A Hybrid Firefly Algorithm Based on Orthogonal Opposition

YingYing Ge
College of Computer Science and Technology, Wuhan University of Science and Technology
Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System
Wuhan, China
xxmu20141280128@163.com

Jun Li*
College of Computer Science and Technology, Wuhan University of Science and Technology
Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System
Wuhan, China
lijun@wust.edu.cn

ChenYing Meng
College of Computer Science and Technology, Wuhan University of Science and Technology
Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System
Wuhan, China
2974331809 @qq.com

***Abstract*—Firefly Algorithm (FA) may suffer from lower convergence accuracy when solving high-dimensional and complex optimization problems. To solve this problem, a completely new strategy named Hybrid Firefly Algorithm Based on Orthogonal Opposition (OHFA) is proposed. In OHFA, we perform differential evolution (DE) on brighter fireflies (j) and orthogonal opposition-based learning (OOBL) on globally optimal firefly to improve the search ability of the population. Besides, in high-dimensional and large-scale search space, there is an obvious long Euclidean distance between fireflies, which reduces attraction in movement. Therefore, OHFA adopts a new movement to improve the application of the firefly algorithm in high-dimensional space. Computational results show the effectiveness of OOBL and DE. Our findings suggest that OHFA achieves better solutions than other proposed algorithms on most of the test functions.**

Keywords—Firefly algorithm, Orthogonal opposition-based learning, Differential evolution, Global search capabilities, Convergence accuracy

I. INTRODUCTION

Proposed by Xin-She Yang in 2008 [1], the firefly algorithm (FA) is a swarm intelligence algorithm that has emerged in recent years. Inspired by nature, it simulates the social behavior of fireflies that attract each other due to brightness to find the optimal solution of the function. Concerning multimodal global optimization problems, the firefly algorithm is more efficient than particle swarm optimization (PSO), differential evolution algorithm (DE), and genetic algorithm (GA) [2] [3]. Therefore, the firefly algorithm and its variants have been applied to optimization problems in engineering and science including image and hydropower plant group optimal dispatching [4].

Scholars have improved the firefly algorithm in recent years. Their work falls into the following groups: (1) Fine-tuning the parameters. Yu et al. [5] proposed a variable step size firefly algorithm(VSSF) that changes the size of alpha in each iteration. Yang et al. [6] proposed the Levy-flight firefly, which combines the Levy-flights with search strategy via the Firefly algorithm to improve the global search capability. Wang et al. [7] proposed an adaptive adjustment strategy to reanalyze the step size and attractiveness between fireflies with a comprehensive analysis of their characteristics. A.H. Gandomi et al. [8] introduced chaos mapping technology into the firefly algorithm and changed the attraction parameter. Although fine-tuning helps improve the algorithm, the effect is barely measurable about the complex optimization problems. (2) Adjusting the attraction model. Wang et al. [9] [10] proposed a random attraction model (RaFA) and neighbor attraction model (NaFA). In [9], a firefly was randomly selected as the attractor; In [10], the population is sorted into a ring Topology and the K neighbors of the firefly are used as attractors, which avoids oscillations during the search process and high computational time complexity. Wang et al. [11] proposed the Elite Neighbor Attraction Model (pFA). Wu et al. [12] proposed the best neighbor-guided search strategy model (BNFA). This algorithm randomly selects N fireflies, while the current fireflies are only attracted by the best individuals among N fireflies. Zhang et al. [13] used the adaptive grouping strategy to solve the continuous space optimization problem. These models reduce the time complexity at the expense of convergence accuracy. (3) Integrating other technologies into the Firefly algorithm. Zhou et al. [14] integrated the orthogonal opposition-based learning strategy (OOBL) into the Firefly algorithm, which significantly improved algorithms and was applied to routing optimization in the data grid. Zhang et al. [15] applied the improved firefly algorithm to neural networks, which improved the convergence speed and accuracy of BP neural networks. Mohammadiyan et al. [16] integrated the genetic algorithm with the firefly algorithm, which was better than both the genetic algorithm and the firefly algorithm. Other technologies are utilized by these adjustments to which the proposed algorithm belongs in this paper, which significantly improves the algorithm.

In the field of FA, there are a few of latest studies. Wu et al. [17]proposed an improved FA for global continuous optimization problems. This algorithm provides an adaptive logarithmic spiral-Levy FA (AD-IFA) that strengthens the LF-FA's local exploitation and accelerates its convergence.

* Corresponding author: Jun Li

Liu et al [18] proposed a dynamic adaptive firefly algorithm with globally orientation. It has a global-oriented moving mechanism and can dynamically adjust the step size and attractiveness. Those studies overcome the disadvantages of the standard firefly algorithm and improve the convergence rate and solution precision.

In the above literature, the standard firefly algorithm has been evolved in different ways, and the performance of the algorithm has been improved to a certain extent. However, there is still a problem. The brighter firefly has a very small probability of updating the position, which wastes computing resources and reduces convergence accuracy. Under those cases, A hybrid FA called HFA proposed in [19] applies differential evolution learning to the brighter firefly algorithm, which improves the search ability of populations. HFA increases the brighter ones' movement. However, the global optimal firefly does not perform, which causes a local optimum. On top of that, its exploration of high-dimensional domain space is insufficient.

Inspired by [19], this paper will build a hybrid firefly algorithm based on orthogonal opposition-based learning (OHFA) to avoid a lot of the overhead and improve convergence accuracy. In OHFA, first, the darker fireflies are attracted by the brighter ones. The brighter ones are differentially evolved. Second, the global optimal fireflies are orthogonal opposition-based learning, which specifically maximizes the search ability of the elite individuals in the population. Finally, we modify the formula for the distance between fireflies to make the most of the attraction. OHFA verified by 15 test functions, and the results show that OHFA has significantly improved convergence accuracy and speed. As such, we conducted this research to contribute our share towards the realm of swarm intelligence algorithms with a new approach to improving how fireflies update their states.

The content is organized as follows. The first part presents a brief review of the standard firefly algorithm. The second part introduces the techniques and formulas involved in OHFA. The third part is the verification process. The last part summarizes this paper and offers our vision of the future.

II. Basic Knowledge

A. Firefly Algorithm(FA)

Inspired by nature, the firefly algorithm, proposed by Yang, is based on flashing patterns and social behavior of fireflies. To apply it to practical problems, Yang made the following assumptions:

- Sex is not taken into account.
- The attractiveness is proportional to the brightness. The darker will move to the brighter fireflies, and the attractiveness will decrease as their distance increases.
- The brightness is determined by the fitness returned by the objective function.

In the firefly algorithm, the brightness is determined by the fitness function. The brightness changes according to the distance. The formula is as follows:

$$I = I_0 e^{-\gamma r_{ij}^2} \quad (1)$$

Where I_0 is the brightness at $r=0$, γ is the light absorption coefficient.

The attractiveness is proportional to the brightness. It can be defined as follows:

$$\beta = \beta_0 e^{-\gamma r_{ij}^2} \quad (2)$$

Where β_0 is the attractiveness at $r=0$, r_{ij} is the distance between firefly x_i and firefly x_j:

$$\mathrm{r}_{ij} = \| x_j - x_i \|_2 = \sqrt{\sum_{d=1}^{D} (x_{jd} - x_{id})^2} \quad (3)$$

Where D is the number of dimensions.

For two fireflies x_i and x_j, if x_j is brighter than x_i, then x_i will move towards x_j. The formula for movement as follows:

$$x_i(t+1) = x_i(t) + \beta_0 e^{-\gamma r_{ij}^2} (x_j(t) - x_i(t)) + \alpha \varepsilon_i \quad (4)$$

Where x_i、 x_j is the coordinate of firefly x_i and firefly x_j, α is a random number with the range of [0,1], ε_i is a Gaussian random number. t is the number of iteration.

B. Opposition-Bassed Learning(OBL)

Proposed by Tizhoosh et al [20], OBL evaluates the current point and its reverse point simultaneously with a merit-based system. The definition of the reverse point is as follows:

$$ox = a + b - x \quad (5)$$

Where x is a real number defined on [a, b], ox is the inverse point of x.

Proposed by Rahnamayan et al [21], Centroid Opposition learning(CO) is based on OBL, which makes the most population information. The inverse point of gravity of the group is defined as follows:

$$M_j = \frac{1}{N} \sum_{j=1}^{D} x_{ij}, i = 1,2,3,...,N \quad (6)$$

Where $(x_1, x_2, ..., x_N)$ is the number of N particles in the D dimension. D is the number of dimensions.

Definition based on gravity center reverse point:

$$OX_i = 2M - X_i \quad (7)$$

Where OX_i is the centroid opposition point of X_i.

To limit the reverse point to the search space, the definition of a dynamic boundary is proposed. When the reverse point exceeds the boundary, the reverse point is recalculated according to formula (8). Where *rand* is a random number between [0,1].

$$ox_{ij} = \begin{cases} a_j + rand(M - a_j), x_{ij} < a_j \\ M + rand(b_j - M), x_{ij} > b_j \end{cases} \quad (8)$$

$$a_j = \min(X_j), b_j = \max(X_j) \quad (9)$$

C. *Differential Evolution(DE)*

DE is a heuristic algorithm that performs three basic operations: differential mutation, hybridization, and selection. As mentioned earlier, in standard FA, the brighter fireflies move with a low probability, which wastes population search resources. Differential evolution mutation of brighter fireflies can improve the search ability of the population.

Assuming that x_j is a brighter individual, its mutant v_j can be generated by the following equation:

$$v_{jd} = x_{t1,d} + F0 * (x_{t2,d} - x_{t3,d}) \tag{10}$$

Where *t1, t2, t3* are mutually different random integers between 1 and *NP*. The parameter *F0* is a scale factor that controls the amplification of the difference vector.

The crossover strategy is a new experimental individual U_j obtained by recombining the mutant v_j with the current individual x_j according to the crossover probability *CR*. The corresponding equation is as follows:

$$U_{jd} = \begin{cases} v_{jd}, if\,(rand < CR \vee d = l) \\ x_{jd}, otherwise \end{cases} \tag{11}$$

Where $CR \in (0,1)$ is called crossover rate; $rand$ is a random value between 0 and 1; $l \in (1, 2, \ldots, D)$ is a random index.

The selection is to compare the fitness value of the experimental individual with the original one, and the individual with the best fitness value is preserved in the next generation, otherwise, it is discarded.

$$X_j = \begin{cases} U_j, if\,(f(U_j) < f(X_j) \\ X_j, otherwise \end{cases} \tag{12}$$

III. Hybrid Firefly Algorithm Based on Orthogonal Opposition (OHFA)

The standard firefly algorithm and most of its variants [2-19] use a full attraction model which suggests that each firefly moves to the brighter ones. However, this model will cause the fireflies to oscillate. Besides, the brighter fireflies will move with a very low probability, and the global optimal firefly does not work as expected, which wastes computing resources and reduces convergence accuracy. If the brighter fireflies can increase the probability of movement, the population can find the optimal solution faster and more accurately. Therefore, we propose an orthogonal opposition-based hybrid firefly algorithm. First, we compare the brightness of each firefly. With the differential evolution strategy introduced, the darker fireflies are attracted by the brighter ones, which specifically maximizes the search ability of the elite individuals. Therefore, the local search ability of the algorithm is improved. Second, the OOBL strategy is used to update the global optimal individual to expand the population search range, thereby improving the global search ability of the population. Finally, the formula for the distance between fireflies was changed to scale down the fireflies. In the early stage of iteration, the fireflies can also be seen by each other, thereby improving the FA in high dimensions.

A. *Orthogonal Opposition-Bassed Learning(OOBL)*

OBL takes the reverse values of all the dimensions when calculating the reverse individuals. Although this can improve the population search ability, the reverse individuals are overly dependent on coordinates. Besides, for each individual, not all the reverse values on all dimensions are better than the original ones. Therefore, the OOBL strategy that combines orthogonal experimental design and OOBL is proposed. The OOBL strategy maximizes the information hidden in individuals and their inverse individuals. We conducted orthogonal experiments to generate a set of candidates that select inverse values in some parts of dimensions to obtain useful information from individuals and reverse ones. The steps of the OOBL strategy are shown in Table I.

TABLE I. ORTHOGONAL OPPOSITION-BASED LEARNING STRATEGY

Input: Test Function, Population X, Index of the globally optimal individual(ind)	
Output: New Population X	
Steps:	
1)	Make the orthogonal table L
2)	Calculate the gravity of the current population according to equation (6)
3)	Calculate the reverse individual of the globally optimal individual according to equation (7)
4)	Boundary check according to equation (8)
5)	for i = 1:H
6)	for j = 1:D
7)	if L(i , j)==1
8)	oox(i , j)=X(ind,j)
9)	else
10)	oox(i , j)=ox(j)
11)	end if
12)	end for
13)	end for
14)	Evaluate orthogonal inverse candidate solutions
15)	Choose the top n best individuals as the new population from the current population X and the inverse candidate solution.

a. H: the rows of table L.

b. oox: the opposition population of globally optimal

c. ox: the opposition value of globally optimal in d dimension

In OOBL, we made an orthogonal table of 2 levels of *D* factors. The number of dimensions of the problem corresponds to the factors (i.e. the number of columns) in the orthogonal table, and the values of the individuals and inverse ones in each dimension are treated as the level of each factor. In the orthogonal table, when the number of elements is 1, it corresponds to the value of the individual; when the number of elements is 2, it corresponds to the value of the inverse one.

The most important part of OOBL strategy is to find an experimental solution M. An individual can find M experimental solutions (that is, the number of rows of the orthogonal table), the formula is as follows:

$$M = 2^{\lceil \log_2 (D+1) \rceil} \tag{13}$$

B. Modification of the Movement Strategy

In FA, the distance between the two fireflies is calculated based on the Euclidean distance. However, this distance is extremely long in high-dimensional space, thereby reducing attractiveness, which decreases the algorithm's search ability. To avoid the problem, a new distance formula is proposed as follows: (the movement of equation (4) is preserved).

$$r_{ij} = \sqrt{\sum_{d=1}^{D} \left(\frac{x_{jd}}{scale} - \frac{x_{id}}{scale}\right)^2} \quad (14)$$

$$scale = abs(X_{\max} - X_{\min})$$

Where *scale* is the length of the search space. X_{min} and X_{max} are the maximum and minimum values of search space respectively. x_i and x_j are the coordinates of fireflies; D is the number of dimensions.

In the early of the iteration, the distance between fireflies is so long that larger random perturbations to seek a better position are possible. During the later stage of the iteration, most fireflies gather at the optimal position. The deviation caused by the random perturbations needs to be reduced, so we update the value of α dynamically:

$$\alpha = \alpha \bullet \delta_i \quad (15)$$

C. Framework of OHFA

In OHFA, *NP* is the population size, *MaxG* is the maximum number of iterations; *Xmax* and *Xmin* are the maximum and minimum of the search range respectively. The implementation details are shown in Table II.

TABLE II. THE PROPOSED OHFA ALGORITHM

```
Randomly initialize population with NP individuals
Evaluate the objective function f(x_i);
While (G<MaxG)
  Sort the population by fitness;
  for i = 1:NP
    for j = 1:i
     if firefly j is brighter than firefly i
        Update the distance r_ij according to Eq. (14);
        Move firefly i towards j according to Eq. (4);
        Evaluate the new solution;
    else
        Generate a mutant V_j based on X_j via Eq. (10)
        Generate U_j via Eq. (11);
        Check the boundary;
        Evaluate U_j;
        Select the new individual X_j via Eq. (12);
     end if
    end for
  end for
  Perform OOBL strategy on globally optimal
according to algorithm 1;
  Evaluate the new population;
  Update the value of α via Eq. (15);
end while
```

IV. EXPERIMENTAL STUDY

A. Parameters

To evaluate the OHFA comprehensively, we compare its convergence accuracy with the standard firefly algorithm and its variants proposed in recent years including the chaotic firefly algorithm CFA [8], the neighbor attraction model firefly algorithm NaFA [10], orthogonal opposition-based firefly algorithm OOFA [14], hybrid firefly algorithm HFA [19], and memetic firefly algorithm MFA [22]. We mainly compare the average, minimum and variance of each algorithm in solving the objective function. The iteration would break on the condition that the maximum number of iterations is reached.

To make it fair, for all algorithms, we set parameters as below: population size *NP*=40, the dimensions size *D*=30 or *D*=100, maximum iteration numbers *MaxG*=5000, random perturbation coefficient α= 0.2 and σ = 0.97, Light absorption coefficient $\gamma = 1$, brightness $\beta 0 = 1$. For OHFA, cross probability *Cr*= 0.9, scaling factor *F0*= 0.5. Other parameters are consistent with their literature.

Experimental environment: Processor Intel® Core ™ I7-4720HQ @ 3.60GHz, RAM 8GB, Win10 64-bit operating system, and MATLAB R2018b.

B. Test Functions

15 test functions of CEC2015 [23] are selected to verify the effectiveness of the proposed algorithm. Details of the functions are shown in Table III. Among them, f1 and f2 are unimodal functions; f3 to f5 are simple multimodal functions; f6 to f8 are mixed-functions; f9 to f15 are compound functions.

TABLE III. CEC2015 TEST FUNCTIONS

Function	*Function name*	*Dimension*	*Domain*	*Best*
F1	Rotated High Conditioned Elliptic Function	30/100	[-100,100]	100
F2	Rotated Cigar Function	30/100	[-100,100]	200
F3	Shifted and Rotated Ackley's Function	30/100	[-100,100]	300
F4	Shifted and Rotated Rastrigin's Function	30/100	[-100,100]	400
F5	Shifted and Rotated Schwefel's Function	30/100	[-100,100]	500
F6	Hybrid Function 1 (N=3)	30/100	[-100,100]	600
F7	Hybrid Function 2 (N=4)	30/100	[-100,100]	700
F8	Hybrid Function 3 (N=5)	30/100	[-100,100]	800
F9	Composition Function 1 (N=3)	30/100	[-100,100]	900
F10	Composition Function 2 (N=3)	30/100	[-100,100]	1000
F11	Composition Function 3 (N=5)	30/100	[-100,100]	1100
F12	Composition Function 4 (N=5)	30/100	[-100,100]	1200
F13	Composition Function 5 (N=5)	30/100	[-100,100]	1300
F14	Composition Function 6 (N=7)	30/100	[-100,100]	1400
F15	Composition Function 7 (N=10)	30/100	[-100,100]	1500

C. Experimental Results and Analysis

The experiment independently runs each algorithm 20 times in 30 dimensions and 100 dimensions. The two dimensions are selected to add to the difficulties of optimization, thereby verifying the ability of the OHFA algorithm to deal with various complex problems. The averages, minimums, and variances of functions are calculated. The results are shown in Table IV, Table V, and Figure 1. Table IV compares the algorithms in 30 dimensions, while table V compares them in 100 dimensions. The bold parts indicate the optimal solution on the same function. Figure 1 is the convergence curve of each algorithm for the test functions F1, F3, F8, and F10 in the 30 dimensions. Because of the lack of space, only some representative functions are selected.

1) Comparison of Solution Accuracy

In Table IV, When we take into consideration the average, minimum, and variance of the test functions F1, F2 (single-peak function), F6, F7, F8 (composite function), and F10 (complex function), the OHFA algorithm is better than other algorithms. For F3 and F9 test functions, the OHFA is at the same level as other algorithms on the average and minimum, while it is better than other algorithms on variance. For the F14 and F15 test functions, the OHFA is better than other algorithms on average and minimum values. In summary, if the dimensional size is 30, the OHFA generally has 10 test functions is generally superior to (at least equal to) other algorithms in average, which indicates that differential evolution learning for brighter individuals and the OOBL strategy for the best individuals maximize the elite individuals, thereby increasing the convergence and stability of the algorithm.

In Table V, when we take into consideration the test functions F1, F2, F3, F9, F10 and F12, the OHFA is better than (at least equal to) other algorithms on the average, minimum, and variance. For the test function F6, F7 and F14, OHFA is superior to other algorithms on the minimum and average. This shows that the change of the OHFA's movement strategy makes it easier to solve high-dimensional problems. It further verifies that the OHFA has good adaptability and stability to optimization.

TABLE IV. COMPARISON OF OHFA WITH OTHERS ALGORITHM (D=30)

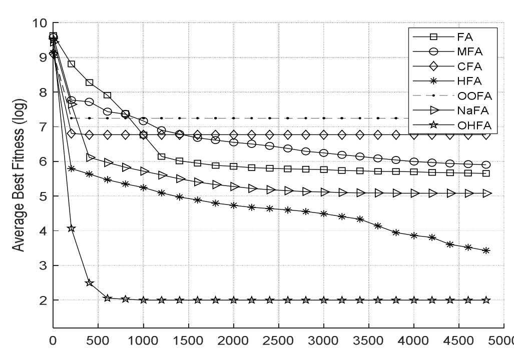
Function	***Indicators***	***FA***	***CFA***	***MFA***	***OOFA***	***HFA***	***NaFA***	***OHFA***
F1	mean	1.78E+06	2.29E+07	2.47E+06	3.39E+07	2.12E+04	1.03E+07	**1.02E+02**
	min	5.04E+05	3.00E+06	5.71E+05	1.85E+07	1.47E+03	5.55E+05	**1.00E+02**
	variance	3.77E+12	1.53E+14	1.08E+12	9.58E+13	3.31E+08	9.54E+13	**8.41E+01**
F2	mean	1.73E+05	4.84E+03	3.73E+03	4.33E+05	3.23E+03	3.58E+08	**2.00E+02**
	min	1.28E+05	3.25E+02	4.51E+02	2.57E+02	2.00E+02	2.00E+02	**2.00E+02**
	variance	1.37E+09	1.38E+07	1.04E+07	6.37E+11	1.36E+07	4.89E+17	**1.30E-17**
F3	mean	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**
	min	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**
	variance	1.73E-03	4.69E-04	1.80E-08	1.33E-01	5.35E-09	9.80E-09	**1.13E-12**
F4	mean	7.41E+02	5.29E+02	**4.68E+02**	5.41E+02	6.23E+02	5.93E+02	5.66E+02
	min	6.27E+02	4.62E+02	**4.33E+02**	4.98E+02	5.48E+02	5.52E+02	4.99E+02
	variance	3.53E+03	1.21E+03	6.37E+02	3.72E+02	2.06E+03	6.18E+02	1.57E+03
F5	mean	5.15E+03	4.26E+03	**3.28E+03**	3.50E+03	4.61E+03	4.86E+03	4.13E+03
	min	3.73E+03	3.15E+03	**2.29E+03**	3.15E+03	3.05E+03	3.99E+03	3.21E+03
	variance	3.81E+05	4.44E+05	**2.29E+05**	7.16E+04	7.38E+05	2.95E+05	3.29E+05
F6	mean	9.88E+04	1.57E+06	2.19E+05	7.22E+05	3.64E+03	7.50E+04	**1.93E+03**
	min	3.08E+04	8.89E+04	1.33E+04	1.61E+05	1.67E+03	1.40E+04	**1.37E+03**
	variance	4.62E+09	9.49E+11	2.03E+10	3.26E+11	2.67E+06	4.26E+09	**1.40E+05**
F7	mean	7.90E+02	7.18E+02	7.14E+02	7.17E+02	7.15E+02	8.09E+02	**7.12E+02**
	min	7.35E+02	7.14E+02	7.10E+02	7.16E+02	7.11E+02	7.55E+02	**7.06E+02**
	variance	1.65E+03	2.50E+01	3.37E+00	8.88E-01	5.73E+00	1.48E+03	**1.65E+01**
F8	mean	2.63E+04	4.19E+05	9.13E+04	2.68E+05	2.91E+03	2.29E+04	**1.76E+03**
	min	1.33E+04	3.98E+04	2.75E+04	5.68E+04	2.08E+03	8.68E+03	**1.25E+03**
	variance	1.56E+08	1.71E+11	4.36E+09	2.44E+10	7.92E+05	1.15E+08	**9.41E+04**
F9	mean	1.25E+03	1.02E+03	**1.01E+03**	**1.01E+03**	1.04E+03	1.21E+03	**1.01E+03**
	min	1.01E+03	1.00E+03	**1.00E+03**	**1.00E+03**	1.00E+03	1.02E+03	1.01E+03
	variance	5.94E+04	3.83E+03	1.40E+03	**1.79E-01**	1.42E+04	2.88E+04	1.18E+00
F10	mean	1.55E+05	2.01E+06	3.50E+05	9.53E+05	4.77E+03	1.69E+05	**2.38E+03**
	min	4.23E+04	1.44E+05	3.09E+04	1.61E+05	2.57E+03	4.48E+04	**1.92E+03**
	variance	1.39E+10	4.05E+12	5.45E+10	7.32E+11	2.21E+06	1.95E+10	**1.04E+05**
F11	mean	2.62E+03	1.94E+03	**1.59E+03**	1.94E+03	2.16E+03	2.54E+03	1.64E+03
	min	1.41E+03	1.44E+03	1.41E+03	1.50E+03	1.40E+03	1.45E+03	**1.40E+03**
	variance	6.69E+05	5.22E+04	**2.75E+04**	9.46E+04	2.63E+05	3.68E+05	1.87E+05
F12	mean	1.33E+03	1.31E+03	1.31E+03	1.31E+03	1.31E+03	1.36E+03	1.31E+03
	min	1.31E+03	1.31E+03	1.31E+03	1.31E+03	1.31E+03	1.33E+03	1.31E+03
	variance	9.88E+02	3.94E+00	**7.31E-01**	5.31E-01	2.72E+00	3.21E+02	5.89E+00
F13	mean	2.55E+03	1.44E+03	**1.43E+03**	1.44E+03	1.48E+03	2.40E+03	1.44E+03
	min	1.67E+03	1.42E+03	**1.41E+03**	1.42E+03	1.44E+03	1.68E+03	1.42E+03
	variance	6.84E+05	3.99E+01	**3.71E+01**	8.47E+01	1.29E+03	1.55E+06	1.07E+02
F14	mean	3.91E+04	3.75E+04	3.42E+04	3.59E+04	3.43E+04	5.42E+04	**3.28E+04**
	min	1.83E+03	3.33E+04	3.25E+04	3.41E+04	1.50E+03	3.89E+04	**1.50E+03**
	variance	1.79E+08	6.53E+06	**3.73E+06**	3.08E+06	6.21E+07	1.04E+08	1.20E+08
F15	mean	1.61E+03	**1.60E+03**	**1.60E+03**	1.63E+03	**1.60E+03**	1.68E+03	**1.60E+03**
	min	1.60E+03	**1.60E+03**	**1.60E+03**	1.62E+03	**1.60E+03**	1.66E+03	**1.60E+03**
	variance	8.68E+01	3.57E+01	6.07E-07	1.91E+01	**2.02E-11**	1.49E+02	2.62E+01

TABLE V. COMPARISON OF OHFA WITH OTHERS ALGORITHM (D=100)

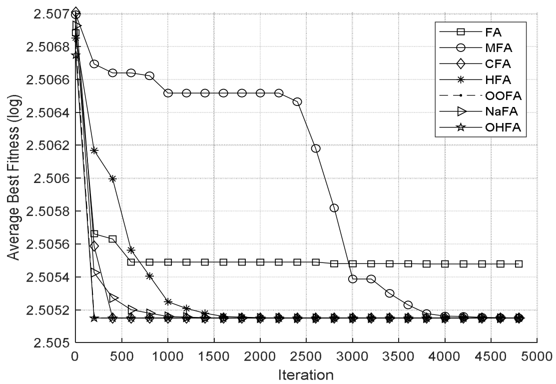
Function	*Indicators*	*FA*	*CFA*	*MFA*	*OOFA*	*HFA*	*NaFA*	*OHFA*
F1	mean	2.62E+10	5.22E+08	2.99E+07	8.92E+08	2.49E+06	1.84E+09	**9.09E+05**
	min	1.75E+10	2.11E+08	2.00E+07	7.37E+08	1.68E+06	9.85E+08	**3.32E+05**
	variance	2.63E+19	3.31E+16	4.35E+13	9.80E+15	3.30E+11	2.65E+17	**2.12E+11**
F2	mean	6.75E+11	2.80E+10	1.26E+04	7.72E+10	3.26E+03	1.95E+11	**2.00E+02**
	min	6.00E+11	1.68E+10	5.57E+03	6.52E+10	2.75E+02	1.68E+11	**2.00E+02**
	variance	2.18E+21	3.83E+19	4.06E+07	2.82E+19	1.67E+07	4.37E+20	**1.14E-02**
F3	mean	3.21E+02	**3.20E+02**	**3.20E+02**	3.21E+02	**3.20E+02**	**3.20E+02**	**3.20E+02**
	min	3.21E+02	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**	**3.20E+02**
	variance	4.24E-04	2.96E-03	2.24E-07	2.55E-01	1.78E-10	1.74E-09	**8.17E-14**
F4	mean	3.18E+03	1.35E+03	**7.55E+02**	1.66E+03	1.34E+03	1.37E+03	1.08E+03
	min	2.91E+03	1.14E+03	**6.43E+02**	1.53E+03	1.15E+03	1.24E+03	9.63E+02
	variance	2.03E+04	1.67E+04	**5.37E+03**	2.52E+03	6.23E+03	4.91E+03	9.43E+03
F5	mean	3.50E+04	1.66E+04	**1.13E+04**	1.92E+04	1.55E+04	1.57E+04	1.52E+04
	min	2.58E+04	1.41E+04	**8.69E+03**	1.80E+04	1.37E+04	1.31E+04	1.23E+04
	variance	5.43E+06	2.92E+06	2.35E+06	**3.01E+05**	1.15E+06	1.68E+06	1.83E+06
F6	mean	6.84E+09	2.90E+07	3.43E+06	4.22E+07	2.69E+05	2.47E+08	**2.47E+05**
	min	3.93E+09	8.19E+06	7.22E+05	2.47E+07	1.61E+05	4.58E+07	**1.09E+05**
	variance	5.12E+18	2.10E+14	2.71E+12	1.17E+14	5.59E+09	1.75E+16	1.12E+10
F7	mean	3.15E+04	1.08E+03	8.48E+02	1.33E+03	8.47E+02	2.85E+03	**8.26E+02**
	min	1.13E+04	9.46E+02	8.25E+02	1.24E+03	8.07E+02	2.06E+03	**7.36E+02**
	variance	7.99E+07	8.39E+03	**2.37E+02**	2.20E+03	1.06E+03	3.63E+05	1.96E+03
F8	mean	3.10E+09	1.87E+07	2.20E+06	2.32E+07	**7.56E+04**	2.56E+07	9.27E+04
	min	1.10E+09	2.90E+06	1.12E+06	1.25E+07	3.89E+04	1.03E+07	**2.65E+04**
	variance	9.04E+17	7.11E+13	7.80E+11	5.07E+13	**8.34E+08**	2.47E+14	2.80E+09
F9	mean	6.60E+03	1.44E+03	1.08E+03	1.29E+03	1.25E+03	4.08E+03	**1.02E+03**
	min	4.58E+03	1.08E+03	1.01E+03	1.25E+03	1.01E+03	3.58E+03	**1.01E+03**
	variance	1.62E+06	1.93E+05	2.58E+04	1.32E+03	3.37E+05	1.53E+05	**6.75E+00**
F10	mean	5.96E+09	1.97E+07	2.33E+06	2.47E+07	3.45E+04	2.38E+08	**6.76E+03**
	min	3.54E+09	2.44E+06	7.06E+05	1.02E+07	1.95E+04	5.29E+07	**4.95E+03**
	variance	3.40E+18	1.49E+14	1.22E+12	5.91E+13	6.25E+07	1.59E+16	**2.01E+06**
F11	mean	1.33E+04	4.50E+03	**2.26E+03**	5.38E+03	3.56E+03	6.11E+03	5.43E+03
	min	8.04E+03	4.11E+03	1.43E+03	4.86E+03	**1.41E+03**	3.44E+03	4.67E+03
	variance	6.78E+06	**4.40E+04**	1.96E+05	3.60E+05	4.09E+06	1.30E+06	2.03E+05
F12	mean	2.11E+03	1.39E+03	**1.32E+03**	1.44E+03	1.33E+03	1.77E+03	**1.32E+03**
	min	1.94E+03	1.36E+03	**1.32E+03**	1.39E+03	1.32E+03	1.66E+03	**1.32E+03**
	variance	1.27E+04	3.20E+02	**1.13E+00**	1.82E+03	2.06E+03	5.10E+03	**2.70E+00**
F13	mean	2.25E+04	1.80E+03	**1.75E+03**	1.81E+03	2.48E+03	1.04E+04	1.82E+03
	min	9.69E+03	1.78E+03	**1.74E+03**	1.78E+03	1.96E+03	4.78E+03	1.79E+03
	variance	7.07E+07	1.49E+02	**7.70E+01**	3.14E+03	6.99E+05	1.68E+07	1.00E+03
F14	mean	3.98E+06	2.44E+05	1.30E+05	3.85E+05	1.64E+05	7.43E+05	**1.22E+05**
	min	1.70E+06	2.12E+05	1.21E+05	2.54E+05	1.54E+05	4.35E+05	**1.11E+05**
	variance	2.37E+12	5.40E+08	1.26E+08	2.49E+10	**3.91E+07**	3.02E+10	2.82E+08
F15	mean	1.10E+07	5.14E+03	**1.60E+03**	3.07E+04	1.64E+03	4.83E+04	1.68E+03
	min	5.19E+06	2.67E+03	**1.60E+03**	1.66E+04	1.63E+03	2.52E+04	1.66E+03
	variance	1.78E+13	6.43E+06	1.20E-06	7.90E+07	1.87E+01	4.10E+08	1.09E+02

2) Convergence Analysis

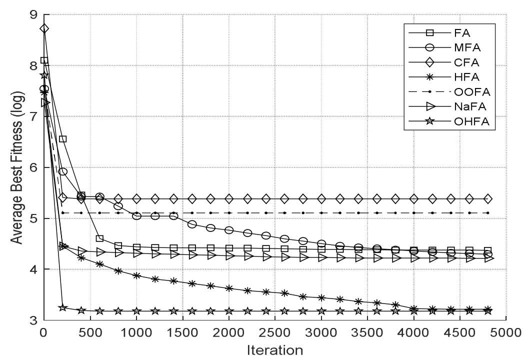
Figure 1 shows the convergence of the representative algorithms on each function. Figures 1 (a) ~ (d) show the convergence of each algorithm in 30 dimensions. Among them, For F1, the OHFA algorithm is far better than other algorithms in terms of convergence accuracy and convergence speed. For F3, f8 and F10, the OHFA's convergence accuracy is slightly superior to (at least equal to) other algorithms, while its convergence speed is far better than them.



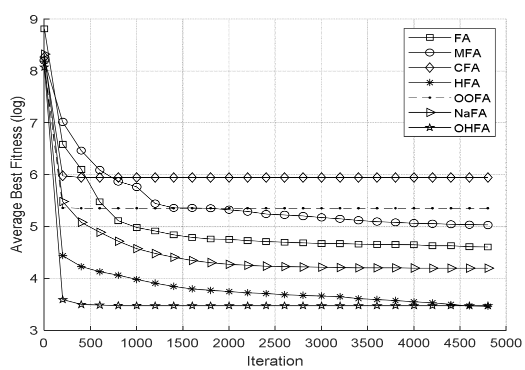
(a) Test Function F1



(b) Test Function F3



(c) Test Function F8



(d) Test Function F10

Fig. 1. *Convergence curves of test function F1, F2, F8, F10 (D=30)*

3) *Complexity Analysis*

Suppose the population size is *N,* the problem dimension size is *D*, and the maximum iteration number is *G.* According to the process of the standard firefly algorithm, the movement is the main factor in the time complexity which is $O(G*N*log(N)*D)$. The OHFA retains the basic framework of the Hybrid Firefly Algorithm (HFA) [17], and adds the OOBL strategy for globally optimal individuals. The main part of the OHFA is the movement and differential evolution in steps (6) to (17) and the OOBL on the globally optimal individual in step (21). During each iteration, the time complexity of the former is $O(N^2*D)$, while that of the latter is $O(D^2)$. So the total time complexity is $O(G*N^2*D)+O(G*D^2)$. With the low-order terms omitted, the total time complexity of the algorithm is $O(G*N^2*D)$. Therefore, the time complexity of the OHFA is slightly higher than that of the standard firefly algorithm. It is worth noting that the time complexity of the OHFA is the same as that of the HFA. Generally, its accuracy and stability are better than the standard firefly algorithm and HFA.

V. CONCLUSION

In the face of the low precision of the standard firefly algorithm for high-dimensional complex problems, we propose the OHFA in this paper. The OHFA is different from other firefly algorithms in that it maximizes the optimization ability of the brighter individuals. It uses differential evolution learning for brighter individuals and the OOBL strategies for the globally optimal individuals to increase the movement probability of brighter individuals, thereby improving the search ability of the population. Besides, the distance formula between fireflies is redefined to reflect the advantages of the movement. Through experiments on 15 test functions, the effectiveness, applicability, and accuracy of the OHFA are demonstrated in comparison with other representative firefly algorithms.

VI. ACKNOWLEDGEMENT

This research is supported by Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System (Grant No. znxx2018QN06), Major Project for New Generation of AI (Grant No. 2018AAA0100400) and the National Natural Science Foundation of Hunan (Grant No.2018JJ2098).

REFERENCES

[1] X.-S. Yang, Nature-Inspired Metaheuristic Algorithms, Luniver Press, 2010

[2] Yang X S. Firefly algorithms for multimodal optimization[C]// International symposium on stochastic algorithms. Springer, Berlin, Heidelberg, 2009: 169-178.

[3] Yang X S, He X. Firefly algorithm: recent advances and applications[J]. arXiv preprint arXiv:1308.3898, 2013.

[4] Florence AP, Shanthi V. A load balancing model using firefly algorithm in cloud computing. Journal of Computer Science. 2014 Jul 1;10(7):1156.

[5] Yu S, Zhu S, Ma Y, et al. A variable step size firefly algorithm for numerical optimization[J]. Applied Mathematics and Computation, 2015, 263: 214-220

[6] Yang, Xin-She. "Firefly algorithm, Levy flights, and global optimization." Research and development in intelligent systems XXVI. Springer, London, 2010. 209-218

[7] Wang H, Zhou X, Sun H, et al. Firefly algorithm with adaptive control parameters[J]. Soft Computing, 2017, 21(17): 5091-5102.

[8] Gandomi A H, Yang X S, Talatahari S, et al. Firefly algorithm with chaos[J]. Communications in Nonlinear Science and Numerical Simulation, 2013, 18(1): 89-98.

[9] Wang, Hui, and Wenjun Wang et al. "Firefly algorithm with random attraction." International Journal of Bio-Inspired Computation 8.1 (2016): 33-41

[10] Wang, Hui, Wenjun Wang, and Xinyu Zhou et al. "Firefly algorithm with neighborhood attraction." Information Sciences 382 (2017): 374-387

[11] Wang, Chunfeng and Xinyue Chu. "An Improved Firefly Algorithm with Specific Probability and Its Engineering Application." IEEE Access 7 (2019): 57424-57439

[12] Wu S, Wu Z, Peng H. Enhancing Firefly Algorithm with Best Neighbor Guided Search Strategy[J]. Wuhan University Journal of Natural Sciences, 2019, 24(6): 524-536.

[13] Ming Z, Shuqun Z, Zhaoyi L. Application of improved firefly algorithm in neural network[J]. Computer Engineering and Applications, 2017, 53(5): 159-163.

[14] Zhou, Lingyun, Lixin Ding, and Yunwen Lei. "An Enhanced Firefly Algorithm with Orthogonal Centroid Opposition-Based Learning." 2018 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2018.

[15] Zhang, Qiang, and P. C. Li. "Adaptive grouping difference firefly algorithm for continuous space optimization problems." Control and Decision 32.7 (2017): 1217-1222.

[16] Mohammadiyan N E, Ghadi A. Enhancing the accuracy of firefly algorithm by using the reproduction mechanism[C]. software engineering artificial intelligence networking and parallel distributed computing, 2017: 1-6.

[17] Wu, Jinran, and YouGan Wang et al. “An Improved Firefly Algorithm for Global Continuous Optimization Problems.” Expert Systems with Applications, vol. 149, 2020, p. 113340.

[18] Liu, Jingsen, Yinan Mao, and Xiaozhen Liu, et al. “A Dynamic Adaptive Firefly Algorithm with Globally Orientation.” Mathematics and Computers in Simulation, vol. 174, 2020, pp. 76–101.

[19] Wang, Wenjun, Hui Wang, and Hui Sun et al. "A hybrid firefly algorithm for continuous optimization problems." International Conference on Cloud Computing and Security. Springer, Cham, 2016

[20] Tizhoosh H R. Opposition-Based Learning: A New Scheme for Machine Intelligence[C]. computational intelligence for modelling, control and automation, 2005: 695-701.

[21] Rahnamayan S, Tizhoosh H R, Salama M M, et al. Opposition-Based Differential Evolution[J]. IEEE Transactions on Evolutionary Computation, 2008, 12(1): 64-79.

[22] Fister I, Yang X, Brest J, et al. Memetic firefly algorithm for combinatorial optimization[J]. arXiv: Optimization and Control, 2012.

[23] Liang J J, Qu B Y, Suganthan P N, et al. Problem definitions and evaluation criteria for the CEC 2015 competition on learning-based real-parameter single objective optimization[J]. Technical Report201411A, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2014, 29: 625-640.