



A novel enhanced exploration firefly algorithm for global continuous optimization problems

Jianxun Liu¹ · Jinfei Shi^{1,2} · Fei Hao² · Min Dai¹ · Xiaoya Zhang³

Received: 3 February 2021 / Accepted: 4 July 2021 / Published online: 6 August 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

In the global optimization process of the firefly algorithm (FA), there is a need to provide a fast convergence rate and to explore the search space more effectively. Therefore, we conduct modular analysis of the FA and propose a novel enhanced exploration firefly algorithm (EE-FA), which includes an enhanced attractiveness term module and an enhanced random term module. The attractiveness term module can improve the exploration efficiency and accelerate the convergence rate by enhancing the attraction between fireflies. The random term module improves the exploration efficiency by introducing a damped vibration distribution factor. The EE-FA uses multiple parameters to balance its exploration efficiency and convergence rate. The parameters have a great influence on the performance of the EE-FA. In order to achieve the best performance of the EE-FA, each parameter of the EE-FA needs to be simulated to determine its optimal value. Compared to multiple variants of the FA, the EE-FA has better exploration efficiency and a faster convergence speed. Experimental results reveal that the EE-FA recreated consistently vanquishes the front for 24 benchmark functions and 4 real design case studies in terms of both convergence rate and exploration efficiency.

Keywords Firefly algorithm · Global continuous optimization · Convergence speed · Damping vibration distribution

1 Introduction

Global continuous optimization problems have been extensively applied to scientific research, engineering technology and industrial design. Examples include image processing [1], path planning [2], complex structure design optimization [3, 4], electronic system design, gene recognition, etc. In most engineering applications, global optimization problems are often regarded as highly nonlinear with irregular multiple peaks, which are usually called NP-hard problems [5]. NP-hard problems always lead to huge computational complexity and poor computational stability. However, it is difficult to find a suitable and practical algorithm to solve any of the NP-hard problems. In fact, designing the best algorithm requires not only a wealth of experience but also a detailed understanding of the problem. Even so, there is no absolute guarantee that the best or even suboptimal solution will be found.

For continuous problems, in order to achieve efficient global optimization performance, many scholars have invested a great deal of effort in research, and many excellent algorithms have been proposed [6–8]. Two categories have been developed from these splendid algorithms:

✉ Jianxun Liu
1536543028@qq.com

✉ Jinfei Shi
shijf@njit.edu.cn

Fei Hao
feehao2012@163.com

Min Dai
ddt@seu.edu.cn

Xiaoya Zhang
1035699430@qq.com

¹ School of Mechanical Engineering, Southeast University, No. 2, Southeast University Road, Jiangning District, Nanjing 211189, Jiangsu, China

² School of Mechanical Engineering, Nanjing Institute of Technology, Nanjing 211167, China

³ College of Mechanical and Electrical Engineering, Guangdong Polytechnic of Industry and Commerce, Guangzhou 510510, China

deterministic algorithms and random algorithms. Deterministic algorithms mainly consist of the hill climbing method [9], Newton iteration method [10], simplex method [11], least squares method [12, 13], etc. The strength of deterministic algorithms is that they have high convergence efficiency for certain problems. It only takes a few iterations to find the optimal solution. Their disadvantage is that they easily fall into a local optimum. Due to the unique random mechanism, random algorithms can easily jump out of the local optimum and obtain the global optimum solution. Their weakness is that they can produce different solution sets under the same initial conditions, so their repeatability is very poor [14].

Most random algorithms can be considered metaheuristics. Many new metaheuristic algorithms are inspired by physical processes or biological intelligence in nature, such as particle swarm optimization (PSO) [15], the firefly algorithm (FA) [16], the flower pollination algorithm (FPA) [17], the grasshopper optimisation algorithm (GOA) [18], and various particle swarm algorithms (IPSO [19], YPSO [20], BPSO [21], etc.). The advantage is that the algorithms themselves have a good information sharing mechanism. The mechanism could make the algorithms converge quickly under certain conditions. They have been used to solve difficult optimization problems encountered in engineering [22]. Gandomi [23] proved that a metaheuristic algorithm has good effects in the global optimization of engineering problems.

The firefly algorithm is a new metaheuristic algorithm that was proposed by Yang [16]. It has the advantages of having simple concepts, being easy to understand, using a clear process, using few parameters, and having a strong random search ability. Lukasik and Žak believed that the FA performs better than PSO on continuous optimization problems [24]. The FA has been used to overcome economic dispatch problems in power system management [25]. Kavousi-Fard used the FA to solve the optimal parameters of the shortest load based on the support vector regression [7]. Similarly, the FA is used to solve NP scheduling problems and travelling salesman problems [26]. Sánchez applied the FA to the field of intelligent recognition [27]. Yang and He discussed the performance of intelligent segmentation and multiple mode processing [28]. Frumen [29] proposed a new ant colony optimization method with dynamic parameter adaptation. Daniela [30] combined PSO and neural networks and applied the resulting method to human recognition. Frumen [31] proposed an improvement to the convergence and diversity of the swarm in PSO using interval type-2 fuzzy logic. Frumen [32] used type-2 fuzzy logic to adjust the dynamic parameters of the GSA. Daniela [33] used the FA to optimize the neural network to realize face recognition.

For global continuous optimization problems, how to make the FA quickly and effectively obtain the global optimal value is key. This requires the FA to have good

exploration efficiency and a fast convergence speed. Therefore, the convergence speed and exploration efficiency of the FA have become the research focus of researchers. Many researchers have incorporated some parameter adjustment mechanisms, such as a fuzzy controller, which could be able to update the system parameters in the FA according to the exploration process, into the algorithm [34, 35]. Yang introduced the Lévy flight strategy into the FA, called the Lévy-flight FA (LF-FA) [36]. In 2021, Shuhao adopted a personalized step strategy to improve the FA and proposed the PSSFA [37]. Ao proposed a sigmoid attractive firefly algorithm (IFA) and used it in IIR filter design [38]. Navid combined the FA with IFF and proposed a hybrid FA-IFF [39]. The FA-IFF and extreme learning machines were used to predict the thermal conductivity of soil.

In 2020, Jinran proposed an adaptive firefly algorithm for switching between the exploration mode and development mode (AD-IFA) [40]. The adaptive switch mechanism introduced by the AD-IFA improves the sufficiency of local development. However, the mechanism leads to a substantial increase in the computational complexity and computational costs of the AD-IFA.

The article has the following three innovative aspects:

1. An improved attractiveness term is proposed to enhance the attraction between fireflies.
2. Based on the characteristics of the damped vibration waveform, a new random item is constructed to enhance the exploitation sufficiency of the FA.
3. Based on feedback theory, the firefly absorption coefficient increases as the number of iterations of FA increases.

The remainder of this paper is organized as follows. First, related work on the FA, the Lévy-flight FA and the AD-IFA are reviewed in Sect. 2. Section 3 outlines the method to improve the attractiveness term and random term and proposes the EE-FA. The optimal parameter setting of the EE-FA, the performance comparison of the algorithms, the benchmark functions and the simulation results are given in Sect. 4. Section 5 shows the efficiency of the EE-FA through four engineering applications. Finally, Sect. 6 concludes the paper.

2 Related work

2.1 The standard firefly algorithm

In nature, fireflies use their own light as a signal to attract other individuals of the same kind. FA is a new metaheuristic algorithm proposed by simulating the behaviour of fireflies. In the FA, each individual in the group corresponds to

a candidate solution of the problem. The global optimization process is actually the process of moving the position of each individual firefly. The FA has the following three basic principles.

1. All firefly individuals have no gender distinction.
2. The attractiveness of each individual is related to its own brightness. Individuals with weaker brightness will be attracted by individuals with stronger brightness and move towards them, and individuals with the strongest brightness could move randomly in the solution space.
3. The objective function value of the feasible solution of the problem to be solved is usually taken as the individual brightness value.

When constructing the FA, two key issues need to be addressed. The two key issues are the relative brightness and the attractiveness between fireflies.

(1) The relative brightness between fireflies can be expressed as follows:

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

In the formula, r represents the distance between the fireflies, γ represents the light intensity attraction coefficient, and I_0 represents the initial fluorescence brightness of the firefly.

(2) The attraction between fireflies can be expressed as follows:

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

where β_0 is the self-attraction of the firefly when $r=0$.

(3) The distance between the fireflies can be expressed as follows:

$$r_{ij} = ||x_i - x_j|| = \sqrt{\sum_{k=1}^D (x_{ik} - x_{jk})^2} \tag{3}$$

In the formula, assuming that fireflies i and j are located at x_i and x_j , respectively, the Cartesian distance between them should be represented by r_{ij} , where D is the space dimension.

The position update formula of the firefly algorithm is as follows:

$$x_{i,t+1} = x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) + \alpha \cdot (\text{rand} - 0.5). \tag{4}$$

2.2 The Lévy-flight firefly algorithm

Like other metaheuristic algorithms, the FA cannot avoid falling into a local optimum. Therefore, Lévy flight was introduced into the FA to improve its ability to jump out of local optima (LF-FA) [36]. Compared with the FA, the LF-FA has better global exploration efficiency and a faster convergence speed. Its main idea is to use a Lévy random distribution instead of a traditional random distribution.

$$x_{i,t+1} = x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) + \alpha \cdot \text{sign}(\text{rand} - 0.5) \otimes \text{Levy}. \tag{5}$$

where α indicates the coefficient of the random term, and \otimes is the Hadamard product. $\text{sign}()$ is a sign function. Levy is a Lévy distribution.

The formula of the Lévy random distribution is as follows:

$$\text{Levy}(\eta) \sim \frac{\phi \times \mu}{|v|^{1/\eta}} \tag{6}$$

where v and μ obey a standard normal distribution. ϕ is calculated as follows:

$$\phi = \left[\frac{\tau(1 + \eta) \times \sin(\pi \times \eta/2)}{\tau((1 + \eta)/2) \times \eta \times 2^{(\eta-1)/2}} \right]^{1/\eta}, \tag{7}$$

where τ is standard Gamma function, and $\eta = 1.5$.

2.3 The adaptive logarithmic spiral-Lévy firefly algorithm

Although the LF-FA has better global exploration efficiency, it breaks the balance between exploration and exploitation. To compensate for this balance, Jinran [40] proposed a new algorithm for exploration and exploitation mode adaptive switching (AD-IFA). Its update position formula is as follows:

$$x_{i,t+1} = \begin{cases} x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) + \alpha \cdot \text{sign}|\text{rand} - 0.5| \otimes \text{Levy}, & u > R_t \\ x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) \otimes e^{b-1} \otimes \cos(2\pi \cdot 1), & u \leq R_t \end{cases} \tag{8}$$

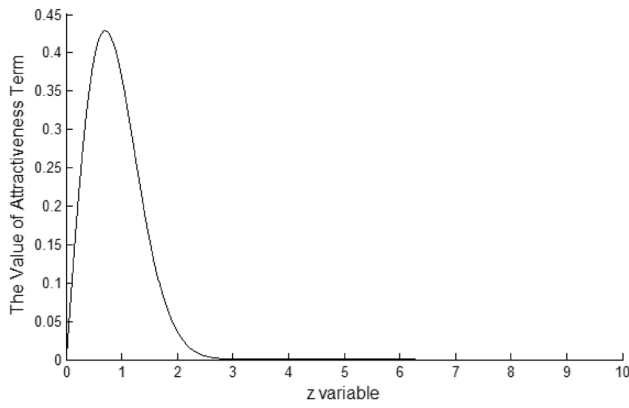


Fig. 1 The characteristic curve of the attractiveness term module

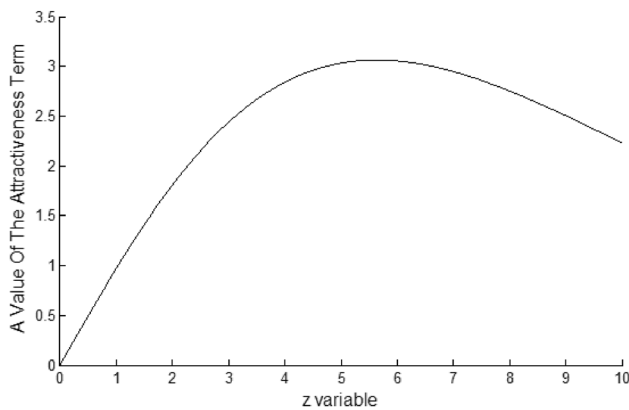
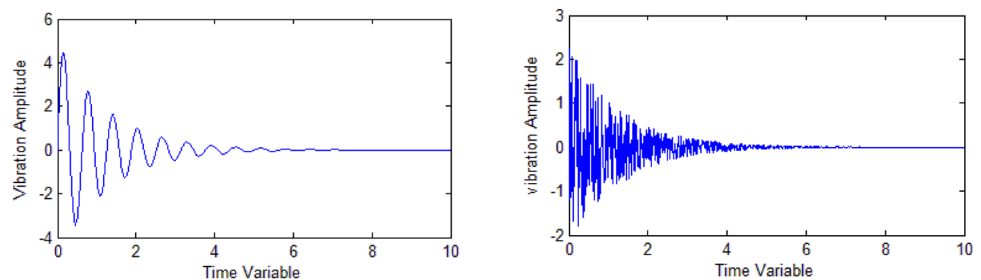


Fig. 2 The characteristic curve of the enhanced attractiveness item module

where u is a uniform random number in $[0, 1]$, and R_t is calculated as follows:

$$R_{t+1} = \begin{cases} \frac{1}{1+\exp\left(-\frac{f_t^*}{f_{t-1}^*}\right)}, & \lceil \lg |f_t^*| \rceil \neq \lceil \lg |f_{t-1}^*| \rceil, \\ \frac{1}{1+\exp\left(-\frac{f_t^* - \theta \cdot \lceil \frac{f_t^*}{\theta} \rceil}{f_{t-1}^* - \theta \cdot \lceil \frac{f_{t-1}^*}{\theta} \rceil}\right)}, & \text{otherwise} \end{cases} \quad (9)$$

Fig. 3 Characteristic curve diagram of formula (19) (left subfigure) and characteristic curve diagram of formula (20) (right subfigure)



where f_t^* is the best fitness function value of the t th iteration, $\lg(\cdot) = \log_{10}(\cdot)$, and $\lceil \cdot \rceil$ is the lower bound function. The value expression of θ is as follows:

$$\theta = 10^{\lceil \lg |f_t^* - f_{t-1}^*| \rceil + 1}. \quad (10)$$

The AD-IFA introduces an adaptive switching mechanism, which truly strengthens the balance between the exploration and exploitation of the algorithm in the global optimization process. However, the AD-IFA has three disadvantages as follows:

1. The adaptive switch mechanism greatly increases the complexity of the AD-IFA.
2. The values of the random parameter α and the light absorption coefficient γ are random, subjective and unscientific. The two parameters can directly affect the global exploration efficiency of the AD-IFA.
3. The complexity of the AD-IFA does not effectively speed up the convergence speed and reduces the number of iterations.

In order to improve the deficiencies of the AD-IFA, the article improves the FA from the following three aspects:

1. The complexity of AD-IFA is reduced, the adaptive switch mechanism is removed, and the related parameters of the distance factor between the fireflies are used instead.
2. Modify the random term of the FA. According to the damping vibration curve law, the damping vibration factor is introduced into the unified random distribution instead of the Lévy random distribution.
3. Each parameter in the newly constructed algorithm is simulated to determine its optimal parameter.
4. In order to accelerate the convergence rate, let the light attraction coefficient γ increase as the number of iterations increases.

3 A enhanced exploration firefly algorithm

In order to facilitate the study of the FA’s position update equation, the FA is divided into three modules: (1) the initial position module, (2) the attractiveness term module, and (3) random term module. Formula (4) can be simplified as follows:

$$x_{i,t+1} = x_{i,t} + A_{ij} + B_{ij}, \tag{11}$$

1. $x_{i,t}$ is the initial position module.
2. The attractiveness term module is as follows:

$$A_{ij} = \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}). \tag{12}$$

$A_{i,j}$ represents the attractiveness term module.

3. The random term module is as follows

$$B_{ij} = \alpha \cdot (\text{rand} - 0.5). \tag{13}$$

B_{ij} is random term module. rand denotes a d-dimensional uniform random vector in $[0, 1]^D$. γ is the light absorption coefficient. α is a parameter in $[0, 1]$. β_0 is the original light intensity. $x_{j,t} - x_{i,t}$ is a multidimensional variable, and the dimension is D .

3.1 Attractiveness term module

Under the condition of one-dimensional space, we analyze the attractiveness term module. Let $\gamma = 1, \beta_0 = 1$, and $x_{j,t} - x_{i,t} = z$, where z represents the distance between two fireflies. The one-dimensional movement area of the firefly is $[0, 10]$. The attractiveness term can be simplified as follows:

$$A(z) = \beta_0 \cdot e^{-\gamma \cdot z^2} \cdot z. \tag{14}$$

The characteristic curve of the attractiveness term module is as follows.

The characteristic curve graph of the attractiveness term module shows that the value of the attractiveness term begins to increase rapidly as the distance between the fireflies increases. Then, as the distance continues to increase, the value of the attractiveness term begins to decrease rapidly. When the distance is greater than 3, the value of the attractiveness term becomes diminutive, almost approaching zero. In other words, it is almost unattractive to distant fireflies. The location update formula only works with the random item module. Long-distance fireflies move randomly, directionlessly and aimlessly without restriction. This will greatly increase the number of invalid iterations of the FA.

In order to overcome this drawback, to make the attractiveness term module more attractive to prolonged distant fireflies, we introduce an attractiveness enhancement factor into the attractiveness term module.

The attractiveness enhancement factor is as follows:

$$K = \frac{1/h}{1 + (b/h) \cdot r_{ij}}. \tag{15}$$

The strengthened attractiveness item module is:

$$A(z) = \beta \cdot e^{-\gamma \cdot \left(\frac{1/h}{1+(b/h) \cdot z}\right) \cdot z^2} \cdot z. \tag{16}$$

where γ, β, b, h are the undetermined constants; and their default initial values are 0.03, 1, 10, and 100.

The characteristic curve of the enhanced attractiveness item module is as follows.

Figure 2 shows that the value of the enhanced attractiveness item module increases to a certain value as the distance between the fireflies increases. Then, the value of the attractiveness item slowly decreases as the distance increases. Compared with Figs. 1, 2 has better attraction to long-distance fireflies. The novel attractiveness item module is able to effectively attract fireflies and reduce the number of moves.

Fig. 4 Characteristic curve diagram of formula (21) (left subfigure) and characteristic curve diagram of formula (17) (right subfigure)

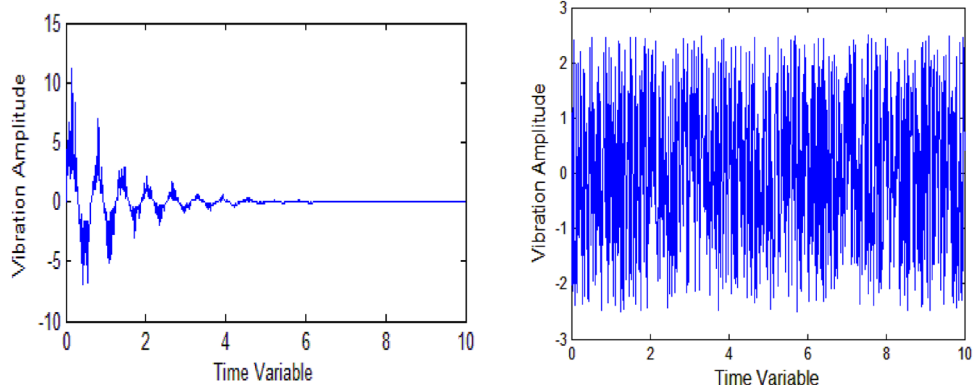


Fig. 5 Convergence trend of different values of β

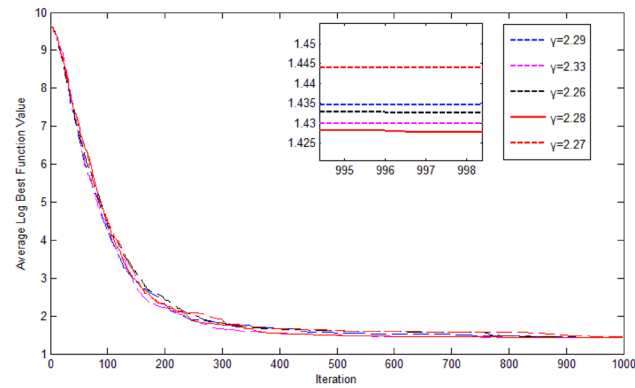
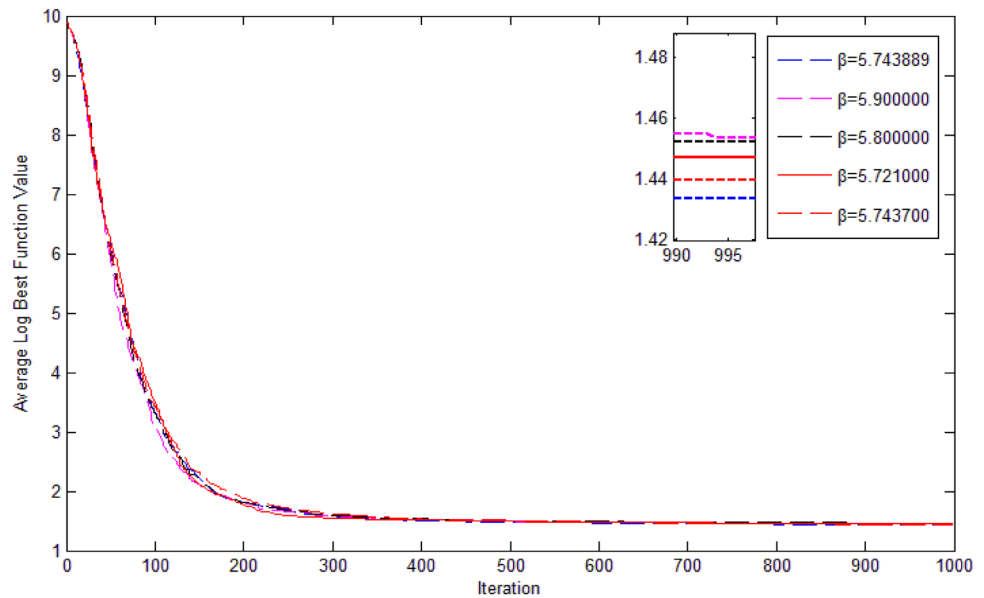


Fig. 6 Convergence trend with different values of γ

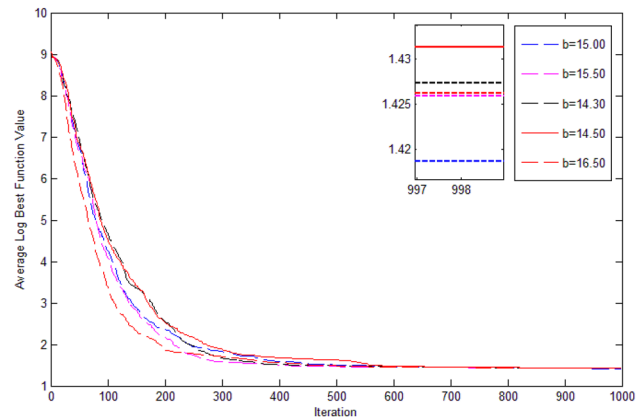


Fig. 7 Convergence trend of different values of b

3.2 Random item module

$$B_{i,j} = \alpha \cdot (\text{rand} - 0.5) \tag{17}$$

The random item module is a unique part of the metaheuristic algorithm. It can prevent the FA from falling into the local optimum and improve the exploration efficiency of the FA. The purpose of our reconstruction of the random item module is to realize the connection between the random item module and the firefly. When the distance between fireflies is far, the module can improve the global search efficiency; and when the distance between fireflies decreases, the module can strengthen the local exploration efficiency.

Obviously, Eq. (17) is a random term that is unrelated to the distance between the fireflies. If α takes a fixed value, it is a random number that obeys a [0, 1] uniform distribution. Regardless of whether the distance between the fireflies is

far or near, the exploration efficiency of the random term is almost the same. This severely weakens the exploration efficiency of the FA. According to the characteristics of the damping vibration curve, the amplitude decreases rapidly as the number of vibrations increases. The frequency of the damped vibration function can be changed. The purpose of the damping vibration characteristic is to play a major role when the distance between the fireflies decreases. When the distance between fireflies increases, the characteristic does not play a major role.

The damping vibration function formula is as follows [41]:

$$y = A \cdot e^{-a_1 \cdot t} \cdot \cos(a_2 \cdot t + a_3). \tag{18}$$

In order to satisfy that condition when $t=0, y=0$, Eq. (19) below is used to replace it.

$$y = A \cdot e^{-a_1 \cdot t} \cdot \sin(a_2 \cdot t). \tag{19}$$

The damping vibration distribution equation is as follows:

$$y = A \cdot e^{-t} \cdot \sin(2 \cdot \pi \cdot (\text{rand} - 0.5)). \tag{20}$$

Let $A=5$, $a_1=1$ and $a_2=2\pi$. The curves corresponding to formula (19) and formula (20) are as follows.

The right subfigure in Fig. 3 shows that the characteristic curve not only has the characteristic curve of the damping vibration but also has the characteristic curve of a random distribution.

$$y = A \cdot e^{-t} \cdot e^{2 \cdot (\text{rand}-0.5)} \cdot \sin(2 \cdot \pi \cdot (\text{rand} - 0.5)). \tag{21}$$

Let $A=5$ and $\alpha=5$. The best values of A and α are obtained in Sect. 4, which are set arbitrarily to illustrate the principle.

The left subfigure in Fig. 4 shows the following two characteristic points:

1. Compared with Fig. 3, the amplitude is significantly enhanced.
2. As the distance between the fireflies increases, the random attribute grows increasingly smaller.

Therefore, when the distance between fireflies is small, it can play a major role; and when the distance between fireflies is large, formula (17) plays a major role. According to formula (17) and formula 21, the new random term module is reconstructed to realize the enhancement of the exploration efficiency of FA.

The new random term model is as follows:

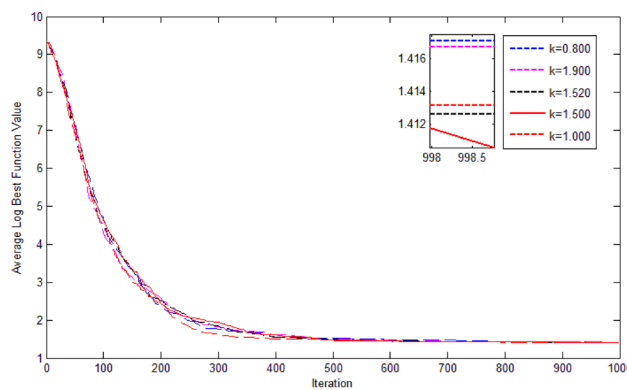


Fig. 9 Convergence trend of different values of k

$$B_{ij} = g \cdot (k \cdot A_{ij} \cdot (\text{rand} - 0.5) + f \cdot e^{-|x_{ij} - x_{i,j}|} \cdot e^{2(\text{rand}-0.5)} \cdot \sin(2 \cdot \pi \cdot (\text{rand} - 0.5))). \tag{22}$$

where k, f , and g are undetermined parameters. The role of k, f , and g is to balance exploration efficiency and convergence behaviour.

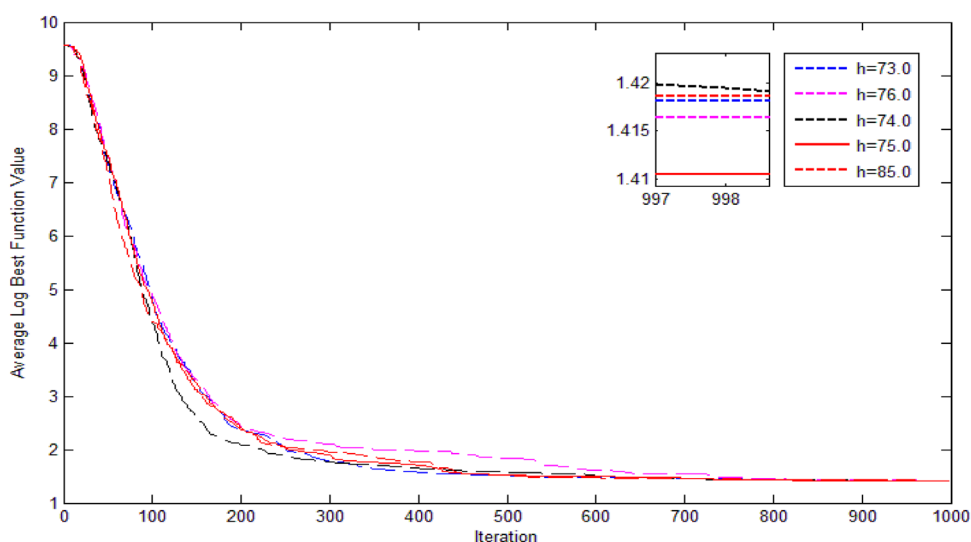
This modifies the light absorption coefficient γ as follows:

$$\gamma = s + \text{iter}/\text{iter max}, \tag{23}$$

where iter is the current number of iterations, itermax is the maximum number of iterations, and s is an undetermined parameter.

The location update formulas of the EE-FA are as follows:

Fig. 8 Convergence trend of different values of h



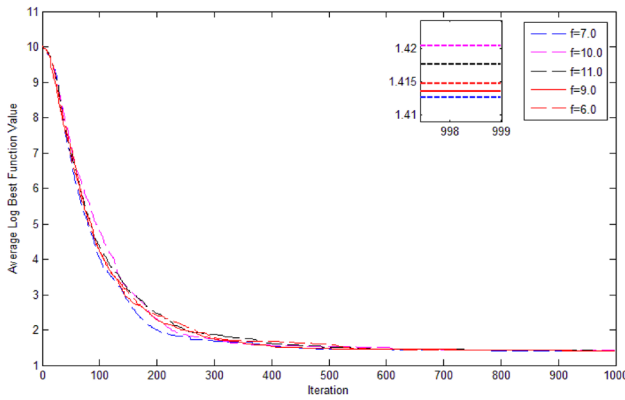


Fig. 10 Convergence trend of different values of f

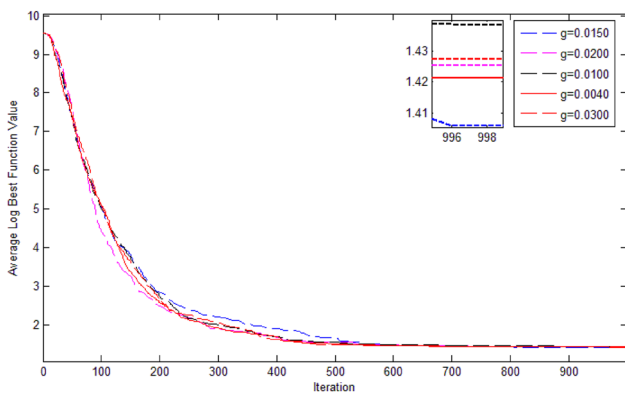


Fig. 11 Convergence trend of different values of g

$$\begin{cases}
 x_{i,t+1} = x_{i,t} + A_{i,j} + B_{i,j} \\
 A_{i,j} = \beta \cdot e^{-\gamma \cdot \left(\frac{1/h}{1+(b/h) \cdot r_{i,j}}\right) \cdot r_{i,j}^2} \cdot (x_{j,t} - x_{i,t}) \\
 \gamma = s + \text{iter}/\text{itermax} \\
 B_{i,j} = g \cdot (k \cdot A_{i,j} \cdot (\text{rand} - 0.5) + f \cdot e^{-|x_{j,t} - x_{i,t}|} \cdot e^{2(\text{rand}-0.5)} \cdot \sin(2 \cdot \pi \cdot (\text{rand} - 0.5)))
 \end{cases} \quad (24)$$

4 Numerical simulations

The EE-FA adopts multiple constants to balance the exploration efficiency and convergence speed and determines the best value of these constants through simulation. Then, the exploration efficiency and convergence speed of the FA, LF-FA, PSSFA, IFA, FA-IFF, AD-IFA and EE-FA were compared. Finally, the benchmark functions are globally

optimized, and the minimum mean (mean) and standard deviation (std) are recorded.

4.1 Determination of the best parameters of the EE-FA

At this time, γ is a fixed parameter. The EE-FA contains seven undetermined parameters: $\beta, \gamma, b, h, k, f, g$. The best values of the seven independent parameters are determined through the different influences of different parameters on the performance of the EE-FA. The sphere function $f(x) = \sum_{i=1}^d x_i^2$ is selected as the benchmark function. The dimension space d is set to 15. The value of x_i ranges from 0 to 100.

To eliminate the influence of random errors, each simulation is executed 50 times, and the average value is calculated.

Using the control variable method, the diverse values of each undetermined parameter are simulated individually. After many simulations, the parameter values are obtained.

4.1.1 Impact of β on the exploration efficiency of the EE-FA

The other six parameter values are initialized as follows: $\gamma = 2, b = 10, h = 100, k = 0.8, f = 6$, and $g = 0.04$. The effects of different values of β on the performance of the algorithm are shown in the following Fig. 5.

Figure 5 shows that when $\beta = 5.8$ and 5.9 , the exploration efficiency of the algorithm is the lowest. When $\beta = 5.721$, the exploration efficiency is improved. When $\beta = 5.7437$, the exploration efficiency is higher than when $\beta = 5.721$. When $\beta = 5.743889$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter β is 5.743889.

4.1.2 Impact of γ on the exploration efficiency of the EE-FA

The optimal parameter value obtained was the following: $\beta = 5.743889$. The other five parameter values are then initialized as follows: $b = 10, h = 100, k = 0.8, f = 6$, and $g = 0.04$. The effects of different values of γ on the performance of the algorithm are shown in the following Fig. 6.

Figure 6 shows that when $\gamma = 2.27$, the exploration efficiency of the algorithm is the lowest. When $\gamma = 2.29$ and 2.26 , the exploration efficiency is improved. When $\gamma = 2.33$, the exploration efficiency is higher than when $\gamma = 2.26$. When $\gamma = 2.28$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter γ is 2.28.

4.1.3 Impact of b on the exploration efficiency of the EE-FA

The optimal parameter values obtained were $\beta = 5.743889$ and $\gamma = 2.28$. The other four parameter values are initialized

as follows: $h = 100, k = 0.8, f = 6,$ and $g = 0.04$. The effect of different values of b on the performance of the algorithm is shown in the following Fig. 7.

Figure 5 shows that when $b = 14.5$, the exploration efficiency of the algorithm is the lowest. When $b = 14.3$, the exploration efficiency is improved. When $b = 15.5$ and 16.5 , the exploration efficiency is higher than when $b = 14.3$. When $b = 15.0$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter b is 15.0 .

4.1.4 Impact of h on the exploration efficiency of the EE-FA

The optimal parameter values obtained were as follows: $\beta = 5.743889, \gamma = 2.28,$ and $b = 15$. The other three parameter values are then initialized as follows: $k = 0.8, f = 6,$ and $g = 0.04$. The effects of different values of h on the performance of the algorithm shown in the Fig. 8.

Figure 8 shows that when $h = 74.0$, the exploration efficiency of the algorithm is the lowest. When $h = 73.0$ and 85.0 , the exploration efficiency is improved. When $h = 76.0$, the exploration efficiency is higher than when $h = 73.0$. When $h = 75.0$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter h is 75.0 .

4.1.5 Impact of k on the exploration efficiency of the EE-FA

The optimal parameter values obtained were as follows: $\beta = 5.743889, \gamma = 2.28, b = 15,$ and $h = 75.0$. The other two parameter values are then initialized as $f = 6$ and $g = 0.04$,

The effects of different values of k on the performance of the algorithm are shown in the Fig. 9.

Figure 9 shows that when $k = 0.8$ and 1.9 , the exploration efficiency of the algorithm is the lowest. When $k = 1.0$, the exploration efficiency is improved. When $k = 1.52$, the exploration efficiency is higher than when $k = 1.0$. When $k = 1.5$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter k is 1.500 .

4.1.6 Impact of f on the exploration efficiency of the EE-FA

The optimal parameter values obtained were as follows: $\beta = 5.743889, \gamma = 2.28, b = 15, h = 75.0,$ and $k = 1.5$. The last parameter value is initialized as follows: $g = 0.04$. The effects of different values of k on the performance of the algorithm are shown in the Fig. 10

Figure 10 shows that when $f = 10.0$, the exploration efficiency of the algorithm is the lowest. When $f = 11.0$, the exploration efficiency is improved. When $f = 9.0$ and 6.0 , the exploration efficiency is higher than when $f = 11.0$. When $f = 7.0$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter f is 7.0 .

4.1.7 Impact of g impact on the exploration efficiency of the EE-FA

The optimal parameter values obtained were as follows: $\beta = 5.743889, \gamma = 2.28, b = 15, h = 75.0, k = 1.5,$ and $f = 7.0$. The effects of different values of g on the performance of the algorithm are shown in the Fig. 11.

Figure 11 shows that when $g = 0.01$, the exploration efficiency of the algorithm is the lowest. When $g = 0.02$ and

Table 1 Types of algorithms

FA	$x_{i,t+1} = x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) + \alpha \cdot (rand - 0.5)$
LF-FA	$x_{i,t+1} = x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) + \alpha \cdot \text{sign}(rand - 0.5) \otimes Levy$
AD-IFA	$x_{i,t+1} = \begin{cases} x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) + \alpha \cdot \text{sign} rand - 0.5 \otimes Levy, & u > R_t \\ x_{i,t} + \beta_0 \cdot e^{-\gamma \cdot r_{ij}^2} \cdot (x_{j,t} - x_{i,t}) \otimes e^{b-1} \otimes \cos(2\pi \cdot 1), & u \leq R_t \end{cases}$
EE-FA	$\begin{cases} x_{i,t+1} = x_{i,t} + A_{ij} + B_{ij} \\ A_{ij} = 5.743889 \cdot e^{-(1.28 + \text{iter} \times (1/\text{itermax})) \cdot (\frac{1/75}{1+(15/75)r_{ij}})^{r_{ij}^2}} \cdot (x_{j,t} - x_{i,t}) \\ B_{ij} = 0.015 \cdot (1.5 \cdot A_{ij} \cdot (rand - 0.5) + 7 \cdot e^{- x_{j,t} - x_{i,t} } \cdot e^{2(rand-0.5)} \cdot \sin(2 \cdot \pi \cdot (rand - 0.5))) \end{cases}$

Table 2 Time complexity experiment setup

Algorithms	Population	Maximum iterations	Dimensional	Others
AD-IFA	15	100	4	Intel(R) Core(TM) i3 CPU, M 380 @ 2.53 GHz
EE-FA	15	100	4	

Table 3 Time complexity calculation of experimental results

Function	Total running time (unit: seconds)	
	AD-IFA	EE-FA
F1	12.52799947	7.01598407
F2	12.21551831	6.96824297
F3	8.279623167	6.97935718
F4	12.39544264	7.14623037
F5	13.85101093	8.71392987
F6	12.85877523	7.99802498
F7	12.09153378	7.97123994
F8	12.56812335	8.07496944
F9	11.41057697	7.88444513

0.03, the exploration efficiency is improved. When $g = 0.004$, the exploration efficiency is higher than when $g = 0.02$. When $g = 0.015$, the exploration efficiency is the best. Therefore, the optimal value of the undetermined parameter g is 0.015.

The optimal parameter values obtained were as follows: $\beta = 5.743889$, $\gamma = 2.28$, $b = 15$, $h = 75.0$, $k = 1.5$, $f = 7.0$, and $g = 0.015$.

The EE-FA proposed in this article is as follows:

Table 4 Description of the 24 benchmark functions

Function ID	Equation description
F1	$f_1(x) = \sum_{i=1}^D x_i^2, x_i \in [-100, 100]$
F2	$f_2(x) = \sum_{i=1}^D \left(\sum_{j=1}^i x_j \right)^2, x_i \in [-100, 100]$
F3	$f_3(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i , x_i \in [-10, 10]$
F4	$f_4(x) = \sum_{i=1}^D ix_i^4, x_i \in [-1.28, 1.28]$
F5	$f_5(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2} \right) - \exp \left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i) \right) + 20 + \exp(1), x_i \in [-32, 32]$
F6	$f_6(x) = \sum_{i=1}^D \sum_{j=1}^i x_j^2, x_i \in [-65.536, 65.536]$
F7	$f_7(x) = \sum_{i=1}^D x_i ^{i+1}, x_i \in [-1, 1]$
F8	$f_8(x) = \sum_{i=1}^D x_i^2 + \left(\sum_{i=1}^D 0.5x_i \right)^2 + \left(\sum_{i=1}^D 0.5ix_i \right)^4, x_i \in [-5, 10]$
F9	$f_9(x) = 10^6 \cdot x_1^2 + \sum_{i=2}^D x_i^2, x_i \in [-1, 1]$
F10	Composition function 5 ($D = 16$), $[-32, 32]$
F11	Composition function 6 ($D = 16$), $[-65, 65]$
F12	Composition function 7 ($D = 16$), $[-1, 1]$
F13	Composition function 8 ($D = 16$), $[-5, 10]$
F14	Composition function 9 ($D = 16$), $[-1, 1]$
F15	Composition function 5 ($D = 16$), $[-24, 24]$
F16	Composition function 6 ($D = 16$), $[24, 24]$
F17	Composition function 7 ($D = 32$), $[-0.5, 0.5]$
F18	Composition function 8 ($D = 32$), $[-5, 10]$
F19	Composition function 9 ($D = 32$), $[-1, 1]$
F20	Composition function 5 ($D = 64$), $[-14, 14]$
F21	Composition function 6 ($D = 64$), $[-14, 14]$
F22	Composition function 7 ($D = 64$), $[-1, 1]$
F23	Composition function 8 ($D = 64$), $[-5, 10]$
F24	Composition function 9 ($D = 64$), $[-1, 1]$

Table 5 Results and comparison

Algorithms	F1		F2		F3	
	Mean	Std	Mean	Std	Mean	Std
FPA	1.0E+01	8.0E+00	3.3E+01	2.9E+01	2.0E+00	1.2E+00
GOA	2.9E+02	2.2E+02	1.6E+02	1.5E+02	2.5E+01	2.1E+01
PSO	8.0E−03	5.7E−03	6.3E−02	7.5E−02	2.1E+00	4.0E+00
IPSO	2.3E−03	1.3E−03	4.2E−03	3.0E−03	8.3E−02	2.8E−02
YSPSO	1.6E+02	1.0E+02	2.3E+02	1.6E+02	6.1E+00	3.4E+00
BPSO	4.6E+02	2.5E+02	5.3E+02	4.0E+02	1.4E+01	6.4E+00
EE-FA	0.0E+00	0.0E+00	0.0E+02	0.0E+00	00E+00	00E+00
Algorithms	F4		F5		F6	
	Mean	Std	Mean	Std	Mean	Std
FPA	1.3E+02	1.9E+02	8.9E+00	2.3E+00	1.0E+02	6.1E+01
GOA	1.5E+05	3.1E+05	1.9E+01	3.0E−01	1.4E+03	1.3E+03
PSO	8.5E−04	2.8E−03	1.9E+01	1.5E+00	4.8E−02	4.1E−02
IPSO	9.6E−06	9.7E−06	1.2E+01	6.7E+00	1.1E−02	7.2E−03
YSPSO	6.8E+04	1.5E+05	1.6E+01	2.7E+00	8.9E+02	7.4E+02
BPSO	1.5E+05	1.6E+05	1.9E+01	1.5E+00	3.1E+03	1.5E+03
EE-FA	0.0E+00	0.0E+00	3.9E+00	1.0E+00	00E+00	00E+00
Algorithms	F7		F8		F9	
	Mean	Std	Mean	Std	Mean	Std
FPA	8.8E+00	4.2E+00	6.5E+03	7.2E+03	6.0E+00	1.2E+01
GOA	4.8E+03	1.0E+04	1.2E+03	4.0E+02	1.5E+00	6.8E−01
PSO	1.9E−01	6.9E−01	3.0E−02	3.8E−02	3.6E+00	2.6E+00
IPSO	6.8E−03	3.9E−03	3.9E−03	3.3E−03	2.0E+00	1.4E+00
YSPSO	1.6E+04	3.6E+04	6.7E+06	6.9E+06	7.2E+00	2.7E+01
BPSO	9.0E+03	1.0E+04	7.8E+06	9.1E+06	2.3E+04	3.4E+04
EE-FA	0.0E+00	0.0E+00	0.0E+00	0.0E+00	1.5E−01	3.6E−01

$$\begin{cases}
 x_{i,t+1} = x_{i,t} + A_{i,j} + B_{i,j} \\
 A_{i,j} = 5.743889 \cdot e^{-2.28 \cdot \left(\frac{1/75}{1+(15/75) \cdot r_{i,j}}\right) \cdot r_{i,j}^2} \cdot (x_{j,t} - x_{i,t}) \\
 B_{i,j} = 0.015 \cdot (1.5 \cdot A_{i,j} \cdot (\text{rand} - 0.5) + 7 \cdot e^{-|x_{j,t} - x_{i,t}|} \cdot e^{2(\text{rand}-0.5)} \cdot \sin(2 \cdot \pi \cdot (\text{rand} - 0.5)))
 \end{cases} \quad (25)$$

4.2 Improve the iteration speed of the EE-FA

Our research found that the light attraction coefficient of fireflies also has a significant effect on the convergence speed of

the FA. Within a certain range, the greater the optical attraction coefficient of the FA is, the faster its iteration speed, but its exploration efficiency is correspondingly weakened.

In order to improve the convergence speed without compromising the exploratory ability, a functional equation to increase the optical attraction coefficient with the number of iterations of the algorithm is proposed (27).

The iterative formula of the light absorption coefficient is as follows:

$$\gamma = 1.28 + \text{iter}/\text{itermax}, \quad (26)$$

where itermax is the maximum number of iterations.

Therefore, the position update formulas of the EE-FA with the best parameter value are as follows:

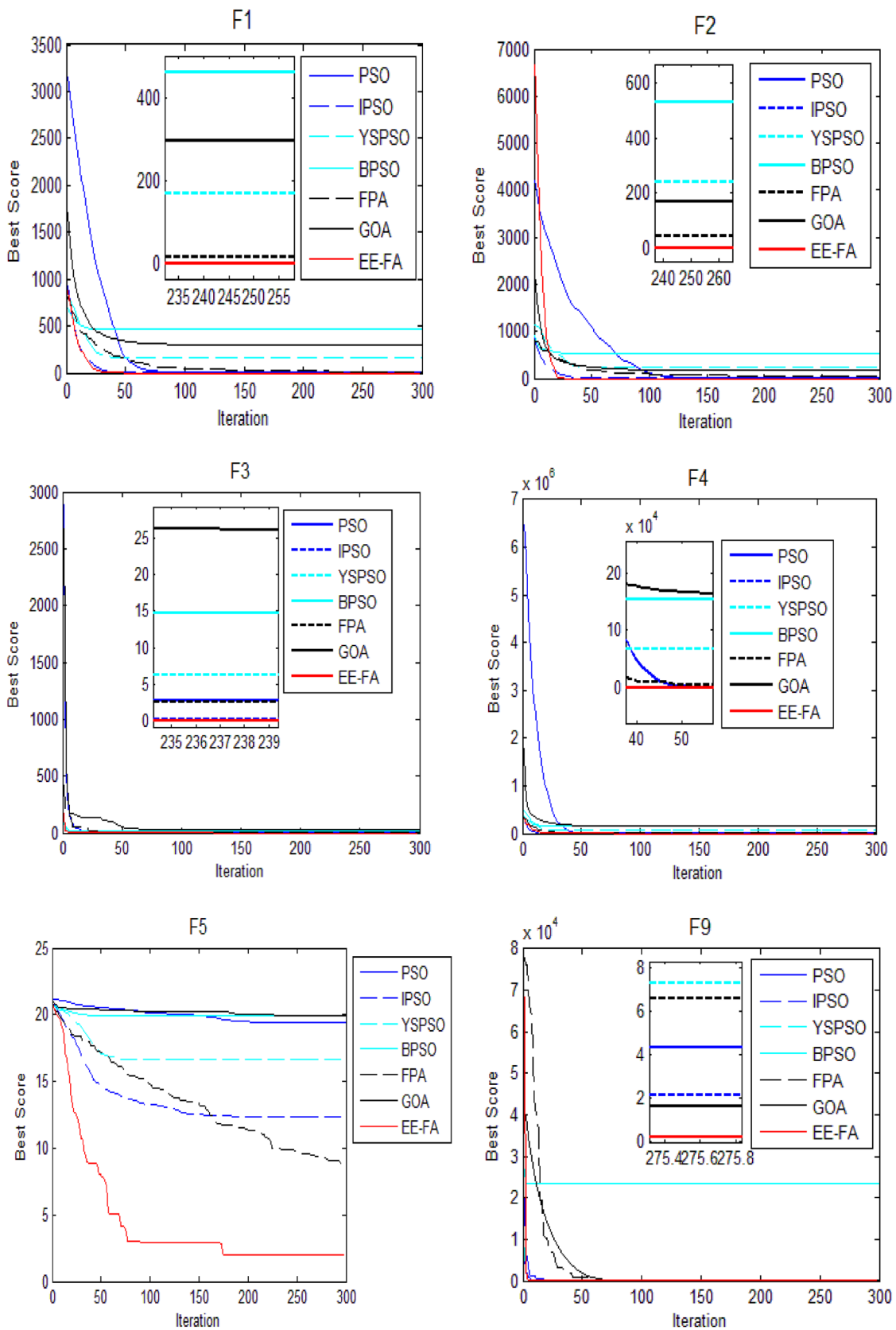


Fig. 12 Simulation Curves for test Functions

Table 6 Results and comparison

Function	FA		LF-FA		PSSFA		IFA		FA-IFF		AD-IFA		EE-FA	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
<i>D</i> = 10														
F1	4.6E-11	8.0E-11	1.2E-10	1.4E-10	1.0E-07	1.3E-07	2.2E-06	3.0E-06	8.1E-11	1.1E-10	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F2	2.1E+03	8.4E+03	2.5E+02	5.6E+02	1.5E+03	1.8E+03	3.3E+3	3.3E+03	2.5E+01	5.7E+01	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F3	1.0E+02	2.6E+01	1.4E+01	1.0E+01	2.7E+01	8.7E+00	5.3E+01	2.3E+01	3.5E+00	3.4E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F4	1.6E+06	8.9E+02	8.9E+02	1.9E+03	4.4E+03	6.2E+03	5.5E+04	5.3E+04	7.0E-02	1.5E-01	0.0E+00	0.0E+00	0.0E+00	0.0E+00
<i>D</i> = 10														
F5	1.8E+01	4.4E-01	1.3E+01	1.6E+00	1.8E+01	4.3E-01	1.8E+1	4.3E-01	1.2E+01	4.1E+00	8.8E-16	0.0E+00	8.8E-16	0.0E+00
F6	1.0E+04	4.0E+03	1.0E+02	1.8E+02	1.3E+02	1.8E+02	1.3E+03	1.0E+03	4.1E-09	4.3E-09	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F7	2.8E+01	1.9E+01	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F8	2.9E+09	1.3E+09	3.2E+05	5.6E+05	6.2E+06	6.7E+06	6.0E+07	4.8E+07	7.4E+03	1.6E+04	3.7E+07	5.4E+07	0.0E+00	0.0E+00
F9	2.4E+02	4.4E+01	6.7E-10	1.0E-09	9.1E-04	9.7E-04	1.4E+00	2.8E+00	9.1E-10	7.4E-10	0.0E+00	0.0E+00	0.0E+00	0.0E+00
<i>D</i> = 16														
F10	1.1E+01	2.6E-01	5.0E-01	7.3E-01	1.1E+01	3.1E-01	1.1E+01	2.7E-01	5.5E-05	1.0E-05	8.8E-16	0.0E+00	8.8E-16	0.0E+00
F11	4.0E+04	7.5E+03	1.0E+03	1.3E+03	5.7E+03	2.6E+03	1.2E+04	4.3E+03	9.2E+01	1.9E+02	4.7E-04	1.0E-03	0.0E+00	0.0E+00
F12	1.0E+03	1.6E+03	0.0E+00	0.0E+00	1.0E+00	1.4E+00	9.2E+00	1.3E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F13	1.2E+09	2.0E+08	3.1E+03	6.2E+03	2.4E+05	3.2E+05	5.6E+05	2.5E+05	2.3E+02	3.0E+02	3.0E+02	4.1E+02	0.0E+00	0.0E+00
F14	5.9E+01	1.5E+01	1.4E-08	1.8E-08	9.1E-03	5.5E-03	4.7E-02	1.8E-02	1.5E-08	9.1E-09	0.0E+00	0.0E+00	0.0E+00	0.0E+00
<i>D</i> = 32														
F15	1.7E+01	3.8E-01	1.4E+01	1.0E+00	1.7E+01	4.2E-01	1.7E+01	3.8E-01	1.3E+01	1.6E+00	1.0E+00	2.2E+00	8.8E-16	0.0E+00
F16	8.4E+04	1.3E+04	7.7E+03	4.2E+03	1.3E+03	5.2E+03	2.8E+04	8.4E+03	2.0E+03	1.6E+03	4.7E+02	4.3E+02	0.0E+00	0.0E+00
F17	5.7E+01	4.4E+01	0.0E+00	0.0E+00	1.0E+00	1.4E+00	9.2E+00	1.3E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F18	4.5E+11	2.3E+11	1.7E+07	2.0E+07	8.1E+08	7.1E+08	3.5E+09	1.5E+09	4.4E+06	6.0E+06	3.0E+05	6.7E+05	0.0E+00	0.0E+00
F19	1.1E+01	2.7E+00	3.1E-04	6.2E-03	7.6E-02	1.6E-02	2.1E-01	8.0E-02	4.9E-05	8.2E-05	0.0E+00	0.0E+00	0.0E+00	0.0E+00
<i>D</i> = 64														
F20	1.3E+01	2.3E-01	9.7E+00	4.0E-01	1.3E+01	2.3E+01	1.3E+01	1.4E-01	7.7E+00	1.2E+00	8.4E+00	1.5E+00	8.8E-16	0.0E+00
F21	9.5E+04	7.8E+03	7.9E+03	3.3E+03	5.4E+03	2.2E+03	2.0E+04	2.6E+03	1.5E+03	1.0E+03	4.0E+03	1.8E+03	0.0E+00	0.0E+00
F22	3.0E-04	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
F23	6.2E+12	1.8E+12	2.7E+09	2.2E+09	1.9E+10	9.8E+09	1.0E+11	8.2E+10	8.1E+07	2.8E+07	1.8+09	3.3E+09	0.0E+00	0.0E+00
F24	4.5E+00	9.2E-01	4.0E-01	1.8E-01	6.2E-01	1.0E-01	5.2E-01	1.8E-01	2.6E-01	1.0E-01	0.0E+00	0.0E+00	0.0E+00	0.0E+00

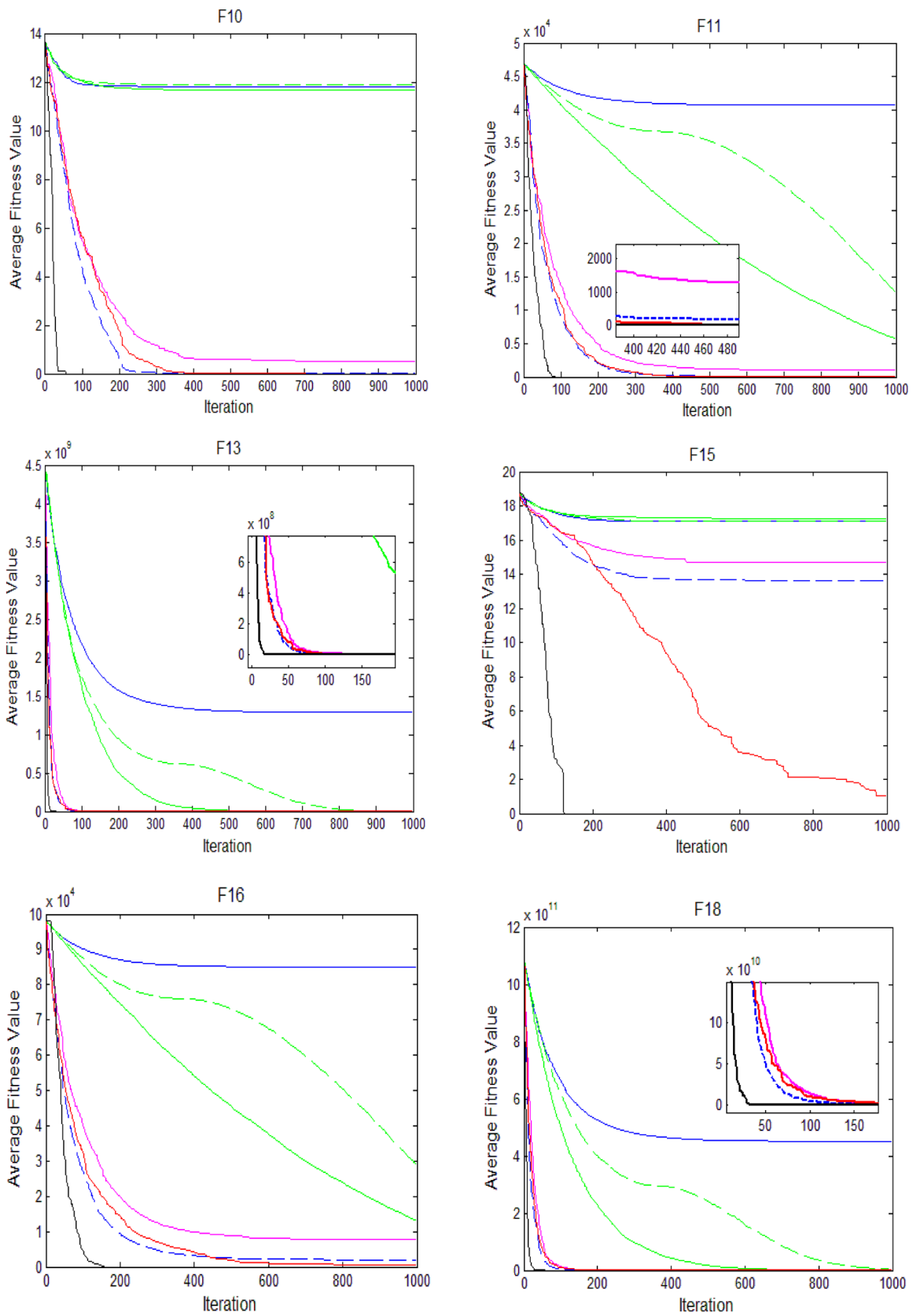


Fig. 13 Simulation curves for test functions

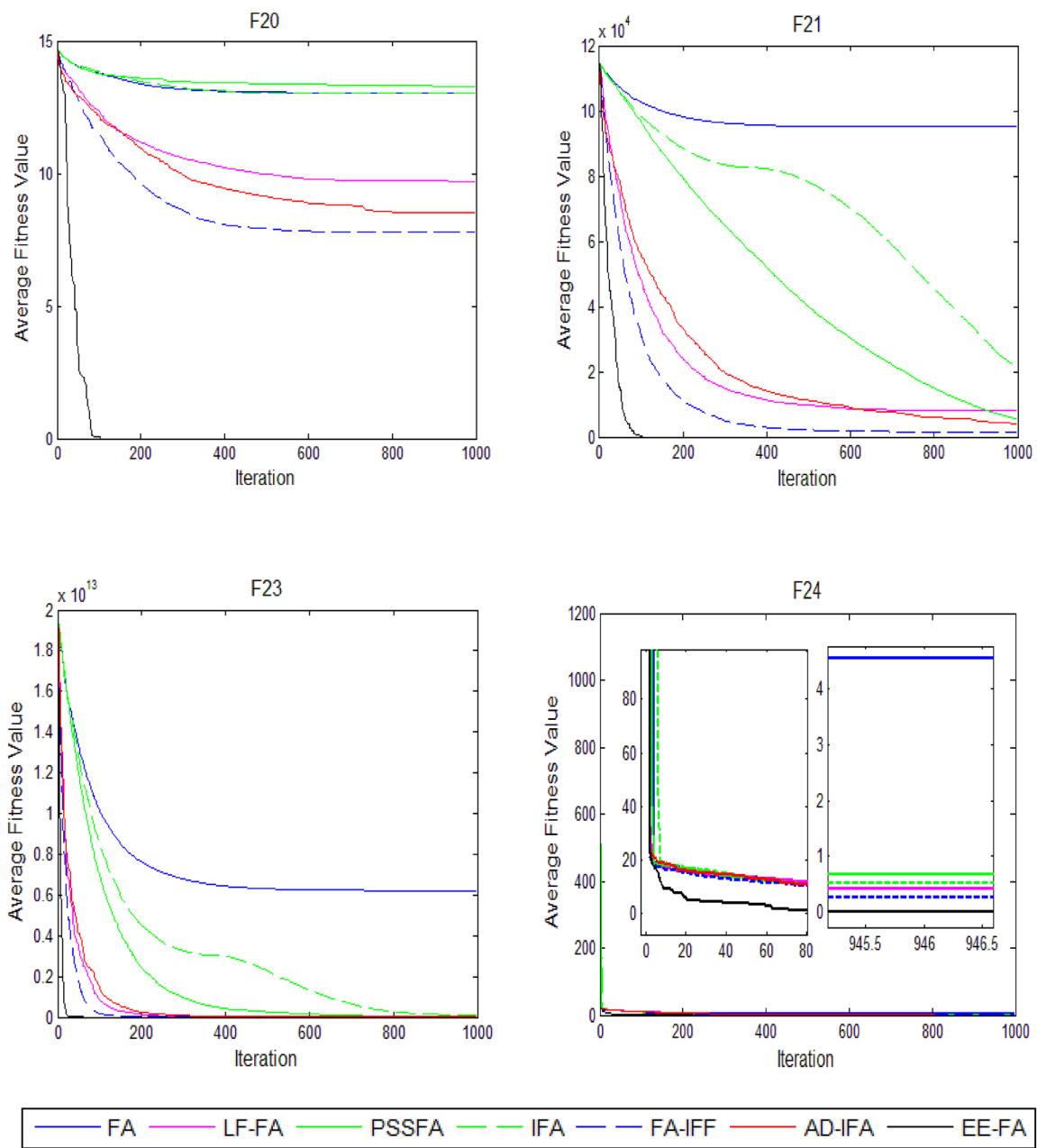


Fig. 13 (continued)

$$\begin{cases}
 x_{i,t+1} = x_{i,t} + A_{i,j} + B_{i,j} \\
 A_{i,j} = 5.743889 \cdot e^{-(1.28 + \text{iter} \times (1/\text{itermax})) \cdot \left(\frac{1/75}{1+(15/75) \cdot r_{i,j}}\right) \cdot r_{i,j}^2} \cdot (x_{j,t} - x_{i,t}) \\
 B_{i,j} = 0.015 \cdot (1.5 \cdot A_{i,j} \cdot (\text{rand} - 0.5) + 7 \cdot e^{-|x_{j,t} - x_{i,t}|} \cdot e^{2(\text{rand} - 0.5)} \cdot \sin(2 \cdot \pi \cdot (\text{rand} - 0.5)))
 \end{cases} \quad (27)$$

Fig. 14 Schematic of a cantilever beam

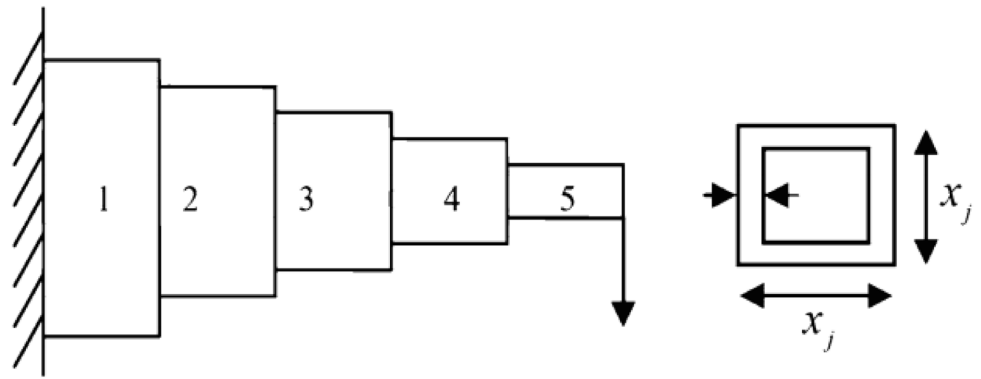


Fig. 15 The iterative trends of the 7 algorithms

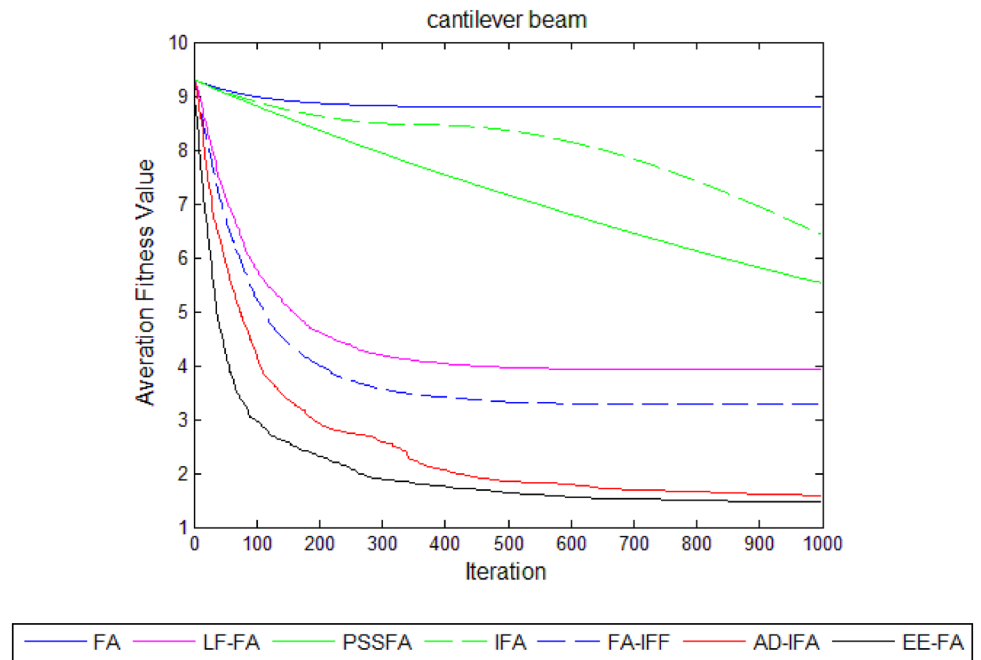
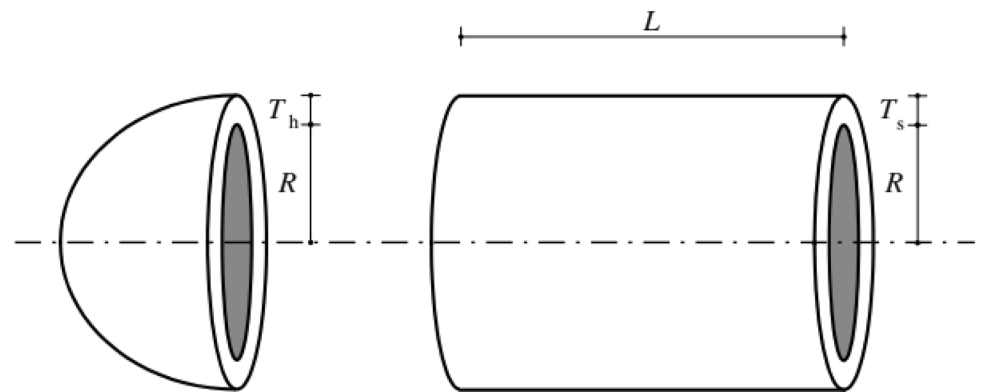


Fig. 16 Schematic of pressure vessel. T_s shell thickness, T_h spherical head thickness, R radius of cylindrical shell, and L shell length



T_s : Shell thickness, T_h : Spherical head thickness, R : Radius of cylindrical shell, and L : Shell length

Algorithm 1 The enhanced exploration firefly algorithm (EE-FA)

```

The fitness function  $f(x)$ ,  $x=(x_1,x_2,x_3,\dots,x_d)^T$ 
Initialize a population of fireflies  $x_i$  ( $i=1,2,3,\dots,n$ )
While  $iter < itermax$  do
  for  $i=1:n$  all  $n$  fireflies do
    for  $j=1:n$  all  $n$  fireflies (inner loop) do
      Light intensity  $l_i$  at  $x_i$  is determined by  $f(x_i)$ 
      if  $l_i < l_j$  then
        Update the  $i$ th firefly's position using Eq. 27
      end if
      Evaluate new solutions and update the light intensity
    end for
  end for
  Rank the fireflies and find the current best fitness function value  $f_{iter}^*$ 
   $iter=iter+1$ 
end while
Output results and visualization
    
```

4.3 Comparison of the time complexity of the EE-FA and AD-IFA

The time complexity of the FA and its variants depend on the population size (n), the dimensional size (d), the maximum number of evaluations (Max_iter), and the type of algorithm. The population size (n), dimensional size (d) and maximum number of evaluations (Max_iter) were the same. The FA and its variants differ greatly in time complexity due to different types of algorithms (Tables 1, 2).

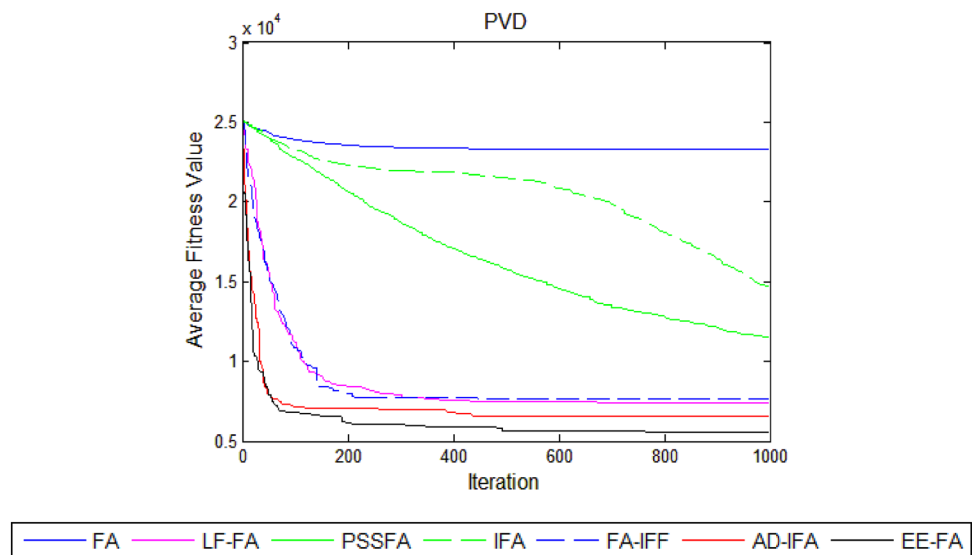
The number of independent simulations for each benchmark function is 100. Then, we calculate the total running time of each variant algorithm. The other experimental conditions of the EE-FA and AD-IFA remained the same.

Time complexity is measured based on the running time on the same computer in the same environment. F1–F9 are described in Table 4.

The data in Table 3 show that the total time for 100 individual runs of the AD-IFA under the same environment and conditions is much higher than that of the EE-FA. This reflects that the AD-IFA has high time complexity, high computational costs and a long running time. The EE-FA removes the adaptive switching mechanism involved in the AD-IFA. The mechanism increases the time complexity and computational costs.

In short, compared to the AD-IFA, the EE-FA is also cheaper to calculate.

Fig. 17 The iterative trends of seven algorithms



4.4 Test function

We choose benchmark functions with different shapes for the simulation experiment [42, 43] in Table 4. Most of these benchmark functions have multiple local minimums or have bowl, plate and valley shapes.

4.5 Compared with other smart swarm algorithms

We selected PSO and its variants, the FPA, the GOA and the EE-FA, for a performance comparison. Standard function F1–F9 is selected, the search space dimension $D=5$, each algorithm is simulated 30 times separately, and the number of search agents is 15.

The comparative data between the EE-FA and the selected object reveal that the mean and std are the smallest. The EE-FA has the highest exploration efficiency. The

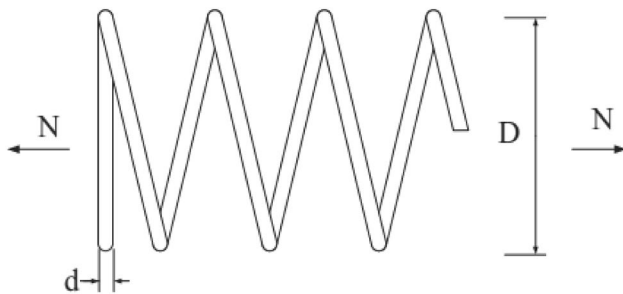
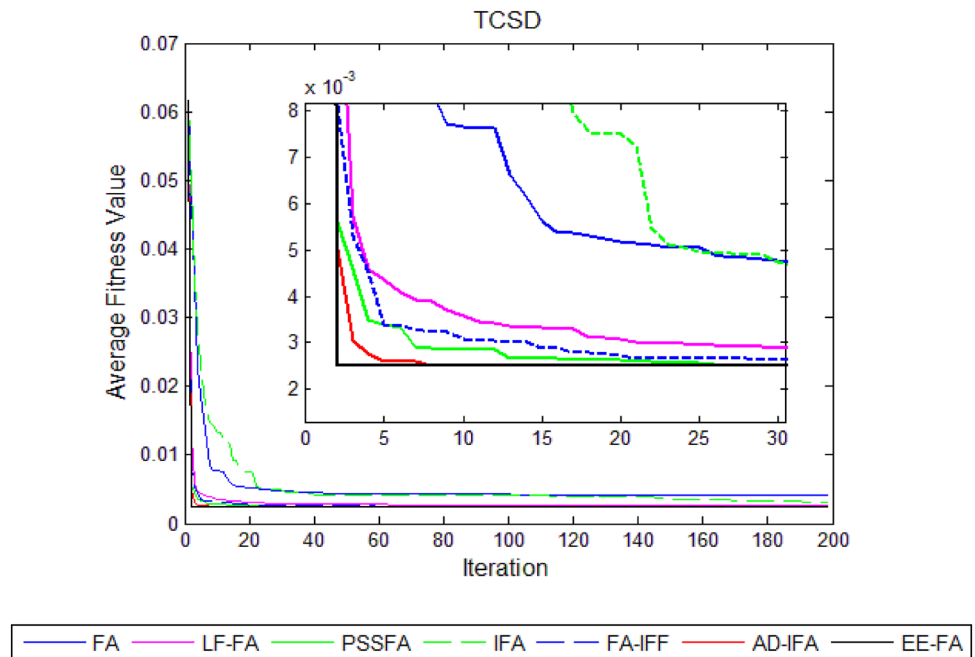


Fig. 18 Schematic of tension/compression spring

Fig. 19 The iterative trends of seven algorithms



data from Table 5 prove that the new random item module greatly improves the FA’s exploration capabilities.

Figure 12 shows that the EE-FA has the fastest iteration speed and the best exploration capability. The improvement in the convergence speed is mainly attributed to the attractiveness item and the random item module. The enhancement of the attractiveness item speeds up convergence.

The improvement of the EE-FA’s exploration efficiency is attributed to the reconstructed random term module.

4.6 Comparison of FA and its variant algorithms

The experimental result data are recorded in Table 6.

As shown in Table 6, for F1, F2, F3, F4, F5, F6, F9, F10, F14, F19 and F24, the minimum mean values of the EE-FA and the AD-IFA are significantly smaller than those of the FA, LF-FA, PSSFA, IFA and FA-IFF. EE-FA and AD-IFA. The EE-FA and AD-IFA have the same exploration efficiency. For F8, F11, F13, F15, F16, F18, F20, F21 and F23, the minimum mean value of the EE-FA was the smallest. The exploration efficiency of the EE-FA is the highest.

For the global continuous optimization problem of 24 benchmark functions, the EE-FA has the highest exploration efficiency, followed by the AD-IFA. Their exploration efficiency is significantly better than that of other algorithms.

Figure 13 shows that the EE-FA has the best exploration efficiency and convergence behaviour. The attractiveness item module enhances the attraction between long-distance fireflies and improves the convergence efficiency. The reconstructed random term has greatly improved the EE-FA’s exploration efficiency.

F20, F21 and F24 reveal that the EE-FA has a much better global optimization capability than the FA, LF-FA, PSSFA, IFA, FA-IFF and AD-IFA. For F11, F13 and F16, the convergence speed of the EE-FA is the fastest. For F15 and F20, other algorithms fall into local optima, but the EE-FA can effectively avoid the situation.

Overall, the EE-FA is superior to the FA, LF-FA, PSSFA, IFA, FA-IFF and AD-IFA in terms of both convergence speed and exploration efficiency. This is due to the seven important optimality constants in the EE-FA: β , γ , b , h , k , f , and g .

Finally, the proposed EE-FA enhances the exploration efficiency and speeds up convergence while avoiding the problem of falling into local optima.

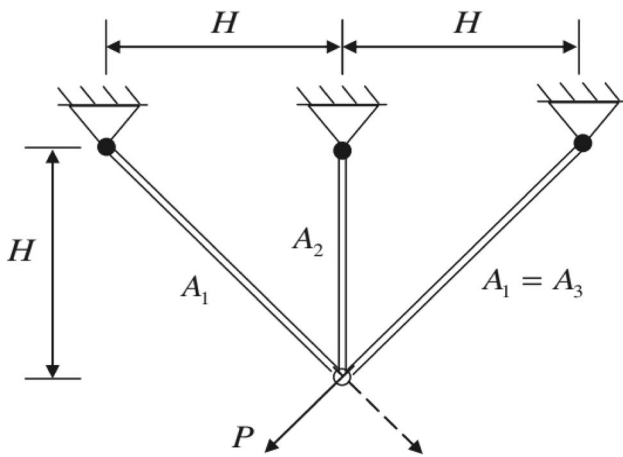
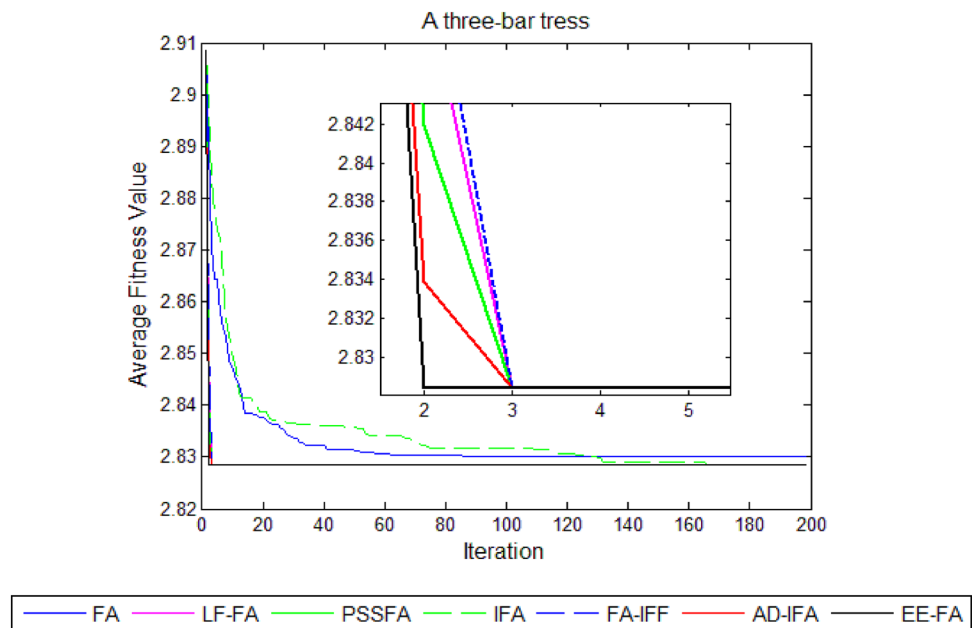


Fig. 20 Schematic of a three-bar truss

Fig. 21 The iterative trends of seven algorithms



5 Real application cases

In this section, the performance of seven algorithms, the FA, LF-FA, PSSFA, IFA, FA-IFF, AD-IFA and EE-FA, are evaluated in the following four engineering real applications: cantilever beam design [44], pressure vessel design [45], tension/compression spring design [46], and a three-bar truss design [45].

Regarding the parameter settings of the FA, LF-FA, PSSFA, IFA, FA-IFF and AD-IFA, α , γ , and β_0 are set to 0.2, 1, and 1, respectively. The dimensional space settings are as follows: cantilever beam ($d=5$), pressure vessel design ($d=4$), tension/compression spring design ($d=3$), and a three-bar truss design ($d=2$). The corresponding maximum numbers of iterations are the following: 1000, 1000, 200, and 200. The experiment of each case was repeated 50 times to evaluate the performance of the 7 algorithms. The experimental results are recorded in the corresponding table.

5.1 Cantilever beam design

This case is related to the weight optimization of a cantilever beam with a square cross section (Figs. 14, 15). The bound constraints are set as $0.01 \leq x_j \leq 100$. This problem can be expressed analytically as follows:

$$\begin{aligned} \min f(x) &= 0.0624(x_1 + x_2 + x_3 + x_4 + x_5) \\ \text{s.t. } g &= \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 0, \end{aligned}$$

where $0.01 \leq x_1, x_2, x_3, x_4, x_5 \leq 100$.

Case	Optimizer	Mean of optimal variables					Mean of optimal value $f(x)$
		x_1	x_2	x_3	x_4	x_5	
Cantilever beam	FA	26.5909	28.0631	30.3021	25.5186	30.3066	8.7848
	LF-FA	11.7234	14.147	14.2108	10.6045	12.1334	3.9199
	PSSFA	16.9475	18.2402	19.3348	14.8616	19.2409	5.5302
	IFA	19.9816	21.113	21.7019	19.0129	21.1807	6.4266
	FA-IFF	9.42642	12.044	12.2957	9.02233	9.6568	3.2714
	AD-IFA	6.4233	6.3218	5.6213	4.4494	2.6913	1.5916
	EE-FA	6.1428	5.6241	4.917	4.0009	2.7068	1.4596

5.2 Pressure vessel design

In order to minimize the total cost of a cylindrical PV, the thickness of the shell (T_s), the thickness of the head (T_h), the inner radius (R), and the length of the annular cross section (L) are optimized (Figs. 16, 17).

Then, the optimization problem can be expressed as follows:

$$\min f(T_s, T_h, R, L) = 0.6224T_sRL + 1.7781T_hR^2 + 3.1661T_s^2L + 19.84T_h^2L,$$

$$\text{s.t.} \begin{cases} g_1 = -T_s + 0.0193R \leq 0, \\ g_2 = -T_h + 0.0095R \leq 0, \\ g_3 = -\pi R^2L - \frac{4}{3}R^3 + 1296000 \leq 0, \\ g_4 = L - 240 \leq 0, \end{cases}$$

where $0.0625 \leq T_s, T_h \leq 99 \times 0.0625$, and $0 \leq R, L \leq 200$.

Case	Opti-mizer	Mean of optimal variables				Mean of optimal value $f(x)$
		T_s	T_h	R	L	
PVD	FA	2.6250	1.2500	60.478	104.87	23320.36
	LF-FA	1.6875	0.5625	52.953	102.35	7386.577
	PSSFA	2.1875	0.6875	54.104	103.90	11506.35
	IFA	2.3750	0.8750	54.759	102.99	14664.36
	FA-IFF	1.8750	0.5625	52.039	102.60	7683.466
	AD-IFA	2.5625	0.5000	56.867	66.349	6540.617
	EE-FA	1.3750	0.6250	64.711	13.540	5560.038

5.3 TCSD (tension/compression spring design) problem

Another classic engineering problem is TCSD. In this case, the decision maker wants to minimize the heaviness of a spring (Figs. 18, 19). The problem consists of the following three variables: the diameter (d), the mean coil diameter (D), and the number of dynamic coils (N).

The expressions for TCSD is as follows.

Consider $\vec{x} = [x_1, x_2, x_3] = [d, D, N]$.

Minimize $f(\vec{x}) = (x_3 + 2)x_2x_1$.

Subject to $g_1(\vec{x}) = 1 - \frac{x_2^3x_3}{7178x_1^4} \leq 0$

$$g_2(\vec{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5180x_1^2} \geq 0$$

$$g_3(\vec{x}) = 1 - \frac{140.45x_1}{x_2^3x_3} \leq 0$$

$$g_4(\vec{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0$$

V a r i a b l e r a n g e s
 $0.05 \leq x_1 \leq 2.00, 0.25 \leq x_2 \leq 1.30, 2.00 \leq x_3 \leq 15.0.$
 $2.00 \leq x_3 \leq 15.0.$

Case	Optimizer	Mean of optimal variables			Mean of optimal value $f(x)$
		d	D	N	
TCSD	FA	0.050	0.25005	4.7611	0.0042266
	LF-FA	0.050	0.25000	2.3783	0.0027365
	PSSFA	0.050	0.25000	2.0000	0.0025000
	IFA	0.050	0.25000	2.9578	0.0030986
	FA-IFF	0.050	0.25000	2.0000	0.0025000
	AD-IFA	0.050	0.25000	2.0000	0.0025000
	EE-FA	0.050	0.25000	2.0000	0.0025000

5.4 A three-bar truss design

This problem was first presented by Nowacki in 1973. The volume of a statically loaded 3-bar truss is to be minimized subject to stress (r) constraints on each of the truss members (Figs. 20, 21). The objective is to evaluate the optimal cross-sectional areas ($A1$ and $A2$).

The mathematical formulation is given as follows:

$$\min f(A_1, A_2) = (2\sqrt{2A_1} + A_2)l,$$

$$\text{s.t.} \begin{cases} g_1 = \frac{\sqrt{2A_1 + A_2}}{\sqrt{2A_1^2 + 2A_1A_2}}P - \sigma \leq 0, \\ g_2 = \frac{A_1}{\sqrt{2A_1^2 + 2A_1A_2}}P - \sigma \leq 0, \\ g_3 = \frac{1}{A_1 + \sqrt{2A_2}}P - \sigma \leq 0, \end{cases}$$

with $l = 100 \text{ cm}$, $P = 2\text{KN}/\text{CM}^2$, and $\sigma = 2\text{KN}/\text{CM}^2$ ($0 \leq A_1, A_2 \leq 1$).

Case	Optimizer	Mean of optimal variables		Mean of optimal value $f(x)$
		A_1	A_2	
A three-bar truss	FA	0.98238	0.026953	2.8300
	LF-FA	1	0	2.8284
	PSSFA	1	0	2.8284
	IFA	1	0	2.8284
	FA-IFF	1	0	2.8284
	AD-IFA	1	0	2.8284
	EE-FA	1	0	2.8284

6 Conclusions

Regarding global continuity optimization, the original FA faced the defects of low exploration efficiency and a slow convergence rate. Although the LF-FA enhances the exploration efficiency by introducing the Lévy distribution, it causes its convergence rate to slow down. On the basis of the LF-FA, the AD-IFA introduces a logarithmic spiral path and adopts an adaptive switch mechanism. The adaptive switch mechanism effectively balances the global search and local development performance while greatly increasing the computational costs. The variant PSSFA adopts a personalized step strategy to improve the defect that the FA has fallen into a local optimum. The IFA uses sigmoid attractiveness to improve convergence efficiency and convergence rate. Compared with the PSSFA, IFA and FA_IFF, the EE-FA has better exploration efficiency and convergence behaviour.

By enhancing the attraction between fireflies and introducing a damping vibration function, the EE-FA can effectively improve the exploration efficiency and accelerate the convergence behaviour. Compared with the AD-IFA, the calculation costs of the EE-FA are lower.

Our future work will further try to strengthen the attractiveness of fireflies and study their exploration efficiency and convergence characteristics. Furthermore, we will explore multiple potential directions. These include tailoring the

application of the algorithm in the actual system, such as the privacy protection network in social networking; and the optimal parameter selection in machine learning model training.

Acknowledgements This work utilizes the resources and services provided by Southeast University. The platform for calculating data is supported by the laboratory of Nanjing Institute of Technology. This work was supported by National Natural Science Foundation of China (Grant No. 51705238)

Author contributions JL conceptualization, investigation, methodology, validation, software, writing, revision, review and editing, visualization; JS supervision, project administration, funding acquisition; FH methodology, formal analysis; writing, revision; formal analysis; MD software, revision, review, conceptualization, formal analysis; XZ revision, review, analysis.

Declarations

Conflict of interest All authors declare no conflict interest.

References

1. Horng MH (2012) Vector quantization using the firefly algorithm for image compression. *Expert Syst Appl* 39(1):1078–1091
2. Montiel O, Sepúlveda R, Orozco-Rosas U (2015) Optimal path planning generation for mobile robots using parallel evolutionary artificial potential field. *J Intell Robot Syst* 79(2):237–257
3. Deb K (1995) Optimization for engineering design: algorithms and examples. Prentice-Hall, New Delhi
4. Yang X-S (2010) Firefly algorithm, stochastic test functions and design optimisation. [arXiv:1003.1409](https://arxiv.org/abs/1003.1409)
5. Maeda K, Fukano Y, Yamamichi S, Nitta D, Kurata H (2011) An integrative and practical evolutionary optimization for a complex, dynamic model of biological networks. *Bioprocess Biosyst Eng* 34(4):433–446
6. Horst R, Pardalos PM (1995) Handbook of global optimization. Springer- Science & Business Media, B.V.
7. Kavousi-Fard A, Samet H, Marzbani F (2014) A new hybrid modified firefly algorithm and support vector regression model for accurate short term load forecasting. *Expert Syst Appl* 41(13):6047–6056
8. Su CT, Lee CS (2003) Network reconfiguration of distribution systems using improved mixed-integer hybrid differential evolution. *Power Deliv* 18(3):1022–1027
9. Goldfeld S M, Quandt R E, Trotter H F. (1996) Maximization by quadratic hill-climbing. *Econometrica:journal of the econometric society* .pp.541–551.
10. Abbasbandy S (2003) Improving Newton-Raphson method for nonlinear equations by modified adomian decomposition method. *Appl Math Comput* 145(2):887–893
11. Nelder JA, Mead R (1965) A simplex method for function minimization. *Comput J* 7(4):308–313
12. Widrow B, Stearns D (1985) Adaptive signal processing. Prentice Hall, Englewood Cliffs
13. Liu J, Wang F, Zhao H, Han G (2017) Filtering algorithm and application of fuze echo signal based on LMS principle. *J Proj Rockets Missiles Guidance* 37(06):45–47+56
14. Yang X-S (2009) Firefly algorithms for multimodal optimization. In: International symposium on stochastic algorithm, pp 169–178

15. Chu X, Niu B, Liang JJ et al (2016) An orthogonal-design hybrid particle swarm optimiser with application to capacitated facility location problem. *Int J Bio Inspired Comput* 8(5):268–285
16. Yang X-S (2008) *Nature-inspired Metaheuristic Algorithms*. Luniver Press, Beckington
17. Yang X-S (2012) Flower pollination algorithm for global optimization. In: Durand-Lose J, Jonoska N (eds) *Unconventional computation and natural computation*, vol 7445. Springer, Berlin, pp 240–249
18. Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. *Adv Eng Softw* 105:30–47
19. Li W, Xue M, Jian-guo L (2011) Feature selection and target recognition based on improved particle swarm optimization algorithm. *Comput Eng Des* 32(11)
20. Zhen-long S, Xiao-ye L, Ying W (2015) Improved simple particle swarm optimization algorithm. *Comput Sci* 42(11A)
21. Zwi-Lee G (2003) Discrete Particle swarm optimization algorithm for unit commitment. In: *IEEE power engineering society general meeting*, vol 1, Ontario, Canada, pp 418–424
22. Gandomi AH, Alavi AH (2011) Multi-stage genetic programming: a new strategy to nonlinear system modelling. *Inf Sci* 181:5227–5239
23. Gandomi AH, Yang XS (2011) Benchmark problems in structural optimization. In: Koziel S, Yang XS (eds) *Computation optimization, methods and algorithms*, Chapter 12. Spring, Berlin, pp 267–291
24. Lukasik S, Żak S (2009) Firefly algorithm for continuous constrained optimization tasks. In: *International conference on computational collective intelligence*, pp 97–106
25. Yang X-S, Hosseini SSS, Gandomi AH (2012) Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect. *Appl Soft Comput* 12(3):1180–1186
26. Jati GK et al (2011) Evolutionary discrete firefly algorithm for travelling salesman problem. In: *Adaptive and intelligent systems*. Springer, pp 393–403
27. Sánchez D, Melin P, Castillo O (2017) Optimization of modular granular neural networks using a firefly algorithm for human recognition. *Eng Appl Artif Intell* 64:172–186
28. Yang X-S, He X (2013) Firefly algorithm: recent advances and applications. [arXiv:1308.3898](https://arxiv.org/abs/1308.3898)
29. Frumen O, Fevrier V, Oscar C, Claudia IG, Gabriela M, Patricia M (2017) Ant colony optimization with dynamic parameter adaptation based on interval type-2 fuzzy logic systems. *Appl Soft Comput* 53:74–87
30. Daniela S, Patricia M, Oscar C (2020) Comparison of particle swarm optimization variants with fuzzy dynamic parameter adaptation for modular granular neural networks for human recognition. *J Intell Fuzzy Syst* 38:3229–3252
31. Frumen O, Fevrier V, Oscar C, Patricia M (2016) Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. *Soft Comput* 20:1057–1070
32. Frumen O, Fevrier V, Patricia M, Alberto S, Oscar C (2019) Interval type-2 fuzzy logic for dynamic parameter adaptation in a modified gravitational search algorithm. *Inf Sci* 476:159–175
33. Daniela S, Patricia M, Oscar C (2017) Optimization of modular granular neural networks using a firefly algorithm for human recognition. *Eng Appl Artif Intell* 64:172–186
34. Lagunes ML, Castillo O, Valdez F, Soria J, Melin P (2018) Parameter optimization for membership functions of type-2 fuzzy controllers for autonomous mobile robots using the firefly algorithm. In: *North American fuzzy information processing society annual conference*, pp 569–579
35. Castillo O, Soto C, Valdez F (2018) A review of fuzzy and mathematic methods for dynamic Parameter adaptation in the firefly algorithm. In: *Advances in data analysis with computational Intelligence methods*. Springer, pp 311–321
36. Yang X-S (2010) Firefly algorithm, Levy flights and global optimization. In: *Research and development in intelligent systems*, vol xxvi. Springer, pp 209–218
37. Shuhao Y, Xukun Z, Xianglin F, Zhengyu L, Mingjing P (2021) An improved firefly algorithm based on personalized step strategy. *Computing*. <https://doi.org/10.1007/s00607-021-00919-9>
38. Ao L, Li P, Deng X, Ren L (2021) A sigmoid attractiveness based improved firefly algorithm and its applications in IIR filter design. *Connect Sci* 33(1):1–25. <https://doi.org/10.1080/09540091.2020.1742660>
39. Navid K, Abidhan B, Pijush S, Majidreza N, Annan Z, Danial JA (2021) A novel technique based on the improved firefly algorithm coupled with extreme learning machine (ELM-IFF) for predicting the thermal conductivity of soil. *Eng Comput*. <https://doi.org/10.1007/s00366-021-01329-3>
40. Jinran W, Wang Y-G, Burrage K, Tian Y-C, Lawson B, Ding Z (2020) An improved firefly algorithm for global continuous optimization problems. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2020.113340>
41. Huang J, Cui X, Li D, Feng Y, Lu D (2004) Observation and data analysis in phase space for Pohl pendulum. *Acta Sci Natur Univ Sunyatseni* 43(Suppl):39–41
42. Surjanovic S, Bingham D (2018) Virtual library of simulation experiments: test functions and datasets. <http://www.sfu.ca/~ssurjano>. Accessed 3 Dec
43. Yelghi A, Köse C (2018) A modified firefly algorithm for global minimum optimization. *Appl Soft Comput* 62:29–44
44. Fleury C, Braibant V (1986) Structural optimization: a new dual method using mixed variables. *Int J Methods Eng* 23(3):409–428
45. Gandomi AH, Yang X-S, Alavi AH (2013) Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Eng Comput* 29(1):17–35
46. Dhiman G, Kumar V (2017) Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. *Adv Eng Softw* 114:48–70

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.