IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 22 June 2022, accepted 7 July 2022, date of publication 14 July 2022, date of current version 21 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3190481

RESEARCH ARTICLE

A Novel Enhanced Arithmetic Optimization Algorithm for Global Optimization

JINZHONG ZHANG, GANG ZHANG, YOURUI HUANG, AND MIN KONG
School of Electrical and Optoelectronic Engineering, West Anhui University, Lu'an 237012, China

Corresponding author: Gang Zhang (zhanggang@wxc.edu.cn)

This work was supported in part by the University Synergy Innovation Program of Anhui Province under Grant GXXT-2021-026.

ABSTRACT The arithmetic optimization algorithm (AOA) is based on the distribution character of the dominant arithmetic operators and imitates addition (A), subtraction (S), multiplication (M) and division (D) to find the global optimal solution in the entire search space. However, the basic AOA has some drawbacks of premature convergence, easily falls into a local optimal value, slow convergence rate, and low calculation precision. To improve the overall optimization ability and overcome the drawbacks of the basic AOA, an enhanced AOA (EAOA) based on the Lévy variation and the differential sorting variation is proposed to solve the function optimization and the project optimization. The Lévy variation increases population diversity, broadens the optimization space, enhances the global search ability and improves the calculation precision. The differential sorting variation filters out the optimal search agent, avoids search stagnation, enhances the local search ability and accelerates the convergence rate. The EAOA realizes complementary advantages of the Lévy variation and the differential sorting variation to avoid falling into the local optimum and the premature convergence. The sixteen benchmark functions and five engineering design projects are applied to verify the effectiveness and feasibility of the EAOA. The EAOA is compared with other algorithms by minimizing the fitness value, such as artificial bee colony, ant line optimizer, cuckoo search, dragonfly algorithm, moth-flame optimization, sine cosine algorithm, water wave optimization and arithmetic optimization algorithm. The experimental results show that the overall optimization ability of the EAOA is superior to that of other algorithms, the EAOA can effectively balance the exploration and the exploitation to obtain the best solution. In addition, the EAOA has a faster convergence rate, higher calculation precision and stronger stability.

INDEX TERMS Arithmetic optimization algorithm, Lévy variation, differential sorting variation, benchmark function, engineering design.

I. INTRODUCTION

The optimization technique is used to describe the complex problems in mathematical form, which adopts certain mathematical logic to abstract the optimization scheme of the problem and obtain the global optimal solution of the problem. That is to say, under certain constraints, the optimization technology finds the best solution from many candidate solutions or search agents to minimize the quality cost, efficiency cost, risk cost and profit cost. As the scale and complexity increase, the traditional optimization methods have the limitations of low computational efficiency, long time consumption, easy to fall into local optimum, and combinatorial explosion. The essence of the meta-heuristic optimization algorithms is to simulate the independent search or complex intelligent behavior of each search agent through mutual cooperation, the search agent is used to adjust its position and update the global optimal solution according to the surrounding iteration information. The meta-heuristic optimization algorithms have some advantages of high operation efficiency, good flexibility, strong stability, good self-organization, easy expansion, simple implementation, strong parallelism and easy combination with other algorithms. The algorithm uses the global search ability and the local search

The associate editor coordinating the review of this manuscript and approving it for publication was Hisao Ishibuchi.

ability to find the optimal solution. Some optimization algorithms have been explained the optimization problem, such as the artificial bee colony (ABC) [1], the ant line optimizer (ALO) [2], the cuckoo search (CS) [3], the dragonfly algorithm (DA) [4], the moth-flame optimization (MFO) [5], the sine cosine algorithm (SCA) [6] and water wave optimization (WWO) [7]. The meta-heuristic optimization algorithms are divided into several categories, such as biology-based, social-based, chemical-based, physics-based, music-based, mathematics-based, sports-based, swarm-based, plant-based and water-based [8]–[11]. The artificial bee colony, ant line optimizer, cuckoo search, dragonfly algorithm and moth-flame optimization are biologically based.

Li *et al.* designed a chaotic AOA to solve the benchmark functions and four engineering design issues, the optimization results showed that the improved algorithm has better optimization accuracy and efficiency [12]. Mahajan *et al.* combined the AOA with the aquila optimization algorithm to solve the global optimization problem, which can avoid falling into the local optimal for efficient optimization. The results showed that the proposed algorithm had certain advantages to enhance the optimization results [13]. Kaveh and Hamedani used discrete design variables and designed an improved AOA to solve the discrete optimization design. The results showed that the proposed algorithm had a strong global search ability and local search ability to obtain the best solution [14]. Hu *et al.* combined the AOA based on point set strategy, optimal neighborhood learning strategy and crisscross strategy to improve the convergence speed and calculation accuracy. The improved algorithm balanced the exploitation and exploration to efficiently complete the function optimization and engineering optimization [15]. Mahajan *et al.* combined AOA with the hunger games search algorithm for function optimization problems. Compared with other algorithms, the proposed algorithm had better superiority and stronger stability [16]. Zhang *et al.* proposed a hybrid optimization algorithm of AOA and aquila optimization algorithm to solve the mathematical optimization problems, the hybrid algorithm adopted the exploration and exploitation to obtain the optimal solution [17]. Abualigah *et al.* designed a hybrid optimization algorithm of flow direction algorithm and AOA to solve the optimization problems of data clustering. The improved algorithm can take advantage of the two algorithms to overcome premature convergence and fall into the local optimum. The results showed that the hybrid algorithm has certain effectiveness and feasibility to complete the optimization problem [18]. Liu *et al.* proposed AOA with a golden sine algorithm to solve the engineering design problem, and the optimization results of the proposed algorithm were better than those of other algorithms [19]. Pashaei and Pashaei introduced a hybrid binary AOA with a simulated annealing algorithm to solve the feature selection problem, the proposed algorithm obtained better classification accuracy and optimization results [20]. Liu *et al.* proposed an improved AOA based on circle chaotic mapping, elite mutation approach and Cauchy disturbances to solve the function optimization and the engineering design problems, the optimization results of the proposed algorithm were better than those of other algorithms [21]. Khodadadi *et al.* designed a dynamic AOA to solve the truss optimization problems, the proposed algorithm balanced exploration and exploitation to find the global solution in the search space [22]. Zheng *et al.* created an improved AOA based on forced switching mechanism to solve the function optimization and the engineering design problems, the results showed that the proposed algorithm had a strong the global search ability and the local search ability to avoid premature convergence and find the optimal solution [23]. To summarize, the research of the AOA mainly contains two aspects: algorithm improvement and algorithm application [24], [25]. For algorithm improvement, introducing effective search strategies, adopting unique coding methods, or combining with other swarm intelligence algorithms achieves complementary advantages and improves the overall optimization ability. The improved AOA can effectively balance exploration and exploitation to avoid premature convergence and fall into the local optimum, and then improve the convergence rate and the calculation precision. For algorithm application, the improved AOA has strong stability and superiority, and it has a wide range of application prospects in the artificial intelligence, system control, pattern recognition, resource allocation, engineering technology, network communication, finance and other fields.

The basic AOA, which is inspired by the distribution character of the dominant arithmetic operators, obtains the best solution in the whole search space by imitating addition (A), subtraction (S), multiplication (M) and division (D) [26]. The AOA is mathematics-based. To improve the overall search ability, the Lévy variation [27] and the differential sorting variation [28], [29] are introduced into the basic AOA. The Lévy variation increases population diversity, broadens the optimization space and enhances the global search ability. The differential sorting variation filters out the optimal search agent, avoids the search stagnation and enhances the local search ability. The EAOA achieves complementary advantages of the Lévy variation and the differential sorting variation to balance the global search ability and the local search ability. The EAOA is used to solve the function optimization and the project optimization. The experimental results show that the EAOA has a faster convergence rate, higher calculation precision and stronger stability.

The article is divided into the following sections. Section II introduces the AOA. Section III depicts the EAOA. The experimental results and discussion are described in Section IV. Finally, the conclusions and future research are provided in Section V.

II. AOA

The AOA is based on the distribution character of the dominant arithmetic operators to find the best solution in the search space, which contains four operators: addition (A "+"), subtraction (S "−"), multiplication (M "×") and division

(D "÷"). In AOA, each individual represents a search agent. The corresponding relationship between the problem space and the population space is as follows: the solution space corresponds to the search space of the AOA, each solution corresponds to each search agent, and the fitness value of each solution corresponds to the fitness value of the AOA. The AOA adopts exploration or exploitation to solve the optimization problem.

A. INITIALIZATION

In AOA, candidate solutions are randomly generated during the initial population phase. When the iteration of the AOA is continuously updated, the purpose of optimization is to find an optimal or sub-optimal solution from many candidate solutions. The matrix is estimated as follows:

$$X = \begin{bmatrix} x_{1,1} & \cdots & \cdots & x_{1,j} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & \cdots & x_{2,j} & \cdots & x_{2,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & \cdots & x_{N-1,j} & \cdots & x_{N-1,n} \\ x_{N,1} & \cdots & \cdots & x_{N,j} & x_{N,n-1} & x_{N,n} \end{bmatrix} \tag{1}$$

where N is the population size, n is the dimension of the search space, $x_{i,j}$ is the position of ith solution in the jth search space. In AOA, the math optimizer accelerated (MOA) is regarded as an adaptive coefficient for selecting exploration or exploitation to find the global optimal solution. The function is estimated as follows:

$$MOA(C\_Iter) = Min + C\_Iter \times (\frac{Max - Min}{M\_Iter}) \tag{2}$$

where $MOA(C\_Iter)$ is a calculated value, C_Iter is the current iteration. Max or Min are the maximum or minimum values of MOA. In this paper, $Max = 1$ and $Min = 0.2$. The control parameter $r_1 \in [0, 1]$ is a uniformly distributed random number. If $r_1 > MOA$, the AOA performs exploration. If $r_1 \leq MOA$, the AOA performs exploitation.

B. EXPLORATION

In exploration, the AOA utilizes multiplication (M "×") and division (D "÷") to obtain a distribution solution. These two search mechanisms are difficult to find the objective solution due to the high degree of discreteness. The AOA can randomly obtain the global optimal solution according to multiplication (M) and division (D). The search process is obtained by calculating the MOA in the case where $r_1 > MOA$. If $r_2 < 0.5$, the AOA uses division (D) to complete the search task. Otherwise, the AOA uses multiplication (M) to achieve the optimization process. The position update is estimated as follows:

$$x_{i,j}(C\_Iter + 1) = \begin{cases} best(x_j) \div (MOP + \varepsilon) \\ \quad \times ((UB_j - LB_j) \times \mu + LB_j) & if\ r_2 < 0.5 \\ best(x_j) \times MOP \\ \quad \times ((UB_j - LB_j) \times \mu + LB_j) & if\ r_2 \geq 0.5 \end{cases} \tag{3}$$

where r_2 is a random number in [0,1], $best(x_j)$ is the optimal position of the jth search agent, ε is an infinitesimal integer number. UB or LB are the upper or lower boundary, respectively. μ is an adjusted parameter and the value is 0.5.

$$MOP(C\_Iter) = 1 - \frac{C\_Iter^{1/\alpha}}{M\_Iter^{1/\alpha}} \tag{4}$$

where math optimizer probability (MOP) is a factor, $MOP(C\_Iter)$ is a calculated solution, C_Iter is present iteration, M_Iter is maximum iteration. α is a sensitive parameter and the value is 5.

C. EXPLOITATION

In exploitation, the AOA utilizes the addition (A "+") and subtraction (S "−") to obtain a higher precision solution. These two search mechanisms are easy to gain the objective solution due to the low dispersion. The search process is obtained by calculating the MOA in the case where r_1 is less than the MOA. If $r_3 < 0.5$, the AOA uses subtraction (S) to complete the current search plan. Otherwise, the AOA uses addition (A) to achieve the optimization process. The AOA utilizes the local mechanism in several dense areas to attain the fitness value. The position update is estimated as follows:

$$x_{i,j}(C\_Iter + 1) = \begin{cases} best(x_j) - MOP \\ \quad \times ((UB_j - LB_j) \times \mu + LB_j) & if\ r_3 < 0.5 \\ best(x_j) + MOP \\ \quad \times ((UB_j - LB_j) \times \mu + LB_j) & if\ r_3 \geq 0.5 \end{cases} \tag{5}$$

where r_3 is a random number in [0,1], μ is an adjusted parameter and the value is 0.5.

The solution process of the AOA is expressed in Algorithm 1.

III. EAOA

The Lévy variation and the differential sorting variation are introduced into the basic AOA, which achieves complementary advantages to avoid the search stagnation and premature convergence. The EAOA can effectively balance the global search ability and the local search ability to improve the convergence rate and the calculation precision.

A. LÉVY VARIATION

The variation based on a haphazard walk mechanism extends the solution area and intensifies the optimization performance. The search method promotes calculation precision to a certain extent. The position is estimated as follows:

$$X_{i,j}(C\_Iter + 1) = X_{i,j}(C\_Iter) + \mu sign[rand - 1/2] \oplus Levy \tag{6}$$

where $X_{i,j}$ is the current position, μ is a random value, $rand$ is a random value [0,1], $sign[rand - 1/2]$ are −1, 0, and 1. $\oplus$ is the entry-wise multiplication.

The position of the Lévy distribution is estimated as follows:

$$Levy(s) \sim |s|^{-1-\beta}, \quad 0 < \beta \leq 2 \tag{7}$$

Algorithm 1 AOA

Initialize the solutions' positions randomly $X_i(i = 1, \ldots, N)$ and initialize parameters α, μ.
Compute the fitness function of a given solution and achieve the best solution x
while $(C\_Iter < M\_Iter)$
  Update the *MOA* value applying Eq. (2).
  Update the *MOP* value applying Eq. (4).
  for ($i = 1$ *to Solutions*)
    for ($j = 1$ *to Positions*)
  Accomplish the random values between [0,1] (r_1, r_2, r_3)
      if $r_1 > MOA$ **then**
        Exploration
        if $r_2 > 0.5$ **then**
        (1) Utilize the division math operator (D "÷")
        Update the *ith* solution's position using the first rule in Eq. (3).
        else
        (2) Utilize the multiplication math operator (M "×")
        Update the *ith* solution's position using the second rule in Eq. (3).
        end
      else
        Exploitation
        if $r_3 > 0.5$ **then**
        (1) Utilize the subtraction math operator (S "−")
        Update the *ith* solution's position using the first rule in Eq. (5).
        else
        (2) Utilize the addition math operator (A "+")
        Update the *ith* solution's position using the second rule in Eq. (5).
        end
      end
    end
  end
  $C\_Iter = C\_Iter + 1$
end
Return the best solution x

where s is step length of Lévy variation, β is a factor, the s is estimated by Mantega's algorithm as follows:

$$s = \frac{\mu}{|v|^{1/\beta}}, \quad \mu \sim N(0, \sigma_\mu^2),\ v \sim N(0, \sigma_v^2) \tag{8}$$

where β is set to 1.5, u and v is obey normal distributions respectively.

$$\sigma_u = \left[\frac{\Gamma(1+\beta) \cdot \sin(\pi\beta/2)}{\beta \cdot \Gamma\left[(1+\beta)/2\right] \cdot 2^{(\beta-1)/2}} \right]^{1/\beta}, \quad \sigma_v = 1 \tag{9}$$

where Γ is the normal gamma sign.

B. DIFFERENTIAL SORTING VARIATION

In this paper, we assign a ranking for each search agent according to its fitness value. The population is sorted in ascending order (i,e., from the best fitness value to the worst fitness value) based on the fitness value of each solution. The ranking of a solution is estimated as follows:

$$R_i = N - i, \quad i = 1, 2, \ldots, N \tag{10}$$

where N is the population size, the solution with the optimal fitness value has a higher ranking.

A sorting operation is performed for each solution. The selection probability P_i is estimated as follows:

$$p_i = \frac{R_i}{N}, \quad i = 1, 2, \ldots, N \tag{11}$$

The differential sorting variation of "DE/rand/1" is expressed in Algorithm 2. A search agent is randomly selected in the population to calculate the selection probability p_{c_i} of its individual and p_{c_i} is compared with a random number [0,1] to determine whether the selection is successful. In nature, the profitable information is enclosed in an excellent population, and better individuals are arranged for the next generation of evolution. A higher ranking individual is used as the basis or eventual vectors of the mutation operator, and the probability of being chosen will increase, which is beneficial to retain the information of better individuals. The choice of the starting vector is not determined by sorting. The two differential vectors are arranged from the optimal vectors, and the corresponding step size decreases rapidly and causes the algorithm to converge prematurely. Therefore, the choice of the starting vector does not depend on sorting. The differential sorting variation of "DE/rand/1" filters out the optimal search agent, avoids the search stagnation, enhances the local search ability and accelerates the convergence rate.

Algorithm 2 Differential Sorting Variation of "DE/rand/1"

Order the population, determine the sorting and selection probability P_i of each given solution
Randomly assign $c_1 \in \{1, \ldots, N\}$ {base vector index}
while $rand\,[0, 1] > p_{c_1}$ *or* $c_1 == i$
Randomly assign $c_1 \in \{1, \ldots, N\}$
end
Randomly assign $c_2 \in \{1, \ldots, N\}$ {terminal vector index}
while $rand\,[0, 1] > p_{c_2}$ *or* $c_2 == c_1$ *or* $c_2 == i$
Randomly assign $c_2 \in \{1, \ldots, N\}$
end
Randomly assign $c_3 \in \{1, \ldots, N\}$ {starting vector index}
while $c_3 == c_2$ *or* $c_3 == c_1$ *or* $c_3 == i$
Randomly assign $r_3 \in \{1, \ldots, N\}$
end

The EAOA has strong practicality and usefulness to receive the best individual. The EAOA is expressed in Algorithm 3. A flowchart of the EAOA is presented in figure 1.

Algorithm 3 EAOA

Initialize the solutions' positions randomly $X_i(i = 1, \ldots, N)$ and initialize parameters α, μ.
Compute the fitness function of a given solution and achieve the best solution x
while $(C\_Iter < M\_Iter)$
Introduce differential sorting variation, order the population and determine the sorting and selection probability P_i of each solution.
  Update the *MOA* value applying Eq. (2).
  Update the *MOP* value applying Eq. (4).
  for $(i = 1 \ to \ Solutions)$
    for $(j = 1 \ to \ Positions)$
  Accomplish the random values between [0,1] (r_1, r_2, r_3)
    if $r_1 > MOA$ **then**
      Exploration
      if $r_2 > 0.5$ **then**
      (1) Utilize the division math operator (D "÷")
      Update the *ith* solution's position using the first rule in Eq. (3).
      else
       (2) Utilize the multiplication math operator (M "×")
      Update the *ith* solution's position using the second rule in Eq. (3).
      end
    else
      Exploitation
      if $r_3 > 0.5$ **then**
      (1) Utilize the subtraction math operator (S "−")
      Update the *ith* solution's position using the first rule in Eq. (5).
      else
      (2) Utilize the addition math operator (A "+")
      Update the *ith* solution's position using the second rule in Eq. (5).
      end
    end
    end
  end
  Update the position of each solution based upon the Lévy flight in Eq. (6).
  Compute the fitness function of a given solution
  Update x if there is a better solution
  $C\_Iter = C\_Iter + 1$
end
Return the best solution x

C. COMPUTATIONAL COMPLEXITY OF EAOA

The computational complexity of the EAOA is briefly analyzed in this section, the EAOA depends on three important operations: initialization, fitness value evaluation, and refreshing solutions. In EAOA, N indicates the population size, M indicates the maximum iteration, and L indicates the dimension of the problem. The computational complexity of initialization is $O(N)$. The computational complexity of fitness value evaluation is determined by the optimization problem, we will not explore it here. Therefore, $O(M \times N) + O(M \times N \times L)$ is the computational complexity of the refreshing solutions. In sum, The computational complexity of the EAOA is $O(N \times (ML + 1))$. In the next section, the function optimization and the project optimization are used to verify the effectiveness and feasibility of the EAOA.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. SIMULATION ENVIRONMENT

The simulation platform is implemented on a computer with an Intel Core i7-8750H 2.2 GHz CPU, a GTX1060, and 8 GB memory with Windows 10 system. All of the algorithms are programmed in MATLAB R2018b. All of the algorithms are programmed in MATLAB R2018b.

B. BENCHMARK FUNCTIONS

To verify the effectiveness and feasibility of the EAOA, the proposed algorithm is applied to solve the function optimization problem. The purpose of optimization is to avoid the algorithm falling into the local optimum and minimize the fitness value of the objective function. The benchmark functions are split into three types: $f_1 - f_6$ are the unimodal functions, $f_7 - f_{10}$ are the multimodal functions, $f_{11} - f_{16}$ are the fixed-dimension multimodal functions. The benchmark functions are described in Table 1.

The control parameters of each algorithm are representative empirical values, which are derived from the original articles. Different optimization algorithms are used to solve the function optimization problem, such as ABC, ALO, CS, DA, MFO, SCA, WWO and AOA. The initial parameters of each algorithm are described in Table 2.

For all comparison algorithms, the population size is 20, the maximum iteration is 1000 and the independent run is 30. Best, Worst, Mean and Std are the optimal value, worst value, mean value and standard deviation, respectively. To reflect the overall optimization performance of the algorithms, the optimal value is exhibited in bold and the ranking is founded on the standard deviation.

In Table 3, for f_1, f_2 and f_3, the EAOA can find the exact global optimization solution, the optimal value, worst value, mean value and standard deviation of the EAOA are superior to those of other algorithms, which shows that the EAOA has a strong overall optimization ability to avoid premature convergence of the algorithm and falling into the local optimum, the EAOA can realize the best solution in the search space. Compared with other algorithms, the ranking of the EAOA is the first, the EAOA not only has a relatively small standard deviation, but also has strong stability and superiority. For f_4 and f_6, the optimal value, worst value, mean value and standard deviation of the EAOA have been significantly strengthened compared to those of the basic AOA, and the optimization values of the EAOA are the

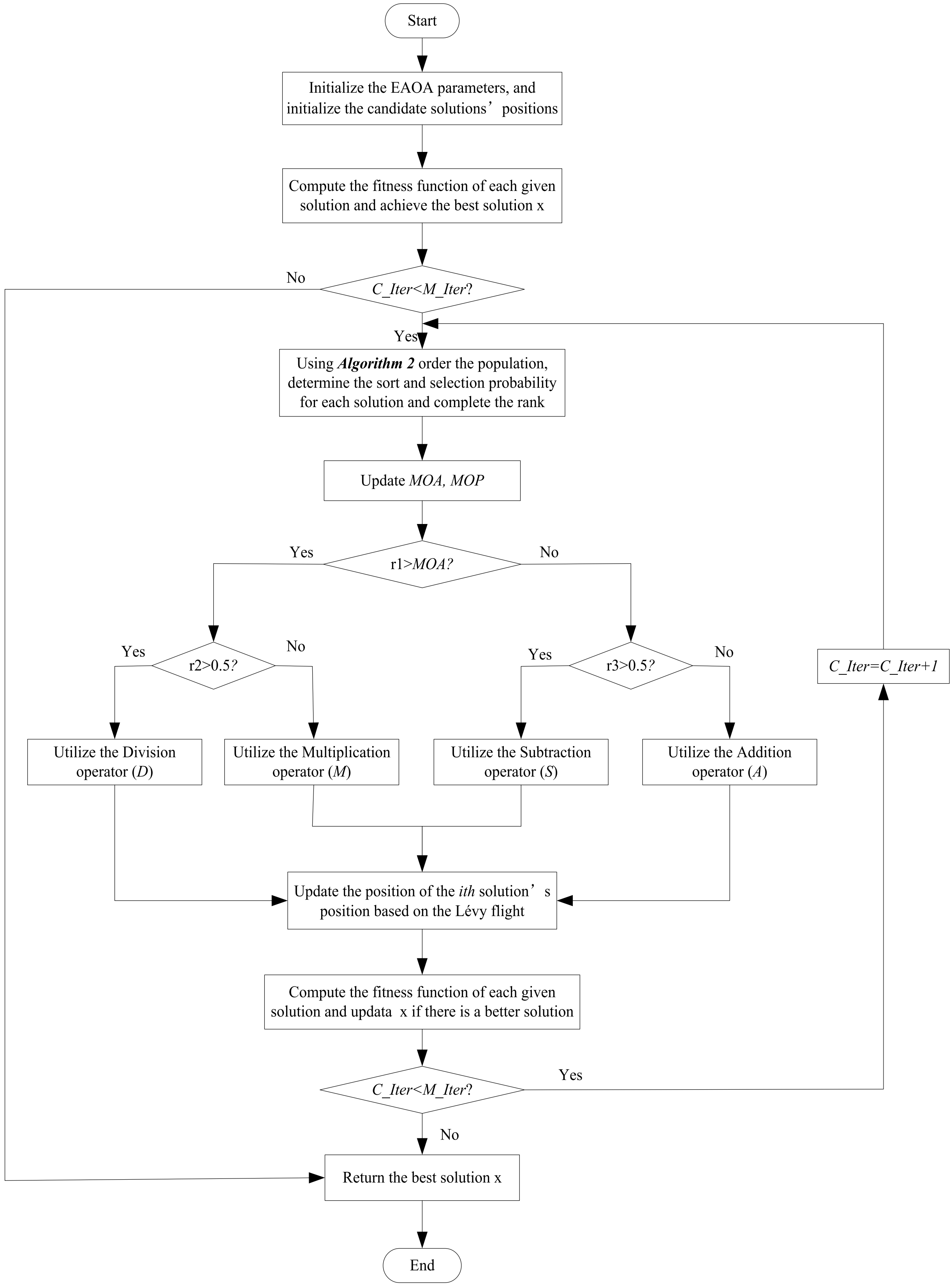


FIGURE 1. Flowchart of EAOA.

TABLE 1. Benchmark functions.

Benchmark Functions	Dim	Range	$f_{\min}$
$f_1=\sum_{i=1}^{n}x_i^2$	30	[-100,100]	0
$f_2(x)=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	30	[-10,10]	0
$f_3(x)=\sum_{i=1}^{n}(\sum_{j=1}^{i}x_j)^2$	30	[-100,100]	0
$f_4(x)=\max_i\{\lvert x_i\rvert,1\le i\le D\}$	30	[-100,100]	0
$f_5(x)=\sum_{i=1}^{D-1}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	30	[-30,30]	0
$f_6(x)=\sum_{i=1}^{n}x_i^4+random(0,1)$	30	[-1.28,1.28]	0
$f_7(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	30	[-5.12,5.12]	0
$f_8(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi x_i\right)\right)+20+e$	30	[-32,32]	0
$f_9(x)=\frac{1}{4000}\sum_{i=1}^{n}\left(x_i^2\right)-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	30	[-600,600]	0
$f_{10}(x)=\frac{\pi}{D}\left\{10\sin^2(\pi y_1)+\sum_{i=1}^{D-1}\begin{matrix}(y-1)^2[1+10\sin^2(\pi y_1)]\\+(y_D-1)^2\end{matrix}\right\}+\sum_{i=1}^{D}u(x_i,10,100,4)$ $y_i=1+\frac{x_i+1}{4}$ $u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m,x^i>a\\0,-a\le x_i\le a\\k(-x_i-z)^m,x_i<a\end{cases}$	30	[-50,50]	0
$f_{11}(x)=(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}(x_i-a_{ij})^6})^{-1}$	2	[-65,65]	1
$f_{12}(x)=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$	2	[-5,5]	-1.0316285
$f_{13}(x)=-\frac{1+\cos(12\sqrt{x_1^2+x_2^2})}{0.5(x_1^2+x_2^2)+2}$	2	[-5.12,5.12]	-1
$f_{14}(x)=-\cos(x_1)\cos(x_2)\exp(-(x_1-\pi)^2-(x_2-\pi)^2)$	2	[-110,100]	-1
$f_{15}(x)=0.5+\frac{\sin^2\left(\sqrt{x_1^2+x_2^2}\right)-0.5}{(1+0.001(x_1^2+x_2^2))^2}$	2	[-100,100]	-1
$f_{16}(x)=\sum_{i=1}^{n}x_i\sin(x_i)+0.1x_i$	10	[-10,10]	0

TABLE 2. Initial parameters of each algorithms.

Algorithms	Parameters	Values
ABC	Maximum number of searches t_{max}	5D
ALO	Haphazard number *rand*	[0,1]
CS	Parameter β	1.5
	Probability P	0.25
DA	Inertia weight ω	[0.2,0.9]
	Separation weight s	0.1
	Alignment weight a	0.1
	Cohesion weight c	0.7
	Food factor f	1
	Enemy factor e	1
MFO	Fixed value b	1
	Haphazard number t	[-1,1]
	Haphazard number r	[-2,-1]
SCA	Haphazard number α	2
	Haphazard number r_2	$[0,2\pi]$
	Haphazard number r_3	[-2,2]
	Haphazard number r_4	[0,1]
WWO	Wavelength λ	0.5
	Wave height h_{max}	12
	Wavelength reduction coefficient α	1.0026
	Breaking coefficient β	[0.01,0.25]
	Maximum number k_{max} of breaking directions	$\min(12, D/2)$
AOA	The minimum value of the accelerated function *Min*	0.2
	The maximum value of the accelerated function *Max*	1
	A sensitive parameter α	5
	A adjust parameter μ	0.5
	Parameter r_1	[0,1]
	Parameter r_2	[0,1]
	Parameter r_3	[0,1]
EAOA	Minimum value of the accelerated function *Min*	0.2
	Maximum value of the accelerated function *Max*	1
	A sensitive parameter α	5
	A adjust parameter μ	0.5
	Scaling value F	0.7
	A power value λ	(1,3]
	Haphazard number β	1.5
	Parameter r_1	[0,1]
	Parameter r_2	[0,1]
	Parameter r_3	[0,1]

greatest in all algorithms, which shows that the EAOA has strong global search ability and local search ability to obtain the best solution. The ranking of the EAOA is first, so that EAOA has strong stability to solve the unimodal functions. For f_5, the optimal value of the EAOA is worse than that of the WWO, but the worst value, mean value and standard deviation of the EAOA are better than those of other algorithms. The ranking and stability of the EAOA are the best in all algorithms. Lévy variation increases population diversity, broadens the optimization space, enhances the global search ability and improves the calculation precision. The differential sorting variation filters out the optimal search agent, avoids the search stagnation, enhances the local search ability and accelerates the convergence rate. The EAOA realizes complementary advantages to avoid falling into the local optimum. Therefore, the EAOA has strong stability and reliability to obtain a faster convergence rate and higher calculation precision.

In Table 4, for f_7, both the basic AOA and the EAOA find the exact excellent solution. The optimal value, worst value, mean value and standard deviation of the EAOA are consistent with those of the AOA. Compared with other algorithms, the optimal value, worst value, mean value and standard deviation of the EAOA are better. The ranking of the EAOA is the first and the EAOA has strong stability. For f_8, the optimal value, worst value, mean value and standard deviation of AOA and EAOA are the same. The optimal value, worst value, mean value and standard deviation of EAOA are superior to those of other algorithms except the AOA. The ranking of the EAOA is the first, which shows that EAOA has excellent stability and superiority to find the global optimal solution. For f_9, the optimal value, worst value, mean value and standard deviation of the EAOA have been improved compared to those of the basic AOA, the optimal value of the EAOA is better than those of other algorithms, the worst value and mean value of the EAOA are better than those of ABC, CS, than those of other algorithms except ALO and WWO. For DA, MFO, SCA and AOA, but the standard deviation of the EAOA is worse than those of ALO and WWO. For f_{10}, the optimal value, worst value, mean value and standard deviation of the EAOA have been improved compared to those of the basic AOA. The optimal value of the EAOA is worse than that of the MFO, but the worst value, mean value and standard deviation of the EAOA are superior to those of ABC, ALO, CS, DA, MFO, SCA, WWO and AOA. The EAOA has the best ranking and strong stability. The Lévy variation has the characteristics of a large search range, wide population diversity and strong global search ability. The differential sorting variation has the characteristics of avoiding premature convergence, filtering out the best search agent and having a strong local search ability. The EAOA realizes complementary advantages to avoid search stagnation. Therefore, the EAOA can switch arbitrarily between global search ability and local search ability to find the best solution.

In Table 5, for f_{11}, each comparison algorithm finds the global exact solution in the search space, but the worst value, mean value and standard deviation of the EAOA are worse

TABLE 3. Experimental effect for $f_1 - f_6$.

f	Result	ABC	ALO	CS	DA	MFO	SCA	WWO	AOA	EAOA
f_1	Best	220.0121	9.20E-06	0.000398	602.8512	0.000129	2.84E-05	9.87E-06	5.1E-206	**0**
	Worst	3389.404	0.000154	0.016830	3798.876	20000.00	1.018682	9.48E-05	5.10E-17	3.20E-106
	Mean	1153.976	4.59E-05	0.004598	1561.055	2333.340	0.088575	3.87E-05	1.70E-18	1.10E-107
	Std	750.7550	3.31E-05	0.004019	649.1196	5040.067	0.217231	2.16E-05	9.31E-18	5.90E-107
	rank	8	4	5	7	9	6	3	2	1
f_2	Best	4.732420	0.867857	0.033115	9.089201	0.000667	3.00E-08	0.398786	**0**	**0**
	Worst	18.27635	142.6297	0.306769	41.57023	90.00000	0.000880	22.75694	0	0
	Mean	9.722093	42.17231	0.109771	21.91219	35.34421	0.000107	6.179362	0	0
	Std	3.416250	51.13297	0.067843	8.184085	22.84584	0.000204	5.032948	0	0
	rank	4	8	3	6	7	2	5	1	1
f_3	Best	36587.97	809.8800	119.7235	3927.089	1158.726	27.65403	1347.831	6.3E-161	**0**
	Worst	85434.94	4654.660	450.7443	46885.80	53486.36	20671.14	13523.67	0.079546	0.009419
	Mean	62190.81	2713.907	271.0870	17248.66	23238.13	6000.664	5411.626	0.006052	0.000448
	Std	12073.91	1070.459	73.32985	9340.843	16092.86	5735.566	3417.582	0.015258	0.001846
	rank	8	4	3	7	9	6	5	2	1
f_4	Best	75.61132	7.138034	5.793716	18.77861	46.41657	9.236743	2.423767	1.63E-83	**9.3E-222**
	Worst	95.41573	24.27363	21.44915	39.73984	86.59631	54.37128	19.23032	0.052136	0.042324
	Mean	88.92891	15.99863	11.18181	28.01461	71.45640	26.62079	9.298808	0.031301	0.009969
	Std	4.458185	4.820272	3.756788	6.210137	10.14682	11.56245	4.059841	0.017494	0.016740
	rank	5	6	3	7	8	9	4	2	1
f_5	Best	21522.12	26.95564	25.09698	42538.43	25.39965	28.95335	**23.60544**	27.43900	27.07601
	Worst	1662457	1768.896	235.8464	2109648	90080.00	1487.826	1478.342	28.87363	28.55046
	Mean	545568.5	215.8098	73.91032	366485.8	15537.67	234.6454	124.8487	28.33206	27.96564
	Std	517743.7	354.5182	52.92944	425545.3	33911.45	373.3522	266.4782	0.371513	0.342728
	rank	9	5	3	8	7	6	4	2	1
f_6	Best	0.310918	0.06916	0.026958	0.111792	0.035435	0.005211	0.043817	2.87E-06	**8.85E-07**
	Worst	0.762719	0.415033	0.264290	1.379959	59.38535	0.365181	0.307821	0.000204	7.45E-05
	Mean	0.521649	0.194486	0.090986	0.420197	6.649045	0.064961	0.123428	5.85E-05	1.79E-05
	Std	0.119473	0.082226	0.046376	0.265179	12.56348	0.069812	0.059130	4.78E-05	1.71E-05
	rank	7	6	3	8	9	5	4	2	1

than those of other algorithms. For f_{12}, the optimal value, worst value, and mean value of the EAOA are consistent with those of the ABC, ALO, CS, MFO, WWO and AOA. The relative values of the EAOA are superior to those of the DA and SCA. The standard deviation of the EAOA is smaller than those of DA, SCA and AOA. For f_{14}, all algorithms find the global exact solutions except SCA. The optimal value, worst value, and mean value of ALO, CS, MFO and WWO are consistent, and the relative values are better than those of other algorithms. Compared to the basic AOA, the mean value and standard deviation of the EAOA have been slightly improved. For f_{13}, f_{15} and f_{16}, the AOA and EAOA all find the global exact solution, and their the optimal value, worst value, mean value and standard deviation are the same. The optimal value, worst value, mean value and standard deviation of the EAOA are better than those of other algorithms. Compared with other algorithms, the EAOA has a higher ranking and stronger stability to obtain the best solution. The Lévy variation increases the population diversity of the algorithm and expands the search range of the algorithm, which enhances the exploration ability and improves the calculation precision of the AOA. The differential sorting variation filters out the best individual from multiple candidate solutions and avoids premature convergence of the AOA, which enhances the exploitation ability and accelerates the convergence rate of the AOA. The EAOA realizes complementary advantages to avoid premature convergence. Therefore, the EAOA can effectively balance exploration and exploitation to find the global optimal solution in the whole search space.

TABLE 4. Experimental effect for $f_7 - f_{10}$.

f	Result	ABC	ALO	CS	DA	MFO	SCA	WWO	AOA	EAOA
f_7	Best	81.75113	53.72771	47.93984	96.85954	107.4551	0.000194	55.09535	**0**	**0**
	Worst	174.5590	180.0867	105.5339	262.4798	251.0838	108.0024	181.8466	0	0
	Mean	132.4386	89.11492	75.98921	187.4667	169.0073	20.55007	137.3140	0	0
	Std	19.29859	29.28948	15.85256	38.39968	33.73348	29.13905	27.03133	0	0
	rank	3	6	2	8	7	5	4	1	1
f_8	Best	11.31619	0.613028	0.227647	7.007097	2.222119	0.000868	3.378320	**8.88E-16**	**8.88E-16**
	Worst	19.39339	13.29975	6.746598	12.97522	19.96634	20.36802	6.709562	8.88E-16	8.88E-16
	Mean	16.65250	3.219216	2.939546	10.32817	17.98277	15.50860	4.871146	8.88E-16	8.88E-16
	Std	1.827110	2.850594	1.627025	1.312300	4.346526	8.210879	1.032643	0	0
	rank	5	6	4	3	7	8	2	1	1
f_9	Best	3.341748	0.001471	0.009670	5.554433	0.000169	0.000172	0.001770	0.003911	**0.000115**
	Worst	48.59607	0.053041	0.233692	40.23817	180.9112	1.058661	0.149953	0.418399	0.158719
	Mean	14.67498	0.018018	0.083563	15.72405	27.20445	0.395829	0.053383	0.159879	0.056077
	Std	10.29146	0.013503	0.060666	8.011583	58.83628	0.310316	0.042406	0.114938	0.049930
	rank	8	1	4	7	9	6	2	5	3
f_{10}	Best	4.060414	4.896494	0.432439	11.76364	**7.43E-05**	0.534932	2.581979	0.332927	0.215045
	Worst	1117819	27.77443	5.348732	16119.34	1549816	125.3027	8.918718	0.607500	0.426608
	Mean	139049.9	11.45503	2.790734	1391.420	51878.57	7.319279	5.997193	0.470503	0.320859
	Std	297862.5	5.067765	1.238739	3762.624	282917.7	22.76201	1.535881	0.059123	0.057604
	rank	9	5	3	7	8	6	4	2	1

The P-value Wilcoxon rank-sum test is used to detect whether two sets of data are the significant distinction between the EAOA and the other algorithms [30]. $P > 0.05$ in bold shows that there is no significant distinction between the two sets of data. $P \leq 0.05$ shows that there is a significant distinction between two sets of data. The results of the p-value Wilcoxon rank-sum test are described in Table 6. Most of the P-values are less than 0.05, which shows that there is a significant distinction between the EAOA and the other algorithms, and the data are real and valid, not obtained by chance.

The convergence graphs of these algorithms are presented in figure 2. The convergence curve intuitively demonstrates the convergence rate and calculation precision of different algorithms in solving the function optimization problem. The algorithm has a faster convergence rate and higher calculation precision, which shows that this algorithm has strong overall optimization performance and search ability to obtain the global optimal solution. For $f_1 - f_6$, the EAOA uses the exploration ability and exploitation ability to avoid falling into the local optimum and find the best solution. The optimal value, worst value, mean value and standard deviation of the EAOA are superior to those of other algorithms, as shown in Table 3. The relevant values of the EAOA have been greatly improved compared to the basic AOA, which shows that the EAOA has a strong search ability and optimization ability to find a faster convergence rate and higher calculation precision. For $f_7 - f_{10}$, the EAOA has a large search range and wide population diversity to filter out the best search agent and avoid premature convergence. The EAOA has strong global optimization and local optimization to obtain the global finest solution. Compared with other algorithms, most of the optimal value, worst value, mean value and standard deviation of the EAOA are better, as shown in Table 4. The convergence rate and calculation precision of the EAOA are better than those of other algorithms, which shows that the EAOA has excellent stability and superiority to solve the multimodal functions. For $f_{11} - f_{16}$, the EAOA combines the Lévy variation and the differential sorting variation to achieve complementary advantages and improve the overall search ability. The EAOA can obtain the exact optimal solution in the search space, which shows that the EAOA has certain stability and superiority to solve the fixed-dimension multimodal functions. Most of the optimal value, worst value, mean value and standard deviation of the EAOA are better than those of other algorithms, as shown in Table 5. The convergence rate and calculation precision of the EAOA are the best in all algorithms except f_{14}. The Lévy variation broadens the optimization space and increases population diversity to achieve the global search ability. The differential sorting variation filters out the optimal search agent and avoids the search stagnation to achieve the local search ability. The EAOA effectively adjusts exploration and exploitation to find a faster convergence rate and higher calculation precision.

TABLE 5. Experimental effect for $f_{11} - f_{16}$.

f	Result	ABC	ALO	CS	DA	MFO	SCA	WWO	AOA	EAOA
f_{11}	Best	**0.998004**	**0.998004**	**0.998004**	**0.998004**	**0.998004**	**0.998004**	**0.998004**	**0.998004**	**0.998004**
	Worst	0.998693	5.928845	0.998004	3.968250	10.76318	2.982105	1.992031	12.67051	12.67051
	Mean	0.998036	1.758254	0.998004	1.097012	3.755199	1.527981	1.031138	10.05618	7.389238
	Std	0.000134	1.060275	0	0.542290	2.870741	0.891864	0.181484	3.896102	4.584251
	rank	2	6	1	4	7	5	3	8	9
f_{12}	Best	**-1.03163**	**-1.03163**	**-1.03163**	**-1.03163**	**-1.03163**	**-1.03163**	**-1.03163**	**-1.03163**	**-1.03163**
	Worst	-1.03163	-1.03163	-1.03163	-1.03157	-1.03163	-1.03151	-1.03163	-1.03163	-1.03163
	Mean	-1.03163	-1.03163	-1.03163	-1.03162	-1.03163	-1.03159	-1.03163	-1.03163	-1.03163
	Std	4.46E-08	8.56E-14	6.71E-16	1.43E-05	6.78E-16	2.77E-05	5.5E-14	9.17E-08	5.96E-08
	rank	5	4	1	8	2	9	3	7	6
f_{13}	Best	-0.99998	**-1**	**-1**	**-1**	**-1**	**-1**	**-1**	**-1**	**-1**
	Worst	-0.96191	-0.93625	-1	-0.99954	-0.93625	-1	-1	-1	-1
	Mean	-0.98921	-0.96600	-1	-0.99998	-0.95750	-1	-1	-1	-1
	Std	0.01095	0.032350	1.03E-13	8.35E-05	0.030568	0	7.08E-09	0	0
	rank	5	7	2	4	6	1	3	1	1
f_{14}	Best	**-1**	**-1**	**-1**	**-1**	**-1**	-0.99998	**-1**	**-1**	**-1**
	Worst	-0.99961	-1	-1	-0.99995	-1	-0.99594	-1	-8.1E-05	-8.1E-05
	Mean	-0.99994	-1	-1	-0.99999	-1	-0.99899	-1	-0.70002	-0.90001
	Std	0.000101	5.32E-14	0	1.03E-05	0	0.000994	7.43E-17	0.466054	0.305104
	rank	5	3	1	4	1	6	2	8	7
f_{15}	Best	-0.99511	**-1**	**-1**	**-1**	**-1**	**-1**	**-1**	**-1**	**-1**
	Worst	-0.97858	-0.99028	-0.99028	-0.99028	-0.96278	-0.99028	-0.99028	-1	-1
	Mean	-0.98993	-0.99709	-0.99777	-0.99417	-0.98694	-0.99968	-0.99935	-1	-1
	Std	0.002341	0.004529	0.004065	0.004841	0.009801	0.001774	0.002465	0	0
	rank	3	6	5	7	8	2	4	1	1
f_{16}	Best	0.192169	0.003318	0.046627	0.074677	3.70E-19	3.19E-21	1.81E-10	**0**	**0**
	Worst	1.145616	3.721474	0.510922	7.913816	8.10E-14	0.001402	3.997277	0	0
	Mean	0.660364	0.495535	0.215764	2.801177	1.69E-14	5.14E-05	0.272314	0	0
	Std	0.272572	0.983010	0.102266	1.948636	2.00E-14	0.000256	0.763552	0	0
	rank	5	7	4	8	2	3	6	1	1

The ANOVA test of these algorithms is presented in figure 3. The standard deviation can intuitively reflect the stability of each comparison algorithm in solving the function optimization problem. The algorithm has a smaller standard deviation, which shows that the algorithm has strong overall optimization ability and stability. The ranking is based on the standard deviation. The Lévy variation and the differential sorting variation enhance the exploration ability and the exploitation ability of the AOA to improve the convergence rate and calculation precision. For $f_1 - f_6$, the standard deviation of the EAOA is superior to those of other algorithms, and the ranking of the EAOA is first, which shows that the EAOA not only has a relatively small standard deviation, but also has strong stability and superiority. The EAOA has strong search ability and practicability to solve the unimodal functions. For $f_7 - f_{10}$, the EAOA utilizes two additional strategies to expand the population space and avoid dropping into the local optimal solution, which is beneficial to enhance the global search ability and the local search ability. Compared with other algorithms, the standard deviation of the EAOA is better, which shows that the EAOA has a relatively small standard deviation and strong stability. For f_{13}, f_{15} and f_{16}, the EAOA has strong overall search ability and superiority to find the exact solution in the search space. The standard deviation of the EAOA is better than those of other algorithms. The EAOA has a high ranking and relatively small standard deviation, which shows that the EAOA has strong stability. For f_{11}, f_{12} and f_{14}, the standard deviation of the EAOA is relatively

TABLE 6. Results of the p-value Wilcoxon rank-sum test.

f	ABC	ALO	CS	DA	MFO	SCA	WWO	AOA
f_1	2.6286E-11	2.6286E-11	2.6286E-11	2.6286E-11	2.6286E-11	2.6286E-11	2.6286E-11	1.7225E-10
f_2	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	N/A
f_3	3.0010E-11	3.0010E-11	3.0010E-11	3.0010E-11	3.0010E-11	3.0010E-11	3.0010E-11	9.7929E-08
f_4	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	1.1937E-06
f_5	3.0199E-11	1.0666E-07	7.0430E-07	3.0199E-11	5.5727E-10	3.0200E-11	1.1738E-03	3.5638E-04
f_6	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	3.5923E-05
f_7	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	N/A
f_8	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	N/A
f_9	3.0199E-11	4.8560E-03	3.9167E-02	3.0199E-11	**8.0730E-01**	1.2493E-05	1.3124E-07	1.0407E-04
f_{10}	3.0199E-11	3.0199E-11	3.0199E-11	3.0199E-11	**7.7272E-02**	3.0199E-11	3.0199E-11	4.6159E-10
f_{11}	2.6806E-04	3.8270E-07	1.2118E-12	3.8307E-05	1.5270E-04	7.6973E-04	9.4716E-11	5.0842E-03
f_{12}	2.6015E-08	3.0199E-11	1.7203E-12	7.6973E-04	1.2118E-12	3.0199E-11	9.3611E-12	7.6180E-03
f_{13}	1.2118E-12	1.1980E-12	3.1349E-04	3.4526E-07	8.7250E-08	N/A	1.3056E-07	N/A
f_{14}	9.7917E-05	3.0199E-11	1.2118E-12	2.0523E-03	1.2118E-12	1.0666E-07	2.3657E-12	2.2658E-03
f_{15}	1.2118E-12	1.2108E-12	1.2118E-12	2.6487E-08	4.8687E-13	**8.1523E-02**	1.3041E-07	N/A
f_{16}	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	1.2118E-12	N/A

stable compared to the other algorithms. The Lévy variation increases the population diversity and expands the search range to enhance the global search ability. The differential sorting variation filters out the best individual from multiple candidate solutions and avoids premature convergence to enhance the local search ability. The EAOA realizes complementary advantages to obtain a faster convergence rate and higher calculation precision. The EAOA has strong stability and superiority to solve the function optimization problem.

To verify the robustness of the EAOA, the optimal value, worst value, mean value, standard deviation and *P*-value Wilcoxon rank-sum test are used as some evaluation indicators. The robustness of the EAOA is mainly reflected in the following aspects. First, the EAOA balances exploration and exploitation to obtain a faster convergence rate and higher calculation precision. Second, the EAOA has a relatively small standard deviation, which shows that the algorithm has strong overall optimization ability and stability. Third, if the EAOA has a large standard deviation, which will not cause catastrophic and combinatorial explosions.

C. EAOA FOR SOLVING PROJECT OPTIMIZATION

To corroborate the practicability and availability, the EAOA is used to resolve the project optimization problems, such as the welded beam project [31], tension/compression spring project [32], pressure vessel project [33], cantilever beam project [34], and speed reducer project [35].

1) WELDED BEAM PROJECT

The objective is to consume less creation cost to complete the design project. As presented in figure 4, a few crucial constraint variables are as follows: shear stress (τ), beam bending stress (σ), beam end deflection (δ), bar buckling load (P_c), and boundary constraints. There are four optimization variables: weld thickness (h), clamped bar length (l), bar length (t), and bar thickness (b). The formula is as follows:
Consider

$$x = [x_1 \quad x_2 \quad x_3 \quad x_4] = [h \quad l \quad t \quad b] \tag{12}$$

Minimiz

$$f(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2) \tag{13}$$

Subject to

$$g_1(x) = \tau(x) - \tau_{max} \leq 0 \tag{14}$$
$$g_2(x) = \sigma(x) - \sigma_{max} \leq 0 \tag{15}$$
$$g_3(x) = \delta(x) - \delta_{max} \leq 0 \tag{16}$$
$$g_4(x) = x_3 - x_4 \leq 0 \tag{17}$$
$$g_5(x) = P - P_c(x) \leq 0 \tag{18}$$
$$g_6(x) = 0.125 - x_1 \leq 0 \tag{19}$$
$$g_7(x) = 1.1047x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0 \tag{20}$$

Variable range

$$0.1 \leq x_1 \leq 2, \quad 0.1 \leq x_2 \leq 10, \quad 0.1 \leq x_3 \leq 10, \quad 0.1 \leq x_4 \leq 2 \tag{21}$$

where

$$\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2} \tag{22}$$

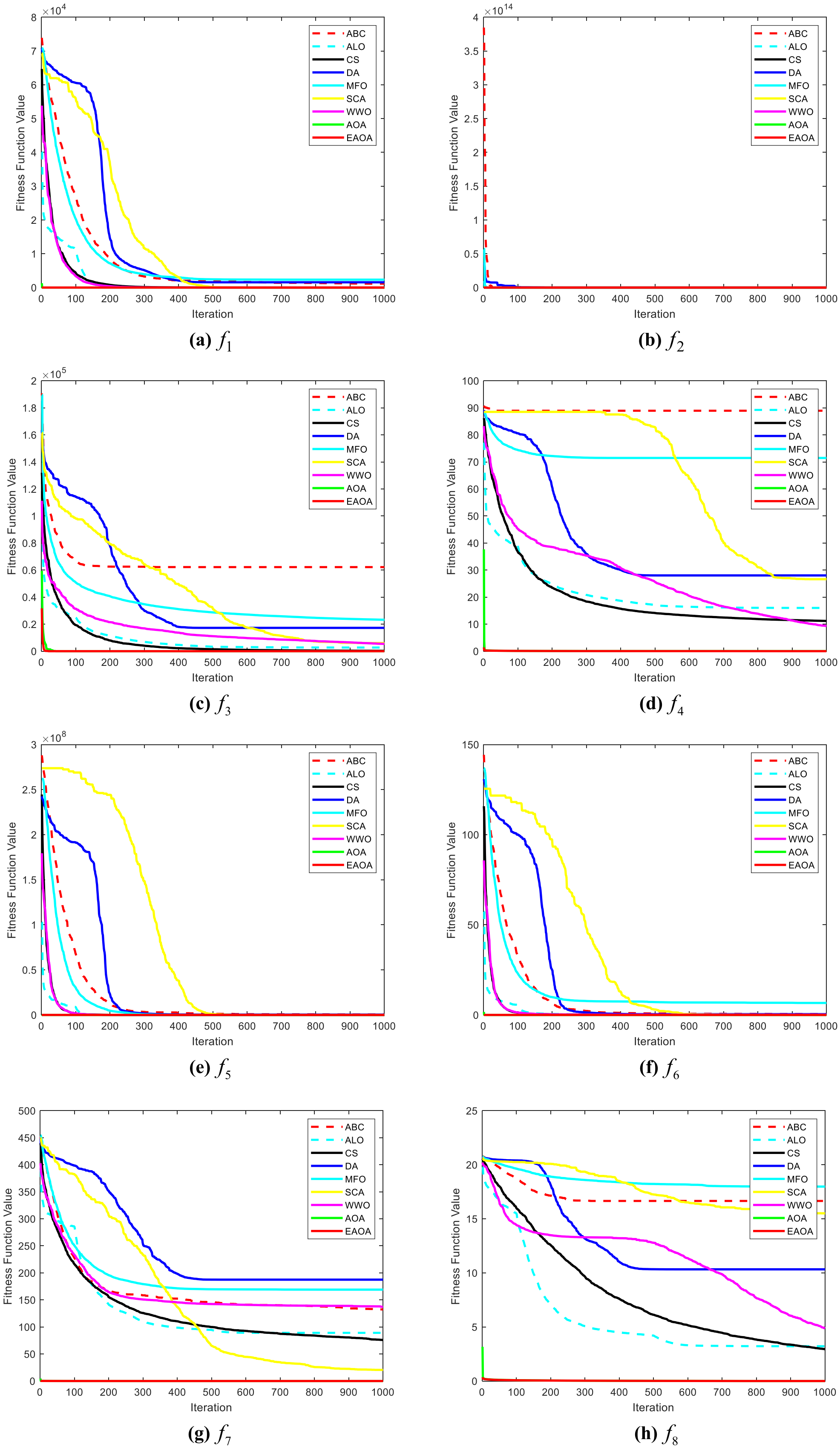


FIGURE 2. Convergence graphs of these algorithms.

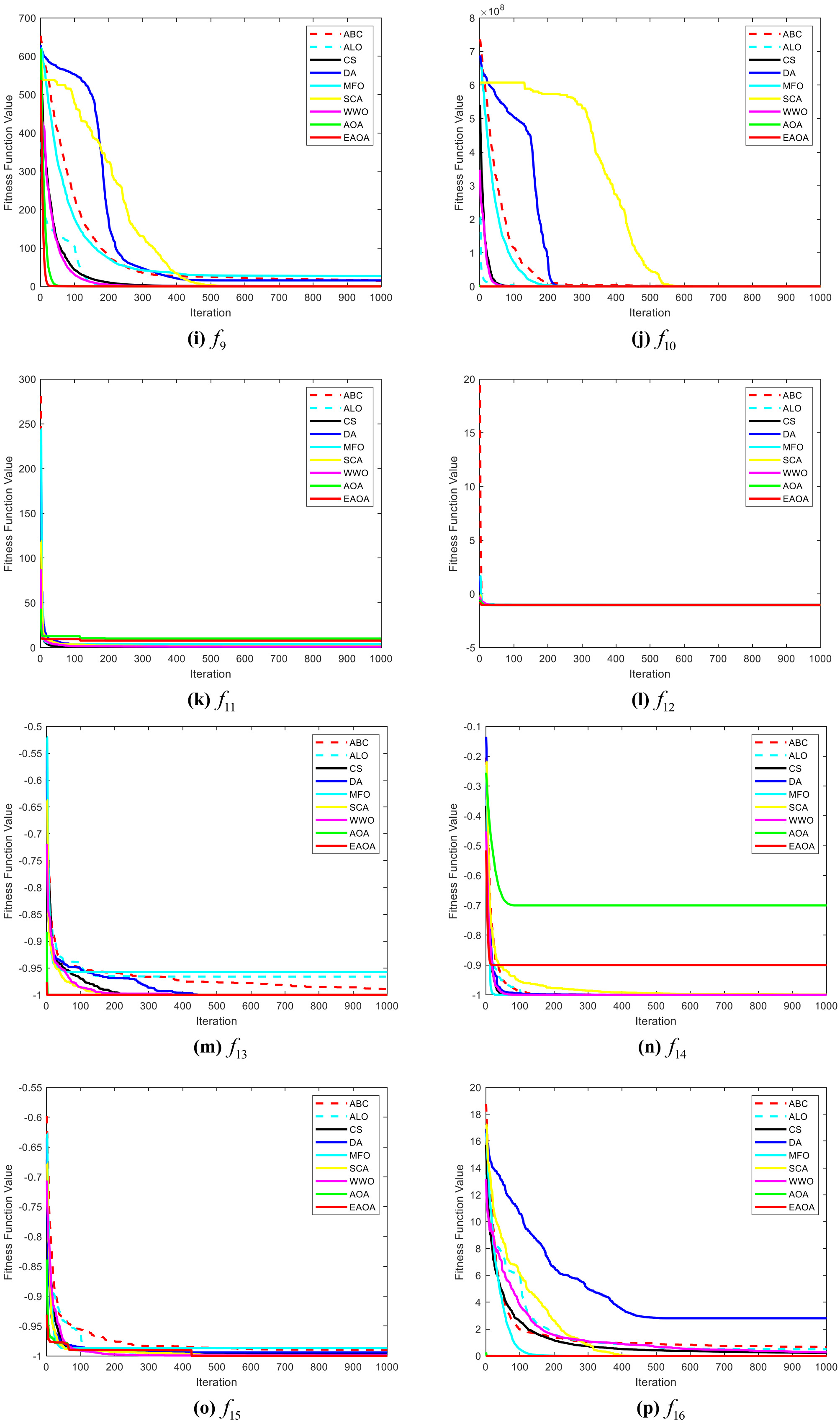


FIGURE 2. ***(Continued.)*** **Convergence graphs of these algorithms.**

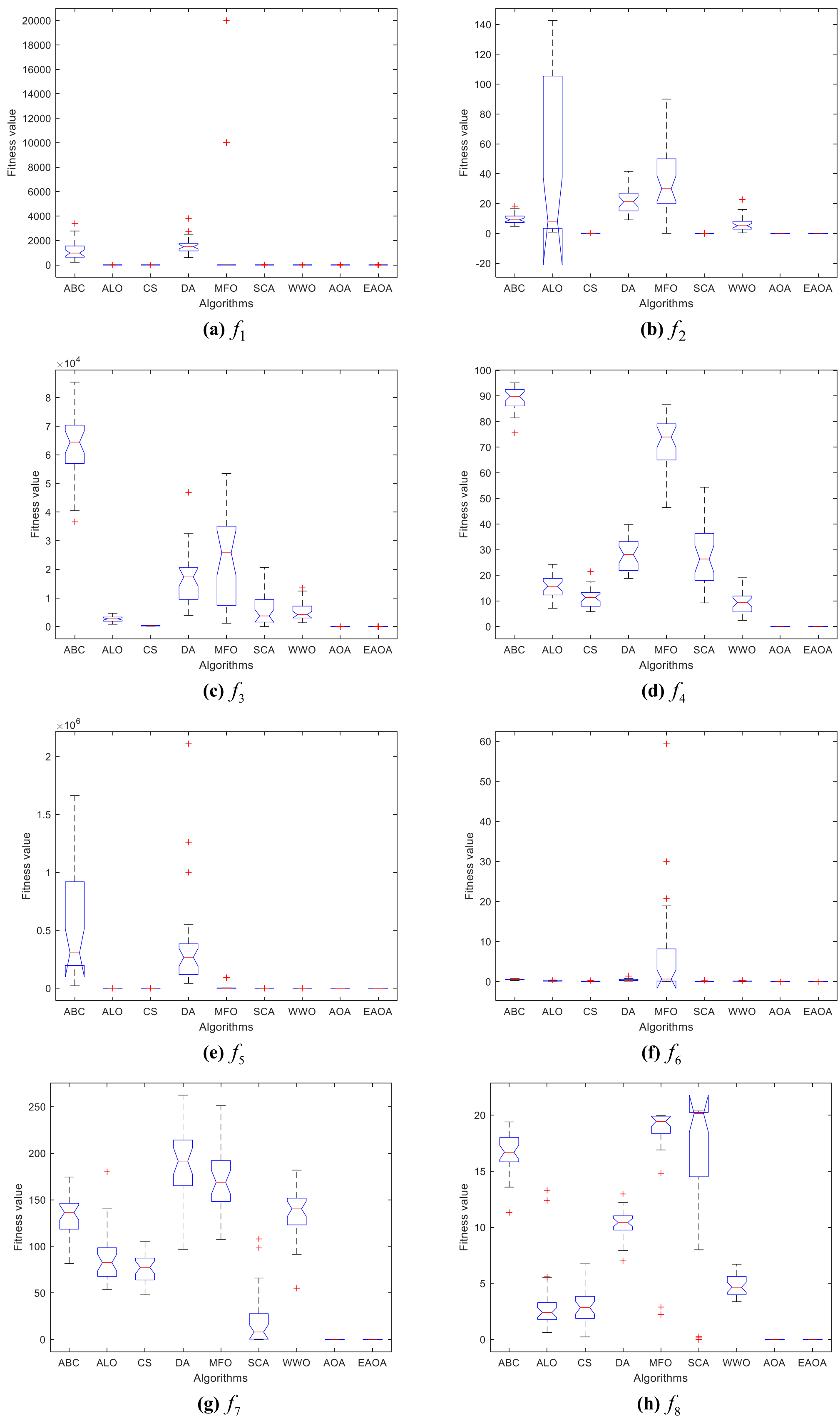
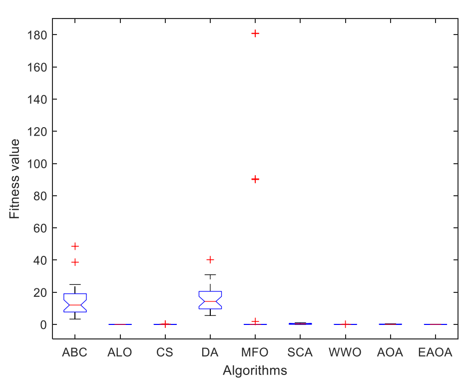
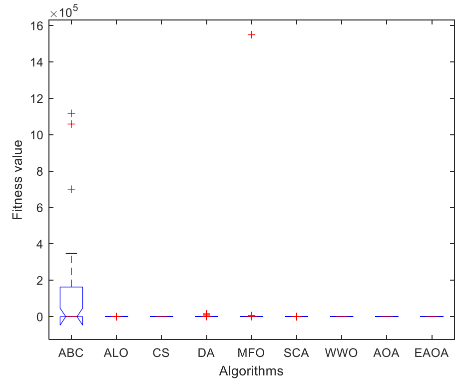


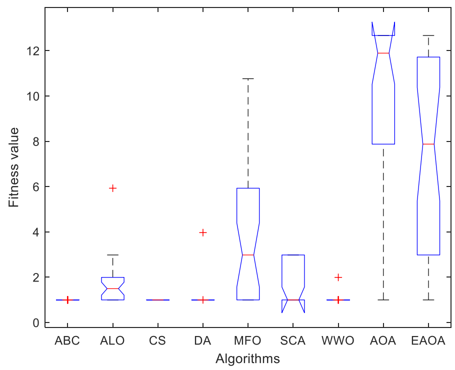
FIGURE 3. ANOVA tests of these algorithms.



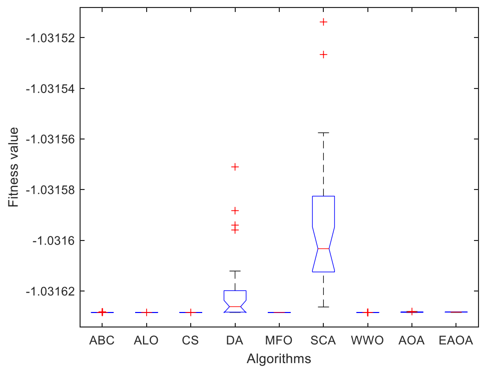
(i) f_9



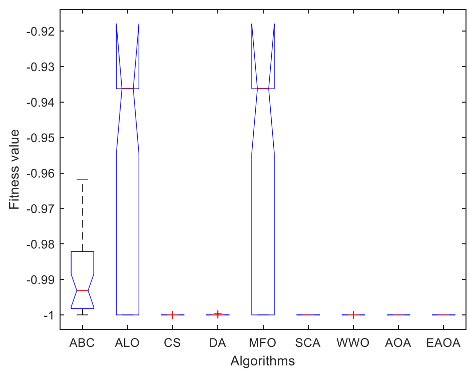
(j) f_{10}



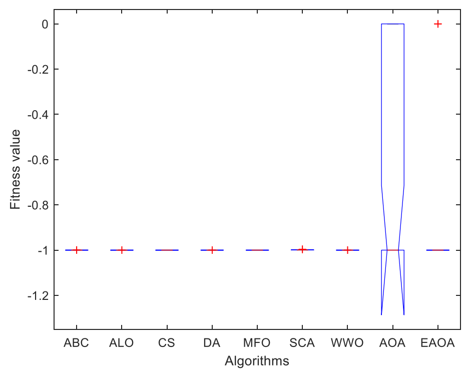
(k) f_{11}



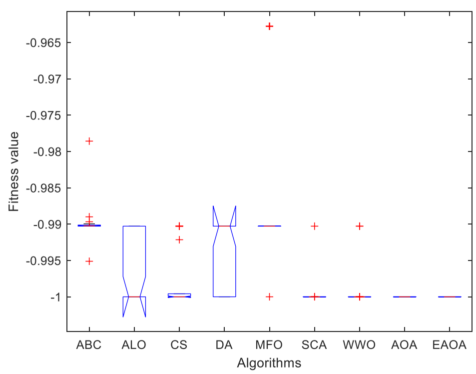
(l) f_{12}



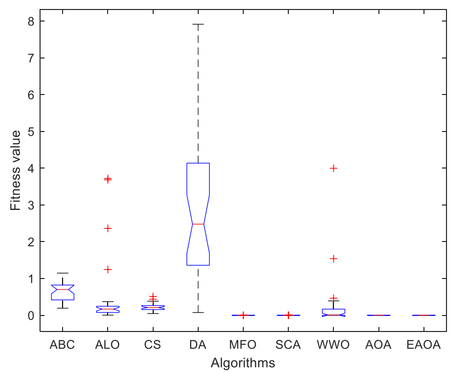
(m) f_{13}



(n) f_{14}



(o) f_{15}



(p) f_{16}

FIGURE 3. ***(Continued.)*** **ANOVA tests of these algorithms.**

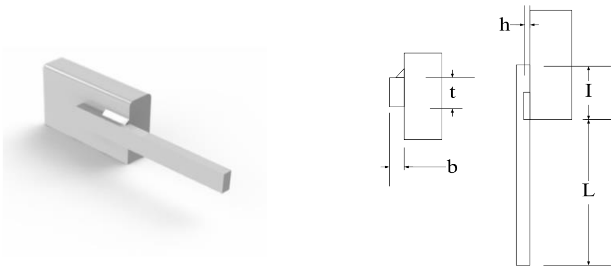


FIGURE 4. Welded beam project.

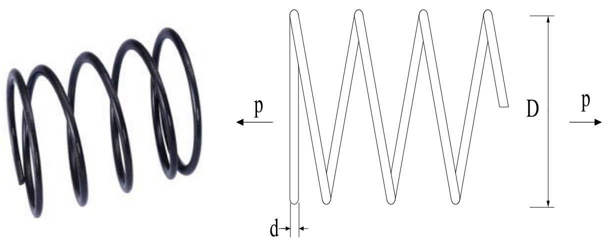


FIGURE 5. Tension/ compression spring project.

$$\tau' = \frac{P}{\sqrt{2}\,x_x x_2}, \quad \tau'' = \frac{MP}{J}, \; M = P(L + \frac{X_2}{2}) \tag{23}$$

$$R = \sqrt{\frac{x_2^2}{4} + (\frac{x_1 + x_3}{2})^2} \tag{24}$$

$$J = 2\left\{\sqrt{2}\,x_1 x_2 \left[\frac{x_2^2}{4} + (\frac{x_1 + x_3}{2})^2\right]\right\} \tag{25}$$

$$\sigma(x) = \frac{6PL}{x_4 x_3^2}, \; \delta(x) = \frac{6PL}{E x_3^2 x_4} \tag{26}$$

$$P_c(x) = \frac{4.103E\sqrt{\frac{x_3^2 x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right) \tag{27}$$

The optimization effect is described in Table 7. The EAOA can obtain a relatively small manufacturing cost in addressing the welded beam design. The control variables and objective fitness solution of the EAOA are better, which shows that the EAOA has better superiority.

2) TENSION/ COMPRESSION SPRING PROJECT

The objective is to achieve the minimum weight of a tension/compression spring. As presented in fig.5, a few constraint variables are as follows: smallest deflection (g_1), shear stress (g_2), vibration amplitude (g_3), and the external diameter (g_4). There are three decision variables: the spring diameter (d), average coil diameter (D), and the number of coils (N). The formula is as follows:
Consider

$$x = [x_1 \quad x_2 \quad x_3] = [d \quad D \quad N] \tag{28}$$

Minimize

$$f(x) = (x_3 + 2)x_2 x_1^2 \tag{29}$$

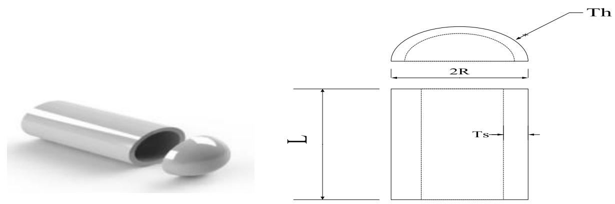


FIGURE 6. Pressure vessel project.

Subject to

$$g_1(x) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \le 0 \tag{30}$$

$$g_2(x) = \frac{4x_2^2 - x_1 x_2}{12566(x_2 x_1^3 - x_1^4)} + \frac{1}{5108 x_1^2} \le 0 \tag{31}$$

$$g_3(x) = 1 - \frac{140.45 x_1}{x_2^2 x_3} \le 0 \tag{32}$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \le 0 \tag{33}$$

Variable range

$$0.05 \le x_1 \le 2.00, \quad 0.25 \le x_2 \le 1.30, \quad 2.00 \le x_3 \le 15.0 \tag{34}$$

The optimization effect is described in Table 8. The optimal cost of the EAOA is the smallest in all algorithms, and the decision variables of the EAOA are superior to those of other algorithms, which shows that the EAOA has a strong global search ability to acquire a higher convergence precision.

3) PRESSURE VESSEL PROJECT

The objective is to optimize the minimum total cost. As presented in fig. 6, a few variables are as follows: the pressure pipe thickness (T_s), the pressure cap thickness (T_h), inside radius (R), cylinder length (L). The formula is as follows:
Consider

$$x = [x_1 \quad x_2 \quad x_3 \quad x_4] = [T_s \quad T_h \quad R \quad L] \tag{35}$$

Minimize

$$f(x) = 0.6224 x_1 x_3 x_4 + 1.7781 x_2 x_3^2 + 3.1661 x_1^2 x_4 + 19.8 x_1^2 x_3 \tag{36}$$

Subject to

$$g_1(x) = -x_1 + 0.0193 x_3 \le 0 \tag{37}$$

$$g_2(x) = -x_3 + 0.00954 x_3 \le 0 \tag{38}$$

$$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \le 0 \tag{39}$$

$$g_4(x) = x_4 - 240 \le 0 \tag{40}$$

Variable range

$$0 \le x_1 \le 99, \quad 0 \le x_2 \le 99, \quad 10 \le x_3 \le 200, \quad 10 \le x_4 \le 200 \tag{41}$$

TABLE 7. Optimization effect of the welded beam project.

Algorithm	h	l	t	b	Effect
GWO [36]	0.2056760	3.4783770	9.0368100	0.2057780	1.72624000
GSA [36]	0.1821290	3.8569790	10.000000	0.2023760	1.87995000
CPSO [36]	0.2023690	3.5442140	9.0482100	0.2057230	1.72802000
GA (Coello) [37]	N/A	N/A	N/A	N/A	1.82450000
GA (Deb) [38]	N/A	N/A	N/A	N/A	2.38000000
GA (Deb) [39]	0.2489000	6.1730000	8.1789000	0.2533000	2.43310000
HS (Lee and Geem) [40]	0.2442000	6.2231000	8.2915000	0.2443000	2.38070000
Random [41]	0.4575000	4.7313000	5.0853000	0.6600000	4.11850000
Simplex [41]	0.2792000	5.6256000	7.7512000	0.2796000	2.53070000
David [41]	0.2434000	6.2552000	8.2915000	0.2444000	2.38410000
Approx [41]	0.2444000	6.2189000	8.2915000	0.2444000	2.38150000
BA [42]	0.2015000	3.5620000	9.0414000	0.2057000	1.73120650
CDE [43]	0.2031700	3.5429980	9.0334980	0.2061790	1.73346200
WCA [44]	0.2057280	3.4705220	9.0366200	0.2057290	1.72485600
MBA [45]	0.2057290	3.4704930	9.0366260	0.2057290	1.72485300
IACO [46]	0.2057000	3.4711310	9.0366830	0.2057310	1.72491800
RO [47]	0.2036870	3.5284670	9.0042630	0.2072410	1.73534400
SaC [48]	0.2444380	6.2379670	8.2885760	0.2445660	2.38543470
PSO-DE [49]	N/A	N/A	N/A	N/A	1.72485310
WSA [50]	0.2057296	3.4704899	9.0366239	0.2057296	1.72485254
EOMSA [32]	0.2242500	3.2486000	8.6518000	0.2244500	1.72460000
EAOA	0.2001900	3.3614000	9.0507000	0.2065200	1.71000000

TABLE 8. Optimization effect of the tension/ compression spring project.

Algorithm	d	D	N	Effect
RO [47]	0.0513700000	0.3490960000	11.762790000	0.0126788000
SaC [48]	0.0521602000	0.3681586950	10.648442200	0.0126692490
BGRA [51]	0.0516747000	0.3563726000	1.3092290000	0.0126652370
IHS [52]	0.0511543000	0.3498711000	12.076432100	0.0126706000
NM-PSO [53]	0.0516200000	0.3554980000	11.3333272000	0.0126706000
CPSO [54]	0.0517280000	0.3576440000	11.2445400000	0.0126747000
YYPO [55]	0.0517050000	0.3571000000	11.2660000000	0.1266500000
GA (Coello) [37]	0.0514800000	0.3516610000	11.6322010000	0.0127048000
ES [56]	0.0516430000	0.3553600000	11.3979260000	0.0126980000
UPSO [54]	N/A	N/A	N/A	0.0131200000
CDE [43]	0.0516090000	0.3547140000	11.4108310000	0.0126702000
ABC [57]	0.0517490000	0.3581790000	11.2037630000	0.0126650000
MFO [5]	0.0519944570	0.3641093200	10.8684218620	0.0126669000
GWO [36]	0.0516900000	0.3567370000	11.2888500000	0.0126660000
AFA [58]	0.0516674837	0.3561976945	11.3195613646	0.0126653049
BA [42]	0.0516900000	0.3567300000	11.2885000000	0.0126700000
LSA-SM [59]	0.0517045300	0.3570899000	11.2671800000	0.0126652400
EAOA	0.0511879200	0.3604960000	11.8610100000	0.0124646000

The optimization effect is described in Table 9. The design variables and optimal effect of the EAOA are better compared to other algorithms. The EAOA has a strong overall optimization ability. The EAOA utilizes exploration and exploitation to strengthen the calculation efficiency and precision, which shows that the EAOA has good robustness and global optimization.

4) CANTILEVEL BEAM PROJECT

The objective is to reduce the weight of the cantilever beam. The formula is as follows:

Consider

$$x = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5] \tag{42}$$

Minimize

$$f(x) = 0.6224(x_1 + x_2 + x_3 + x_4 + x_5) \tag{43}$$

Subject to

$$g(x) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} \leq 1 \tag{44}$$

Variable range

$$0.01 \leq x_1, x_2, x_3, x_4, x_5 \leq 100 \tag{45}$$

TABLE 9. Optimization effect of the pressure vessel project.

Algorithm	T_s	T_h	R	L	Effect
GSA [60]	1.1250000	0.6250000	55.98865980	84.45420250	8538.8359000
GA [61]	0.9375000	0.5000000	48.32900000	112.6790000	6410.3811000
WEO [39]	0.8125000	0.4375000	42.09840000	176.6366000	6059.7143000
PSO-GA [62]	0.7781686	0.3846491	40.31961870	200.0000000	5885.3327736
Lagrangian Multiplier [63]	1.1250000	0.6250000	58.29100000	43.69000000	7198.0428000
Branch-bound [64]	1.1250000	0.6250000	47.70000000	117.7010000	8129.1036000
EOMSA [32]	1.1460200	0.5664770	59.37910000	37.82830000	5879.7727000
GA (Coello) [37]	0.8125000	0.4375000	40.32390000	200.0000000	6228.7445000
CPSO [54]	0.8125000	0.4375000	42.09126600	176.7465000	6061.0777000
CDE [43]	0.8125000	0.4375000	42.09841100	176.7465000	6059.7340000
ABC [57]	0.8125000	0.4375000	42.09844600	176.6365960	6059.7143390
BA [42]	0.8125000	0.4375000	42.09844560	176.6365958	6059.7143348
AFA [58]	0.8125000	0.4375000	42.09844611	176.6365894	6059.7142719
ACO [65]	0.8125000	0.4375000	42.10362400	176.5726560	6059.0888000
ES [56]	0.8125000	0.4375000	42.09808700	176.6405180	6059.7456000
MFO [5]	0.8125000	0.4375000	42.09844500	176.6365960	6059.7143000
TLBO [66]	N/A	N/A	N/A	N/A	6059.7143350
LSA-SM [59]	0.8103764	0.4005695	41.98842000	178.0048000	5942.6966000
HHO [67]	0.8175838	0.4072927	42.09174576	176.7196352	6000.4625900
BIANCA [68]	0.8125000	0.4375000	42.09680000	176.6580000	6059.9384000
MDDE [69]	0.8125000	0.4375000	42.09684460	176.6360470	6059.7016600
EAOA	0.8155682	0.4094455	42.08483000	176.8443000	5879.6838000

TABLE 10. Optimization effect of the cantilever beam project.

Algorithm	x_1	x_2	x_3	x_4	x_5	Effect
MMA [70]	6.01000	5.30000	4.49000	3.49000	2.15000	1.3400000
GCA_I [70]	6.01000	5.30000	4.49000	3.49000	2.15000	1.3400000
GCA_II [70]	6.01000	5.30000	4.49000	3.49000	2.15000	1.3400000
CS [3]	6.00890	5.30490	4.50230	3.50770	2.15040	1.3399900
SOS [71]	6.01878	5.30344	4.49587	3.49896	2.15564	1.3399600
MVO [71]	6.02394	5.30601	4.49501	3.49602	2.15273	1.3399595
ELPSO [72]	6.01600	5.30920	4.49430	3.50150	2.15270	1.3400000
EAOA	6.01530	5.29690	4.49400	3.50290	2.15510	1.3399593

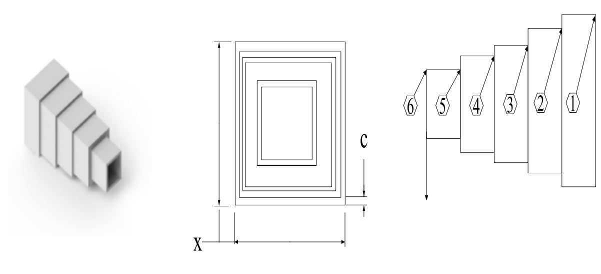


FIGURE 7. Cantilever beam project.

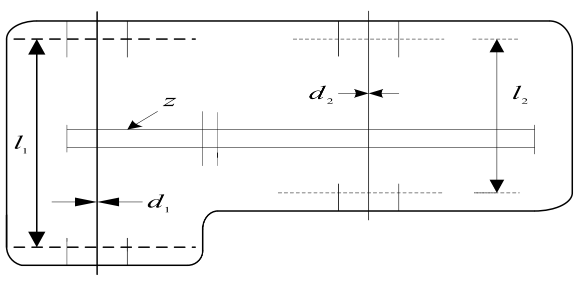


FIGURE 8. Speed reducer project.

The optimization effect is described in Table 10. The design variables and optimal cost of the EAOA are better than those of other algorithms, which shows that the EAOA expands the search space and avoids premature convergence so that the EAOA has strong stability and robustness to achieve the global optimal solution.

5) SPEED REDUCER PROJECT

The objective is to minimize the weight of the speed reducer. As presented in fig. 8, the design variables are as follows: the breadth (b), the number of teeth (m), the number of pinion teeth (z), the first bearing length (l_1), the second bearing length (l_2), first shaft bearing (d_1), the second bearing diameter (d_2). The formula is as follows:

Consider

$$x = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7] = [b \quad m\, z \quad l_1 \quad l_2 \quad d_1 \quad d_2] \tag{46}$$

TABLE 11. Optimization effect of the speed reducer project.

Algorithm	b	m	z	l_1	l_2	d_1	d_2	Effect
ABC [57]	3.500000	0.7000	17.00000	7.300000	7.7153190000	3.35021400	5.28665400	2994.471066
CS [3]	3.500000	0.7000	17.00000	7.300000	7.7153190000	3.35021400	5.28665400	2994.471066
HCPS [73]	3.500022	0.7000	17.000012	7.300427	7.7153770000	3.35023000	5.28666300	2994.499107
SCA [48]	0.350001	0.7000	17.00000	7.300156	7.8000270000	3.35022100	5.28668500	2996.356689
LMFO [74]	3.500000	0.7000	17.00000	7.300000	7.8000000000	3.35021400	5.28668320	2996.348167
MBA [45]	3.500000	0.7000	17.00000	7.300033	7.7157720000	3.35021800	5.28665400	2994.482453
MBFPA [75]	3.500000	0.7000	17.00000	7.300000	7.7153199122	3.35021466	5.28665446	2994.341315
WCA [49]	3.500000	0.7000	17.00000	7.300000	7.7153190000	3.35021400	5.28665400	2994.471066
PSODE [49]	3.500000	0.7000	17.00000	7.300000	7.8000000000	3.35021400	5.28668320	2996.348167
MDE [76]	3.500010	0.7000	17.00000	7.300156	7.8000270000	3.35022100	5.28668500	2996.356689
HEAA [77]	3.500022	0.7000	17.000012	7.300427	7.7153770000	3.35023000	5.28666300	2994.499107
PVS [78]	3.499990	0.6999	17..0000	7.300000	7.8000000000	3.35020000	5.28660000	2996.348100
EWOA [79]	3.510630	0.7000	17.00000	7.300000	7.8000000000	3.35908000	5.28998000	2993.765800
EAOA	3.501720	0.7000	17.00000	7.300000	7.3000000000	3.35607000	5.28792000	2988.197200

Minimize

$$f(x) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2) \quad (47)$$

Subject to

$$g_1(x) = -x_1 + 0.0193x_3 \le 0 \quad (48)$$

$$g_2(x) = \frac{397.5}{x_1x_2^2x_3} - 1 \le 0 \quad (49)$$

$$g_3(x) = \frac{1.93x_4^3}{x_2x_6^4x_3} - 1 \le 0 \quad (50)$$

$$g_4(x) = \frac{1.93x_5^3}{x_2x_7^5x_3} - 1 \le 0 \quad (51)$$

$$g_5(x) = \frac{[(745x_4/x_2x_3)^2 + 16.9 \times 10^6]^{1/2}}{110x_6^3} - 1 \le 0 \quad (52)$$

$$g_6(x) = \frac{[(745x_5/x_2x_3)^2 + 157.5 \times 10^6]^{1/2}}{85x_7^3} - 1 \le 0 \quad (53)$$

$$g_7(x) = \frac{x_2x_3}{40} - 1 \le 0 \quad (54)$$

$$g_8(x) = \frac{5x_2}{x_1} - 1 \le 0 \quad (55)$$

$$g_9(x) = \frac{x_1}{12x_2} - 1 \le 0 \quad (56)$$

$$g_{10}(x) = \frac{1.5x_6 + 1.9}{x_4} - 1 \le 0 \quad (57)$$

$$g_{11}(x) = \frac{1.1x_7 + 1.7}{x_5} - 1 \le 0 \quad (58)$$

$$2.6 \le x_1 \le 3.6,\ 0.7 \le x_2 \le 0.8,\ 17 \le x_3 \le 28 \quad (59)$$

$$7.3 \le x_4 \le 8.3,\ 7.3 \le x_5 \le 8.3,\ 2.9 \le x_6 \le 3.9 \quad (60)$$

$$5.0 \le x_7 \le 5.5 \quad (61)$$

The optimization results are described in Table 11. The EAOA utilizes the Lévy variation and the differential sorting variation to perform global optimization. The optimal variables and optimal cost of the EAOA are the best in all algorithms, which shows that the EAOA has strong stability and feasibility to achieve a better optimal value.

Statistically, the AOA is based on the distribution character of the dominant arithmetic operators to imitate addition (A), subtraction (S), multiplication (M) and division (D) to find the global optimal solution in the whole search space. The EAOA effectively solves the function optimization and the project optimization for the following reasons. First, The EAOA has the characteristics of a simple algorithm framework, better control parameters, less computational cost, stronger stability and easy implementation. Second, the Lévy variation increases population diversity, broadens the optimization space and enhances the global search ability. The differential sorting variation filters out the optimal search agent, avoids the search stagnation and enhances the local search ability. The two optimization strategies can achieve complementary advantages to avoid falling into the local optimum and obtain the best solution. Third, the control parameter r_1 can regulate exploration and exploitation to enhance the overall optimization performance of EAOA. If $r_1 < MOA$, the EAOA uses multiplication (M) and division (D) to perform the exploration phase and find the position of the optimal search agent, which is beneficial to avoid premature convergence and accelerate the convergence rate. If $r_1 \geq MOA$, the EAOA uses addition (A) and subtraction (S) to perform the exploitation phase and enhance the local search ability, which is beneficial to avoid search stagnation and improve the calculation precision. To summarize, the EAOA effectively uses exploration and exploitation to obtain a faster convergence rate, higher calculation precision and stronger stability.

V. CONCLUSION AND FUTURE RESEARCH

In this paper, an enhanced AOA based on the Lévy variation and the differential sorting variation is proposed to solve the function optimization and the project optimization. The purpose of the algorithm optimization is to obtain the best solution of the benchmark function and the minimum consumption cost of the engineering design project. The Lévy variation increases the population diversity of the algorithm and expands the search range of the algorithm, which enhances the exploration ability and improves the calculation precision of the AOA. The differential sorting variation filters out the best individual from multiple candidate solutions and avoids premature convergence of the algorithm, which enhances the exploitation ability and accelerates the convergence rate of the AOA. Therefore, the EAOA can optionally switch between the exploration ability and the exploitation ability to find the global optimal solution in the search space. For function optimization, the EAOA adopts the obvious advantages of two variations to improve the overall optimization performance of the AOA. The EAOA has a strong global search ability and local search ability to avoid the search stagnation of the algorithm. The convergence rate and the calculation precision of the EAOA are better than those of other algorithms. The EAOA has a relatively small standard deviation, which indicates that the EAOA has strong stability. For project optimization, compared with other algorithms, the EAOA has a strong overall search ability to obtain better control parameters and a smaller consumption cost. The experimental results show that the EAOA has a faster convergence rate, higher calculation precision and stronger stability. Meanwhile, the EAOA is an effective and feasible algorithm to solve the optimization problem.

In future research, introducing effective search strategies, adopting unique coding methods (complex-valued encoding, quantum coding, or discrete coding), or combining with other swarm intelligence algorithms will achieve complementary advantages and improve the overall optimization ability. The enhanced AOA will accelerate the convergence rate and improve calculation precision. The enhanced AOA will be used to solve the coordinated path planning of multiple unmanned underwater vehicles, the coordinated path planning of unmanned combat aerial vehicles and unmanned underwater vehicles in underwater target strike missions, and the optimal path planning of an unmanned underwater vehicle undersea terrain matching navigation and the dynamic obstacle avoidance of unmanned underwater vehicles. The purpose of optimization is to effectively avoid all threat areas and find the shortest and safest path with minimal threat cost and fuel cost.

ACKNOWLEDGMENT

The authors would like to thank everyone involved for their contribution to this article. They also would like to thank the editor and anonymous reviewers for the helpful comments and suggestions.

REFERENCES

[1] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optim.*, vol. 39, pp. 459–471, Nov. 2007.

[2] S. Mirjalili, "The ant lion optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, May 2015.

[3] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems," *Eng. Comput.*, vol. 29, no. 1, pp. 17–35, 2011.

[4] S. Mirjalili, "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2015.

[5] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowl.-Based Syst.*, vol. 89, pp. 228–249, Nov. 2015.

[6] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowl.-Based Syst.*, vol. 96, pp. 120–133, Mar. 2016.

[7] Y.-J. Zheng, "Water wave optimization: A new nature-inspired metaheuristic," *Comput. Oper. Res.*, vol. 55, no. 1, pp. 1–11, Mar. 2015.

[8] B. Alatas and H. Bingol, "A physics based novel approach for travelling tournament problem: Optics inspired optimization," *Inf. Technol. Control*, vol. 48, no. 3, pp. 373–388, Sep. 2019.

[9] B. Alatas and H. Bingol, "Comparative assessment of light-based intelligent search and optimization algorithms," *Light Eng.*, vol. 28, no. 6, pp. 51–59, 2020.

[10] S. Akyol and B. Alatas, "Plant intelligence based metaheuristic optimization algorithms," *Artif. Intell. Rev.*, vol. 47, no. 4, pp. 417–462, 2017.

[11] H. Bingol and B. Alatas, "Chaos based optics inspired optimization algorithms as global solution search approach," *Chaos, Solitons Fractals*, vol. 141, Dec. 2020, Art. no. 110434.

[12] X. D. Li, J. S. Wang, W. K. Hao, M. Zhang, and M. Wang, "Chaotic arithmetic optimization algorithm," *Appl. Intell.*, Mar. 2022, doi: 10.1007/s10489-021-03037-3.

[13] S. Mahajan, L. Abualigah, A. K. Pandit, and M. Altalhi, "Hybrid Aquila optimizer with arithmetic optimization algorithm for global optimization tasks," *Soft Comput.*, vol. 26, no. 10, pp. 4863–4881, May 2022.

[14] A. Kaveh and K. Biabani Hamedani, "Improved arithmetic optimization algorithm and its application to discrete structural optimization," *Structures*, vol. 35, pp. 748–764, Jan. 2022.

[15] G. Hu, J. Zhong, B. Du, and G. Wei, "An enhanced hybrid arithmetic optimization algorithm for engineering applications," *Comput. Methods Appl. Mech. Eng.*, vol. 394, May 2022, Art. no. 114901.

[16] S. Mahajan, L. Abualigah, and A. K. Pandit, "Hybrid arithmetic optimization algorithm with hunger games search for global optimization," *Multimedia Tools Appl.*, Mar. 2022, doi: 10.1007/s11042-022-12922-z.

[17] Y.-J. Zhang, Y.-X. Yan, J. Zhao, and Z.-M. Gao, "AOAAO: The hybrid algorithm of arithmetic optimization algorithm with Aquila optimizer," *IEEE Access*, vol. 10, pp. 10907–10933, 2022.

[18] L. Abualigah, K. H. Almotairi, M. A. Elaziz, M. Shehab, and M. Altalhi, "Enhanced flow direction arithmetic optimization algorithm for mathematical optimization problems with applications of data clustering," *Eng. Anal. Boundary Elements*, vol. 138, pp. 13–29, May 2022.

[19] Q. Liu, N. Li, H. Jia, Q. Qi, L. Abualigah, and Y. Liu, "A hybrid arithmetic optimization and golden sine algorithm for solving industrial engineering design problems," *Mathematics*, vol. 10, no. 9, p. 1567, 2022.

[20] E. Pashaei and E. Pashaei, "Hybrid binary arithmetic optimization algorithm with simulated annealing for feature selection in high-dimensional biomedical data," *J. Supercomput.*, Apr. 2022, doi: 10.1007/s11227-022-04507-2.

[21] Z. Liu, M. Li, G. Pang, H. Song, Q. Yu, and H. Zhang, "A multi-strategy improved arithmetic optimization algorithm," *Symmetry*, vol. 14, no. 5, p. 1011, May 2022.

[22] N. Khodadadi, V. Snasel, and S. Mirjalili, "Dynamic arithmetic optimization algorithm for truss optimization under natural frequency constraints," *IEEE Access*, vol. 10, pp. 16188–16208, 2022.

[23] R. Zheng, H. Jia, L. Abualigah, Q. Liu, and S. Wang, "An improved arithmetic optimization algorithm with forced switching mechanism for global optimization problems," *Math. Biosci. Eng.*, vol. 19, no. 1, pp. 473–512, 2022.

[24] T. S. L. V. Ayyarao, N. S. S. Ramakrishna, R. M. Elavarasan, N. Polumahanthi, M. Rambabu, G. Saini, B. Khan, and B. Alatas, "War strategy optimization algorithm: A new effective Metaheuristic algorithm for global optimization," *IEEE Access*, vol. 10, pp. 25073–25105, 2022.

[25] B. Alatas, "ACROA: Artificial chemical reaction optimization algorithm for global optimization," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 13170–13180, 2011.

[26] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," *Comput. Methods Appl. Mech. Eng.*, vol. 376, Apr. 2021, Art. no. 113609.

[27] P. Barthelemy, J. Bertolotti, and D. S. Wiersma, "A Lévy flight for light," *Nature*, vol. 453, no. 7194, pp. 495–498, 2008.

[28] W. Gong and Z. Cai, "Differential evolution with ranking-based mutation operators," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2066–2081, Dec. 2013.

[29] S. S. Kannan and N. Ramaraj, "A novel hybrid feature selection via symmetrical uncertainty ranking based local memetic search algorithm," *Knowl.-Based Syst.*, vol. 23, no. 6, pp. 580–585, Aug. 2010.

[30] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in Statistics*. New York, NY, USA: Springer, 1992, pp. 196–202.

[31] A. Sadollah, H. Eskandar, and J. H. Kim, "Water cycle algorithm for solving constrained multi-objective optimization problems," *Appl. Soft Comput.*, vol. 27, pp. 279–298, Feb. 2015.

[32] Q. Luo, X. Yang, and Y. Zhou, "Nature-inspired approach: An enhanced moth swarm algorithm for global optimization," *Math. Comput. Simul.*, vol. 159, pp. 57–92, May 2019.

[33] J. Zhang, Y. Zhou, and Q. Luo, "Nature-inspired approach: A wind-driven water wave optimization algorithm," *Int. J. Speech Technol.*, vol. 49, no. 1, pp. 233–252, Jan. 2019.

[34] F. Miao, Y. Zhou, and Q. Luo, "Complex-valued encoding symbiotic organisms search algorithm for global optimization," *Knowl. Inf. Syst.*, vol. 58, no. 1, pp. 209–248, Jan. 2019.

[35] Y. Zhou, S. Zhang, Q. Luo, and M. Abdel-Baset, "CCEO: Cultural cognitive evolution optimization algorithm," *Soft Comput.*, vol. 23, no. 23, pp. 12561–12583, Dec. 2019.

[36] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[37] C. A. C. Coello, "Constraint-handling using an evolutionary multiobjective optimization technique," *Civil Eng. Environ. Syst.*, vol. 17, no. 4, pp. 319–346, 2000.

[38] K. Deb, "An efficient constraint handling method for genetic algorithms," *Comput. Methods Appl. Mech. Eng.*, vol. 186, nos. 2–4, pp. 311–338, 2000.

[39] K. Deb, "Optimal design of a welded beam via genetic algorithms," *AIAA J.*, vol. 29, no. 11, pp. 2013–2015, Nov. 1991.

[40] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: Harmony search theory and practice," *Comput. Methods Appl. Mech. Eng.*, vol. 194, no. 3, pp. 3902–3933, 2005.

[41] K. M. Ragsdell and D. T. Phillips, "Optimal design of a class of welded structures using geometric programming," *J. Eng. Ind.*, vol. 98, no. 3, pp. 1021–1025, Aug. 1976.

[42] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization*. Berlin, Germany: Springer, 2010, pp. 65–74.

[43] F.-Z. Huang, L. Wang, and Q. He, "An effective co-evolutionary differential evolution for constrained optimization," *Appl. Math. Comput.*, vol. 186, no. 1, pp. 340–356, 2007.

[44] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, "Water cycle algorithm—A novel metaheuristic optimization method for solving constrained engineering optimization problems," *Comput. Struct.*, vols. 110–111, pp. 151–166, Nov. 2012.

[45] A. Sadollah, A. Bahreininejad, H. Eskandar, and M. Hamdi, "Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems," *Appl. Soft Comput.*, vol. 13, no. 5, pp. 2592–2612, May 2013.

[46] A. Kaveh and S. Talatahari, "An improved ant colony optimization for constrained engineering design problems," *Eng. Comput.*, vol. 27, no. 1, pp. 155–182, Jan. 2010.

[47] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: Ray optimization," *Comput. Struct.*, vol. 112, pp. 283–294, Dec. 2012.

[48] T. Ray and K. M. Liew, "Society and civilization: An optimization algorithm based on the simulation of social behavior," *IEEE Trans. Evol. Comput.*, vol. 7, no. 4, pp. 386–396, Aug. 2003.

[49] H. Liu, Z. Cai, and Y. Wang, "Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization," *Appl. Soft Comput.*, vol. 10, no. 2, pp. 629–640, Mar. 2010.

[50] A. Baykasoğlu and Ş. Akpinar, "Weighted superposition attraction (WSA): A swarm intelligence algorithm for optimization problems—Part 2: Constrained optimization," *Appl. Soft Comput.*, vol. 37, pp. 396–415, Dec. 2015.

[51] T.-J. Hsieh, "A bacterial gene recombination algorithm for solving constrained optimization problems," *Appl. Math. Comput.*, vol. 231, pp. 187–204, Mar. 2014.

[52] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Appl. Math. Comput.*, vol. 188, no. 2, pp. 1567–1579, May 2007.

[53] E. Zahara and Y.-T. Kao, "Hybrid Nelder–Mead simplex search and particle swarm optimization for constrained engineering design problems," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3880–3886, 2009.

[54] Q. He and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Eng. Appl. Artif. Intell.*, vol. 20, pp. 89–99, Feb. 2007.

[55] V. Punnathanam and P. Kotecha, "Yin-Yang-pair optimization: A novel lightweight optimization algorithm," *Eng. Appl. Artif. Intel.*, vol. 54, pp. 62–79, Sep. 2016.

[56] E. Mezura-Montes and C. A. C. Coello, "An empirical study about the usefulness of evolution strategies to solve constrained optimization problems," *Int. J. Gen. Syst.*, vol. 37, no. 4, pp. 443–473, 2008.

[57] B. Akay and D. Karaboga, "Artificial bee colony algorithm for large-scale problems and engineering design optimization," *J. Intell. Manuf.*, vol. 23, no. 4, pp. 1001–1014, 2012.

[58] A. Baykasoğlu and F. B. Ozsoydan, "Adaptive firefly algorithm with chaos for mechanical design optimization problems," *Appl. Soft Comput.*, vol. 36, pp. 152–164, Nov. 2015.

[59] Y. Lu, Y. Zhou, and X. Wu, "A hybrid lightning search algorithm-simplex method for global optimization," *Discrete Dyn. Nature Soc.*, vol. 2017, Jul. 2017, Art. no. 8342694.

[60] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, 2009.

[61] A.-A.-A. Mohamed, A. A. M. El-Gaafary, Y. S. Mohamed, and A. M. Hemeida, "Multi-objective states of matter search algorithm for TCSC-based smart controller design," *Electr. Power Syst. Res.*, vol. 140, pp. 874–885, Nov. 2016.

[62] R. Menzel, U. Greggers, and M. Hammer, *Functional Organization of Appetitive Learning and Memory in a Generalist Pollinator, the Honey Bee*. New York, NY, USA: Springer, 1993.

[63] S. Mirjalili and S. Z. M. Hashim, "A new hybrid PSOGSA algorithm for function optimization," in *Proc. Int. Conf. Comput. Inf. Appl.*, Jan. 2012, pp. 374–377.

[64] H. Garg, "A hybrid PSO-GA algorithm for constrained optimization problems," *Appl. Math. Comput.*, vol. 274, pp. 292–305, Feb. 2016.

[65] H. R. E. H. Bouchekara, "Most valuable player algorithm: A novel optimization algorithm inspired from sport," *Oper. Res.*, vol. 20, no. 1, pp. 139–195, Mar. 2020.

[66] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems," *Comput.-Aided Des.*, vol. 43, no. 3, pp. 303–315, Mar. 2011.

[67] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

[68] M. Montemurro, A. Vincenti, and P. Vannucci, "The automatic dynamic penalisation method (ADP) for handling constraints with genetic algorithms," *Comput. Methods Appl. Mech. Eng.*, vol. 256, pp. 70–87, Apr. 2013.

[69] E. Mezura-Montes, C. A. C. Coello, J. Velázquez-Reyes, and L. Muñoz-Dávila, "Multiple trial vectors in differential evolution for engineering design," *Eng. Optim.*, vol. 39, no. 5, pp. 567–589, 2007.

[70] C. A. C. Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Comput. Ind.*, vol. 41, no. 2, pp. 113–127, Mar. 2000.

[71] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: A nature-inspired algorithm for global optimization," *Neural Comput. Appl.*, vol. 27, no. 2, pp. 495–513, Feb. 2016.

[72] Y. Zhou, Y. Ling, and Q. Luo, "Lévy flight trajectory-based whale optimization algorithm for engineering optimization," *Eng. Comput.*, vol. 35, no. 7, pp. 2406–2428, 2018.

[73] W. Long, W. Zhang, Y. Huang, and Y. Chen, "A hybrid cuckoo search algorithm with feasibility-based rule for constrained structural optimization," *J. Central South Univ.*, vol. 21, no. 8, pp. 3197–3204, Aug. 2014.

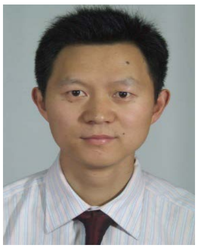
[74] Z. Li, Y. Zhou, S. Zhang, and J. Song, "Lévy-flight moth-flame algorithm for function optimization and engineering design problems," *Math. Problems Eng.*, vol. 2016, Aug. 2016, Art. no. 1423930.

[75] Z. Wang, Q. Luo, and Y. Zhou, "Hybrid Metaheuristic algorithm using butterfly and flower pollination base on mutualism mechanism for global optimization problems," *Eng. Comput.*, vol. 37, no. 4, pp. 3665–3698, Oct. 2021.

[76] E. Mezura-Montes, J. Velazquez-Reyes, and C. A. Coello Coello, "Modified differential evolution for constrained optimization," in *Proc. IEEE Int. Conf. Evol. Comput.*, Jul. 2006, pp. 25–32.

[77] Y. Wang, Z. Cai, Y. Zhou, and Z. Fan, "Constrained optimization based on hybrid evolutionary algorithm and adaptive constraint-handling technique," *Struct. Multidisciplinary Optim.*, vol. 37, no. 4, pp. 395–413, 2009.
[78] P. Savsani and V. Savsani, "Passing vehicle search (PVS): A novel metaheuristic algorithm," *Appl. Math. Model.*, vol. 40, nos. 5–6, pp. 3951–3978, 2016.
[79] Z. Yan, J. Zhang, J. Zeng, and J. Tang, "Nature-inspired approach: An enhanced whale optimization algorithm for global optimization," *Math. Comput. Simul.*, vol. 185, pp. 17–46, Jul. 2021.

JINZHONG ZHANG was born in Fuyang, China, in 1989. He received the M.S. degree in computer science from Guangxi University for Nationalities, in 2018, and the Ph.D. degree in control science and engineering from Harbin Engineering University, in 2022. He is currently a Lecturer at the School of Electrical and Optoelectronic Engineering, West Anhui University. His research interests include computation intelligence, image processing, and evolutionary computation.



GANG ZHANG was born in 1957. He received the B.S. degree in applied physics and the Ph.D. degree in electromagnetic fields and microwave technologies from Anhui University, in 1999 and 2009, respectively. He is currently a Professor. His research interests include intelligent control, quantum information, and evolutionary computation.

YOURUI HUANG was born in 1971. He received the B.S. degree in automated industry from the Anhui University of Science and Technology, in 1994, and the Ph.D. degree from the Anhui Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, in 2000. He is currently a Professor and a Doctoral Advisor. His current research interests include intelligent control, the Internet of Things technology, and so on.

MIN KONG was born in Anhui, China, in 1964. He received the B.S. degree from the Radio Department, Anhui University, Hefei, China, in 1985, and the Ph.D. degree in computer application technology from Anhui University, in 2006. He is currently a Professor with West Anhui University, Lu'an, China. His current research interests include modern signal processing, pattern recognition, and image processing.