi\sqrt{\sum_{i=1}^5 x_i^2}) + 0.1\sqrt{\sum_{i=1}^5 x_i^2}$	[-10,10]	0	[0,0,0,0,0]
f_8	$f(x) = \frac{1}{4000} \sum_{i=1}^{15} x_i^2 - \prod_{i=1}^{15} \cos(\frac{x_i}{\sqrt{i}}) + 1$	[-10,10]	0	[0,0,...,0]
f_9	$f(x) = \frac{1}{15} \sum_{i=1}^{15} (x_i^4 - 16x_i^2 - 5x_i)$	[-10,10]	-78.332314	[2.905,2.905,...,2.905]
f_{10}	$f(x) = \sum_{i=1}^{15} x_i \sin(x_i) + 0.1x_i $	[-10,10]	0	[0,0,...,0]
f_{11}	$f(x) = -20\exp(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{30} x_i^2}) - \exp(\frac{1}{30}\sum_{i=1}^{30} \cos(2\pi x_i)) + 20 + e$	[-32,32]	0	[0,0,...,0]
f_{12}	$f(x) = \frac{1}{30} \sum_{i=1}^{30} (10(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	[-10,10]	0	(1,1,...,1)

In order to verify the effectiveness and advancement of SEDI-FOA, several improved FOAs such as DFOA [4], IFFO [6] and LGMS-FOA [16] are used to compare. Otherwise, three widely-used evolutionary algorithms are compared with the proposed SEDI-FOA approach: they are the particle swarm optimization (PSO) algorithm [18], the covariance matrix adaptation evolution strategy (CMAES) [19], the self-adaptive differential evolution (SaDE) algorithm [20]. Owing to their stochastic nature, evolutionary algorithms may arrive at solutions that are better or worse than solutions they have previously reached during their search for new solutions. For this reason, it is beneficial to use statistical tools to compare the problem-solving success of one algorithm with that of another [17]. All the seven algorithms (DFOA, IFFO,

LGMA-FOA, PSO, CMAES, SaDE and SEDI-FOA) are run 50 times for the 12 test functions.

The parameter settings of seven evolutionary algorithms are as follow:

(1)DFOA: In the experiments, we set $S=100$, $M=1000$ and $F=0.9$ according to the literature [4].

(2)IFFO: In the experiments, we set $S = 100$ and $M = 1000$ for IFFO algorithms; the other parameters are fixed as $\lambda_{max} = (UB - LB) / 2$ and $\lambda_{min} = 10^{-5}$, according to the literature [6].

(3)LGMS-FOA. LGMS-FOA generates a new individual $X_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n})$ by setting $x_{i,j} = \delta_j + \omega \times rand() \times (UB_j - LB_j)$ for $j = 1, 2, \dots, n$, where ω is a weight generated by $\omega = \omega_0 \times \alpha^{iter}$, ω_0 is the initial weight and α is the weight coefficient. According to Shan et al. [16], we set the population size $S = 100$, the iteration number $M = 1000$, $\omega_0 = 1$, and

$\alpha = 0.95$. UB and LB are the top and bottom limitations of the search region.

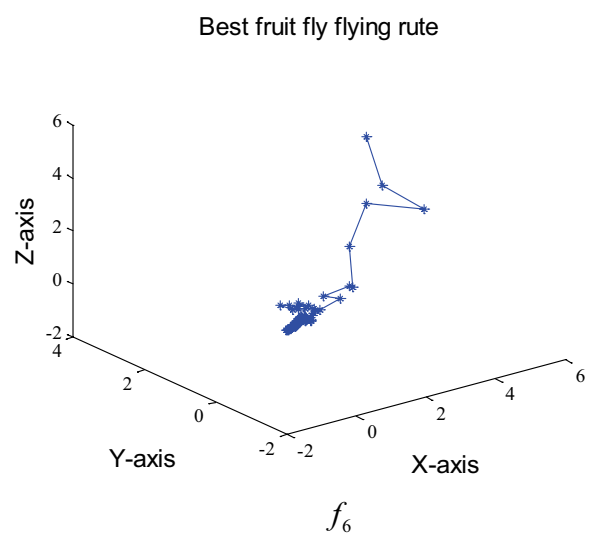
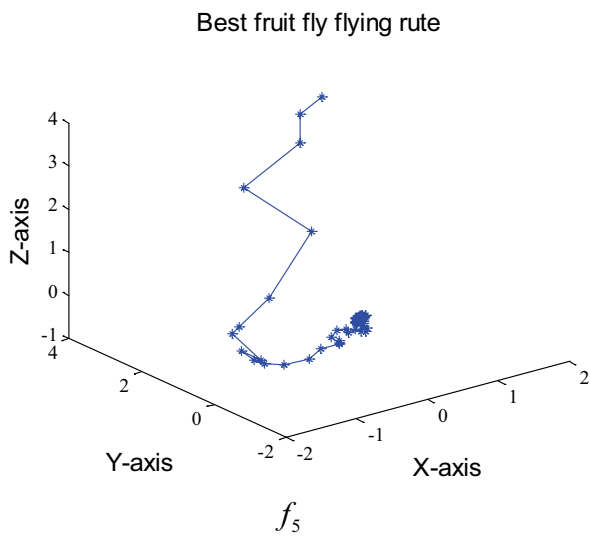
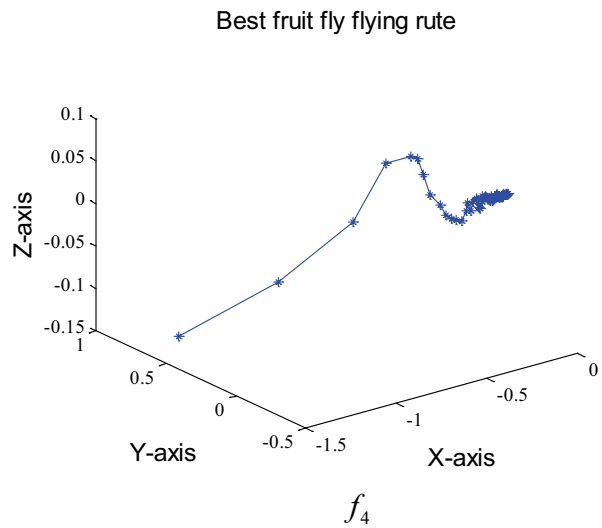
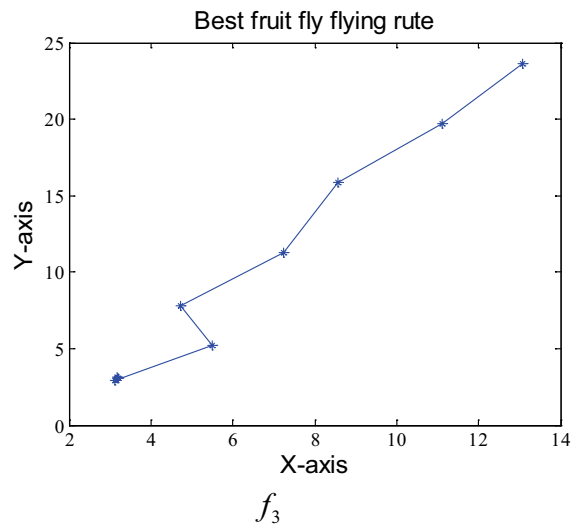
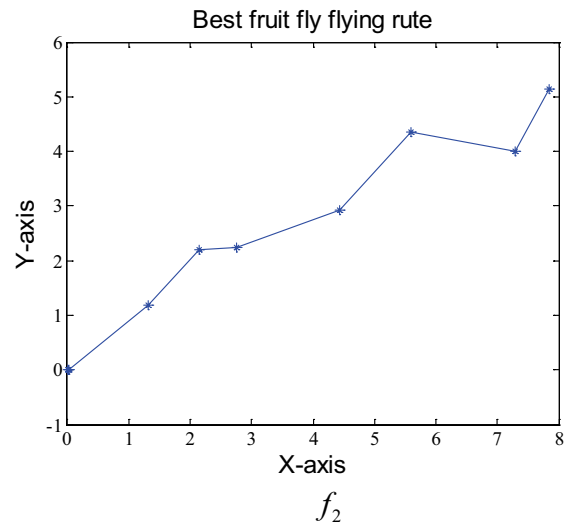
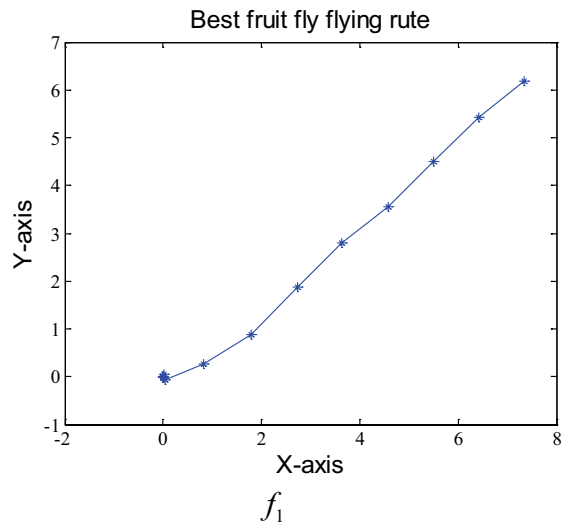
(4) The iteration number $M=1000$, the population size $S=100$. Other parameter settings of PSO, CMAES and SaDE, are according to the literature [18], [19] and [20] respectively.

(5) SEDI-FOA. In the experiments, we set $S=100$, $M=1000$, $C=1/\pi$. Besides, the iteration step value R is set according with the region of search. Usually, the value of R is one-tenth of the value of region.

The comparison of the results are reported in Table 2.

Table 2. Comparison of the results

Functions	Stats	DFOA	IFFO	LGMS-FOA	PSO	CMAES	SaDE	SEDI-FOA
f_1 (0)	Best	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	Mean	-3.84E -07	-2.34E -05	-7.02E -05	-1.94E -06	-2.41E -05	-6.11E -04	-2.01E -08
	Std.	5.3E -06	2.3E -04	1.7E -04	2.3E -05	4.8E -04	6.7E -03	5.8E -07
f_2 (1)	Best	1.0	1.0	1.0	1.0	0.999925	0.999998	1.0
	Mean	0.999965	0.999847	0.999248	0.999984	0.996273	0.997836	0.999971
	Std.	2.7E -04	3.5E -03	2.7E -03	7.0E -06	1.8E -03	1.7E -03	4.9E -05
f_3 (1)	Best	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Mean	0.999962	0.99574	0.998056	0.999927	0.995412	0.994681	0.999966
	Std.	8.6E -06	4.7E -03	3.3E -03	3.4E -04	2.7E -03	5.4E -03	5.1E -05
f_4 (0)	Best	3.09E -07	0.0	0.0	0.0	0.0	0.0	0.0
	Mean	1.5E -07	6.32E -09	2.58E -08	3.57E -08	4.81E -07	6.33E -07	8.76E -10
	Std.	2.1E -05	3.5E -08	4.8E -07	3.7E -07	2.9E -06	1.7E -06	3.7E -09
f_5 (0)	Best	6.95E -07	0.0	0.0	0.0	2.30E -06	2.27E -05	0.0
	Mean	7.36E -04	6.34E -07	4.18E -06	1.20E -03	7.01E -03	4.59E -03	5.84E -07
	Std.	1.6E -03	1.82E -04	3.49E -03	8.6E -01	2.9E -02	3.2E -02	1.75E -05
f_6 (0)	Best	5.13E -08	0.0	0.0	2.58E -05	0.0	3.47E -5	0.0
	Mean	1.03E -05	2.64E -04	3.11E -03	8.41E -02	2.14E -05	3.77E -02	3.46E -06
	Std.	1.1E -04	3.5E -02	2.8E -01	3.4E +01	6.5E -03	2.4E -01	4.7E -05
f_7 (0)	Best	0.0	0.0	3.71E -06	0.0	2.83E -05	0.0	0.0
	Mean	2.69E -03	2.74E -02	4.92E -03	2.64E -02	8.11E -03	2.37E -02	9.13E -04
	Std.	0.2489	1.2748	0.8126	2.3548	0.4571	3.4582	0.0981
f_8 (0)	Best	6.69E -06	0.0	0.0	0.0	0.0	0.0	0.0
	Mean	5.20E -03	1.27E -02	3.48E -02	3.80E -03	1.01E -03	2.77E -03	6.27E -04
	Std.	1.23E -02	1.62E -02	2.47E -01	3.0E -02	8.9E -01	2.0E -02	4.84E -03
f_9 (-78.332314)	Best	-78.307571	-78.151970	-78.265894	-78.322302	-78.312289	-78.322297	-78.332209
	Mean	-74.012450	-70.254862	-71.463281	-68.367517	-71.123059	-66.549080	-75.207516
	Std.	2.3E +01	4.1E +02	2.8E +02	7.2E +03	6.6E +01	9.9E +01	1.7E +01
f_{10} (0)	Best	3.56E -05	3.48E -04	7.16E -03	8.11E -03	6.48E -02	3.48E -04	2.67E -05
	Mean	2.34E -02	3.48E -02	4.89E -01	3.57E -01	9.79E -01	3.48E -02	6.44E -02
	Std.	5.8E +01	2.7E +02	3.7E +02	3.7E +01	4.8E +02	4.4E +01	2.3E +01
f_{11} (0)	Best	2.36E -02	3.54E -01	6.48E -02	4.58E -02	2.67E -02	4.58E -01	3.57E -02
	Mean	1.378462	5.345670	1.223694	0.845684	2.154870	3.2658410	1.0348710
	Std.	3.5E +02	2.6E +02	3.5E +01	2.3E +01	6.4E +02	3.4E +02	5.7E +01
f_{12} (0)	Best	5.61E -05	3.54E -04	2.66E -03	8.6E -04	0.0	3.0E -06	1.35E -06
	Mean	0.154110	1.332694	0.945860	1.798256	0.254623	0.387544	0.098543
	Std.	6.1E -01	2.3E +01	4.4E -01	8.8E -01	1.0E -01	3.1E -01	7.83E -03



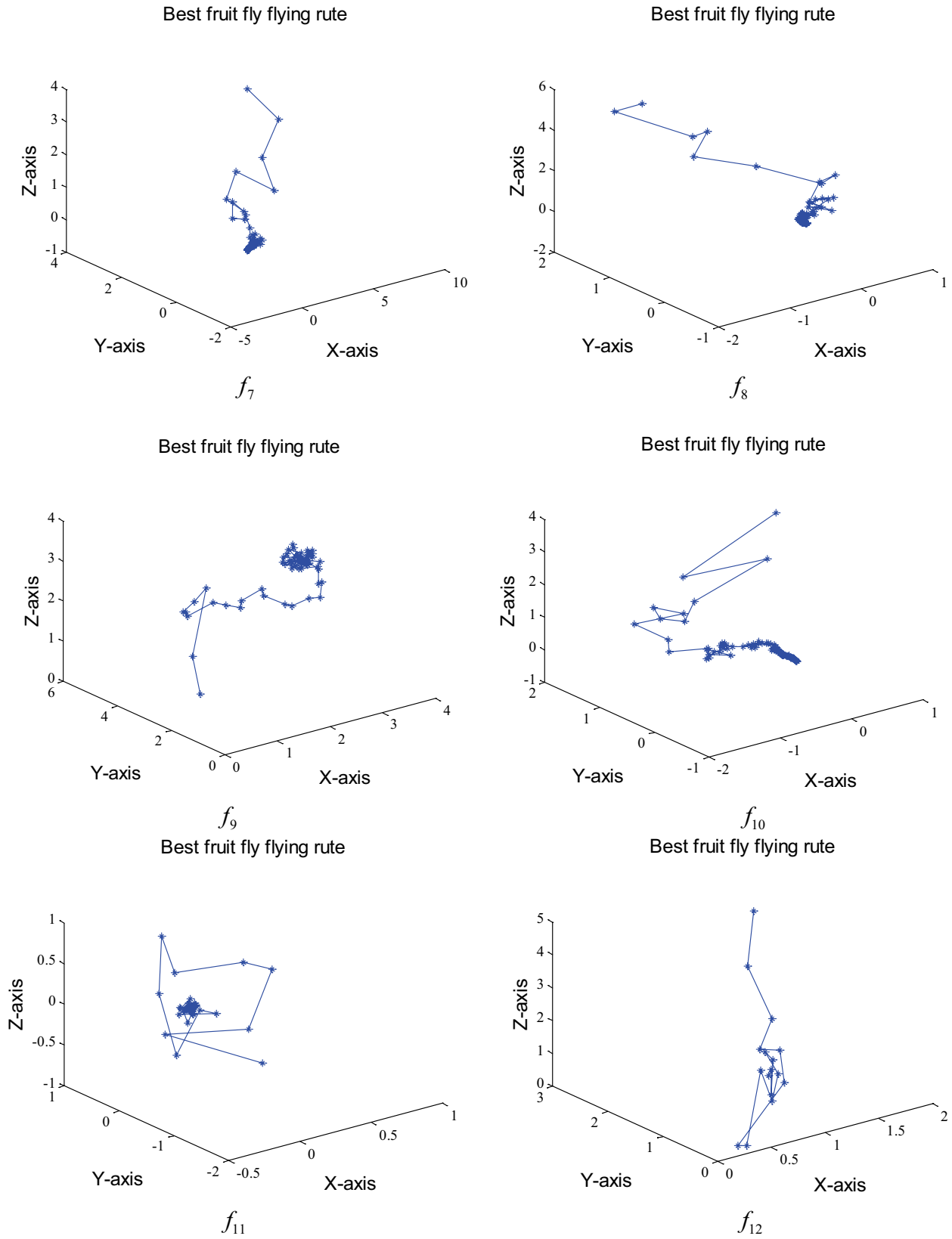


Fig.4. Best fruit fly flying route for benchmark functions using SEDI-FOA approach

In Table 3, where the “Best” means the optimal objective function value, the “Mean” reflects the precision and rate of convergence, and the “Std.” reflects the stability and robustness of the algorithm [21].

From the Table 3 we can know that SEDI-FOA has the best performance in finding the “Best”, “Mean”, “Std.” of $f_1, f_4, f_5, f_6, f_7, f_8$ and f_9 . Refer to f_2 , SEDI-FOA has found the “Best”, but the performance in “Mean” and “Std.” is ranking only second to PSO. SEDI-FOA has best performance in “Best” and “Mean” of f_3 , but the performance in “Std.” is worse than DFOA. Besides, SEDI-FOA has best performance in “Best” and “Std.” of f_{10} , “Best” of f_{11} , “Mean” and “Std.” of f_{12} . Taken together, SEDI-FOA achieved 29 optimum value in 36 testing items. So we can concluded that SEDI-FOA has made great improvement in finding the optimal value.

The best fruit fly flying routes for benchmark functions in this simulation using the proposed SEDI-FOA approach are illustrated in Fig. 4, which shows that the best fruit fly can direct the global optimal solution in an efficient way.

5. Conclusion

In order to overcome the defect of original fruit fly optimization FOA, an improved FOA called SEDI-FOA based on selecting the evolutionary direction intelligently was proposed. Fruit fly of the swarms fly towards the best search direction or along a nearby direction to find the optimal solution, so that the efficiency of SEDI-FOA will increase. Simulation results showed that the SEDI-FOA achieved 29 optimum value in 36 testing items compared with three improved FOAs and three evolutionary algorithms.

But SEDI-FOA still has many problems that need to be improved, such as solving for the high-dimensional continuity function, identifying how to code when solving engineering problems, learning how to realize rapid convergence in practical problems and so on.

Acknowledgements

This work was supported in part by the project foundation of china Ministry of industry and information technology “Research of gordian technique of deep-water semi-submersible platforms”, and the

National 863 plan project foundation of china “Key technology research of automated processing of deep-water drilling rig and string” (No. 2012AA09A203), and Project of scientific and technological achievements of Jiangsu province “Research and industrialization of the key techniques of drilling string used in marine deep water”.

References

1. W.T. Pan, A New Fruit Fly Optimization Algorithm: Taking the Financial Distress Model as An Example, *Knowledge-Based Systems*. **26**(2012) 69–74.
2. L. Wang, Y.L. Shi, S. Liu, An improved fruit fly optimization algorithm and its application to joint replenishment problems, *Expert Systems with Applications*. **42**(9)(2015)4310–4323.
3. H.D. Dai, A.L. Liu, J.H. Lu, et al, Optimization about the layout of IMUs in large ship based on fruit fly optimization algorithm, *Optik*. **126**(4)(2015)490–493.
4. J.W. Niu, W.M. Zhong, Y. Liang, et al, Fruit fly optimization algorithm based on differential evolution and its application on gasification process operation optimization, *Knowledge-Based Systems*. **88**(2015)253–263.
5. J.Q. Li, Q.K. Pan, K. Mao, et al, Solving the steelmaking casting problem using an effective fruit fly optimization algorithm, *Knowledge-Based Systems*. **72**(2014)28–36.
6. Q.K. Pan, H.Y. Sang, J.H. Duan, et al. An improved fruit fly optimization algorithm for continuous function optimization problems. *Knowledge-Based Systems*. **62** (2014)69–83.
7. W.T. Pan, Using modified fruit fly optimization algorithm to perform the function test and case studies, *Connection Science*. **25**(2–3)(2013)151–160.
8. X.L. Zheng, L. Wang, S.Y. Wang, A novel fruit fly optimization algorithm for the semiconductor final testing scheduling problem, *Knowledge-Based Systems*. **57**(2014)95–103.
9. S.M. Lin, Analysis of service satisfaction in web auction logistics service using a combination of fruit fly optimization algorithm and general regression neural network, *Neural Computing and Applications*. **22**(2–3)(2013)783–791.
10. C. Li, S. Xu, W. Li, et al, A novel modified fly optimization algorithm for designing the self-tuning proportional integral derivative controller, *Journal of Convergence Information Technology*. **7**(16)(2012)69–77.

11. H.Z. Li, S. Guo, C.J. Li, et al, A hybrid annual power load forecasting model based on generalized regression neural network with fruit fly optimization algorithm, *Knowledge-Based Systems*. **37**(2013)378–387..
12. L. Wang, X.L. Zheng, S.Y. Wang, A novel binary fruit fly optimization algorithm for solving the multidimensional knapsack problem, *Knowledge-Based Systems*. **48**(2013)17–23.
13. Z.Z. Abidin, M.R. Arshad, U.K. Ngah, A simulation based fly optimization algorithm for swarms of mini autonomous surface vehicles application, *Indian Journal of Geo-Marine Science*. **40**(2)(2011)250–266.
14. X.F. Yuan, Y.M. Liu, Y.Z. Xiang, et al, Parameter identification of BIPT system using chaotic-enhanced fruit fly optimization algorithm, *Applied Mathematics and Computation*. **268**(2015)1267–1281.
15. W.C. Wang, X.G. Liu, Melt index prediction by least squares support vector machines with an adaptive mutation fruit fly optimization algorithm, *Chemometrics and Intelligent Laboratory Systems*. **141**(2015)79–87.
16. D. Shan, G.H. Cao, H.J. Dong, LGMS-FOA: An Improved Fruit Fly Optimization Algorithm for Solving Optimization Problems. *Mathematical Problems in Engineering*. **2013**(Article ID108768).
17. X.F. Yuan, X.S. Dai, J.Y. Zhao, et al. On a novel multi-swarm fruit fly optimization algorithm and its application. *Applied Mathematics and Computation*. **233**(2014)260–271.
18. R. Thangaraj, M. Pant, A. Abraham, P. Bouvry, Particle swarm optimization: hybridization perspectives and experimental illustrations, *Applied Mathematics and Computation*. **217**(12)(2011)5208–5226.
19. C. Igel, N. Hansen, S. Roth, Covariance matrix adaptation for multi-objective optimization, *Evolutionary Computation*. **15**(1)(2007)1–28.
20. J. Brest, S. Greiner, B. Boskovic, et al. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, *IEEE Transactions on Evolutionary Computation*. **10**(6) (2006)646–657.
21. W.S. Xiao, L. Wu, X. Tian et al. Applying a new adaptive genetic algorithm to study the layout of drilling equipment in semisubmersible drilling platforms [J], *Mathematical Problems in Engineering*. **2015**(Article ID146902).