Hindawi
Complexity
Volume 2021, Article ID 6694695, 17 pages
https://doi.org/10.1155/2021/6694695

Research Article

A Novel BBO Algorithm Based on Local Search and Nonuniform Variation for Iris Classification

Lisheng Wei,[1,2] **Ning Wang**,[1,3] **and Huacai Lu**[1]

[1]*Anhui Key Laboratory of Electric Drive and Control, Anhui Polytechnic University, Wuhu 241000, China*
[2]*Key Laboratory of Advanced Perception and Intelligent Control of High-end Equipment of Ministry of Education, AnHui Polytechnic University, Wuhu 241000, China*
[3]*Anhui Key Laboratory of Detection Technology and Energy Saving Devices, Anhui Polytechnic University, Wuhu 241000, China*

Correspondence should be addressed to Lisheng Wei; lshwei_11@163.com and Ning Wang; 18375318758@163.com

Received 6 November 2020; Revised 8 January 2021; Accepted 27 March 2021; Published 15 April 2021

Academic Editor: Shi Cheng

Copyright © 2021 Lisheng Wei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to improve the iris classification rate, a novel biogeography-based optimization algorithm (NBBO) based on local search and nonuniform variation was proposed in this paper. Firstly, the linear migration model was replaced by a hyperbolic cotangent model which was closer to the natural law. And, the local search strategy was added to traditional BBO algorithm migration operation to enhance the global search ability of the algorithm. Then, the nonuniform variation was introduced to enhance the algorithm in the later iteration. The algorithm could achieve a stronger iris classifier by lifting weaker similarity classifiers during the training stage. On this base, the convergence condition of NBBO was proposed by using the Markov chain strategy. Finally, simulation results were given to demonstrate the effectiveness and efficiency of the proposed iris classification method.

1. Introduction

Iris classification is especially suitable for recognition with uniqueness, stability, inviolability, and reliability characteristics and has been one of the hottest biological characteristic recognition research spot recently [1]. It has been widely used technology in national defence, financial industry, and entrance guard system [2, 3]. Alfred Wallace and Charles Darwin proposed the theory of biogeography to study the distribution, migration, and extinction of habitats of biological species in the 19th century. Inspired by biogeography, Simon proposed the BBO algorithm in 2008 [4]. BBO is a new swarm intelligence optimization algorithm and solves the optimization problem by simulating the mathematical model of species migration in biogeography. In general, BBO can find a good initial candidate solution and generate an acceptable optimization solution, which has been widely used in image processing [5], scheduling optimization [6, 7], parameter estimation [8], power flow calculation [9], and load analysis [10]. However, BBO has some shortcomings, such as weak search ability and easy to fall into local optimum in the later period of operation. How to use BBO global search ability to improve the iris classification rate has important theoretical and practical value.

In recent years, many researchers have worked on the improvement of the BBO performance. For instance, Wang et al. [11] combined the chaotic mapping strategy with the BBO optimal migration model and proposed a biogeography optimization algorithm based on based on the adaptive population migration mechanism. Feng et al. [12] proposed an improved BBO with random ring topology and Powell's method, in which the local loop topology was used instead of the global topology. The adaptive Powell method was modified to adapt the evolutionary process to improve the accuracy of the solution. Li et al. [13] designed a hybrid algorithm by combining the artificial bee colony algorithm (ABC) with BBO to obtain ABC's exploring ability and BBO's developing ability. Chen et al. [14] combined Cuckoo Search (CS) and bBBO using two search strategies: heterogeneous rhododendron search and biogeography-based discovery. Zhao et al. [15] proposed an optimization method based on two-stage differential biogeography to solve the

problem of premature convergence and reduce rotation variance.

In order to improve the iris classification rate, the NBBO based on local search and nonuniform variation was proposed in this paper. Firstly, the hyperbolic cotangent migration model was used to replace the original linear migration model to improve the adaptability of the algorithm. Then, the local search strategy was added to the BBO migration operation in order to enhance the algorithm search ability and improve the convergence speed of the algorithm. The nonuniform variation was used in the operation to increase the search ability of the algorithm and prevent the local optimum, which could achieve a stronger iris classifier by lifting weaker similarity classifiers during the training stage.

The rest of the paper was organized as follows. Section 2 presented the improved biogeography-based optimization algorithm, including migration model, migration operations, and nonuniform variation operation. Section 3 presented NBBO test on different test functions and iris classification samples. Finally, a brief conclusion and future work are given in Section 4.

2. Methods

In nature, biological species live in different areas with obvious boundaries, which are called habitats. Habitat suitability index (HSI) is used to describe the survival degree of species suitable for habitat. HSI is affected by humidity, temperature, habitat area, and vegetation diversity. These factors are called suitability variables (SIV). Habitats with higher HSI can accommodate more biological species, whereas habitats with lower HSI can only accommodate fewer biological species. BBO is a population-based algorithm, which regards each solution as a habitat, fitness as HSI, and each component of solution as a SIV. The optimization problem can be solved by simulating the migration and mutation process in biogeography. The two main operations of BBO are migration operation and mutation operation [16, 17].

Like other intelligent algorithms, BBO is prone to oscillation in the iterative process, resulting in slow and premature convergence, poor local search ability, and inaccuracy. This section presents the NBBO algorithm in terms of migration model and operation. The flow chart of the improved algorithm is shown in Figure 1.

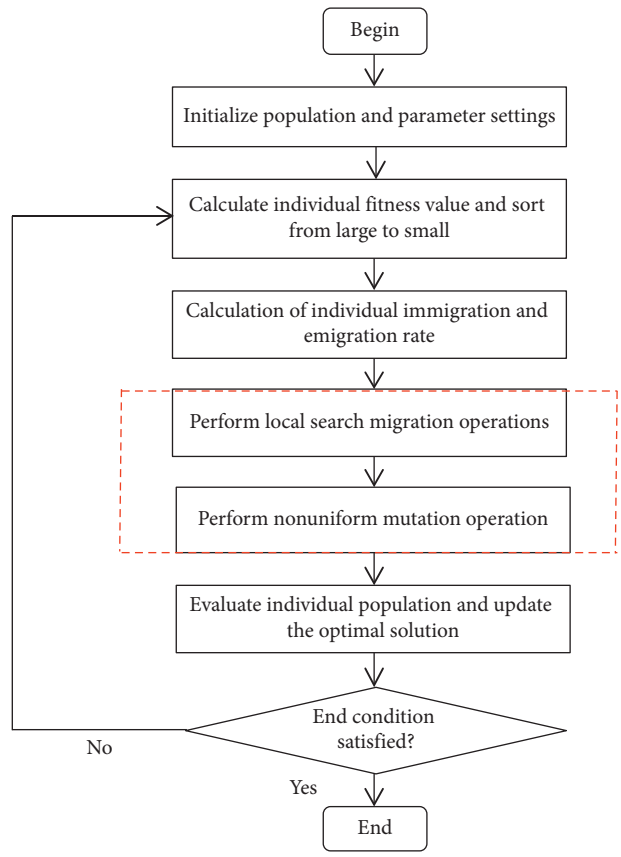

Figure 1: NBBO algorithm flowchart.

2.1. Migration Model. There are different migration models in biogeography. As shown in formula (1), the original BBO adopts the simplest linear migration model. But, the ecosystem is nonlinear in nature; small changes in a certain part of the system may have complex effects on the whole system. For the actual species, the migration process is more complicated than what is described by the linear model. Among the related improved algorithms, the cosine mobility model proposed by Ma performs best under most of the functions tested [18], and the mobility model close to the natural law is better than the simple linear model. In order to better adapt to the nonlinear migration problem, this paper proposes a hyperbolic cotangent nonlinear migration model to improve the performance of the basic BBO algorithm.

$$\begin{cases} \lambda_k = I \cdot \left(1 - \dfrac{k}{n}\right), \\ \mu_k = E \cdot \dfrac{k}{n}, \end{cases} \tag{1}$$

$$\lambda_k = \frac{I}{2} \cdot \left(-\frac{e^{k\pi/n} - e^{-k\pi/n}}{e^{k\pi/n} + e^{-k\pi/n}} + 1\right), \tag{2}$$

$$\mu_k = \frac{E}{2} \cdot \left(\frac{e^{k\pi/n} - e^{-k\pi/n}}{e^{k\pi/n} + e^{-k\pi/n}} + 1\right), \tag{3}$$

where λ_k and μ_k are related to the number of species k and n is the maximum number of species. λ_k represents the immigration rate which refers to the probability of species moving into the habitat, μ_k represents the emigration rate which refers to the probability of species moving out of the habitat, I represents the maximum immigration rate, and E represents the maximum emigration rate, all of which are functions of the number of habitat populations k. When the number of habitat species is 0, the immigration rate is the highest, that is, $\lambda_k = I$, the migration rate is the smallest, that is, $\mu_k = 0$. When the number of species reaches the

maximum number of species n that can be accommodated in the habitat, the immigration rate λ_k drops to 0, and the migration rate reaches the maximum, that is, $\mu_k = E$. When the number of species is k_0, the immigration rate and the emigration rate of the species are equal, that is, $\lambda_k = \mu_k$, and the dynamic equilibrium state is reached at this time.

In the hyperbolic tangential mobility model, the trend of mobility changing with species number is like that of cosine model, but the amplitude is more moderate than that of cosine model.

2.2. Local Search Migration Operations. In the original BBO, a generalized search was carried out based on the immigration rate and the emigration rate, a habitat with a higher emigration rate was randomly selected, and the characteristics of the corresponding position (SIV) were moved to the habitat with low HSI. However, in some habitats with high migration rates, the characteristics of the corresponding locations SIV are not necessarily the best for the migrated habitats. If migrated directly, it may result in a decrease in the suitability index of the move into habitat.

Therefore, in order to enhance the global search ability of the algorithm and improve the convergence speed of the algorithm, the selected migration characteristics can be modified by using equation (6), which can reduce the poor migration characteristics from the habitat with higher migration rate to the low migration rate. Similarly, if you move out of the same habitat with a high migration rate each time, as the number of evolutions increases, the population diversity decreases, and the convergence speed will slow down:

$$C = \text{MinVal} + \text{rand} * (\text{MaxVal} - \text{MinVal}), \quad (4)$$

$$\text{randTemp} = \text{rand}(1, 2), \quad (5)$$

$$X'(k, j) = X(k, j) + C * \text{randTemp} * X(k, j), \quad (6)$$

where C represents a space allocated to each individual in the population for storing algorithm parameters. $X(k, j)$ represents the current individual.$X'(k, j)$ represents the new individual generated by the local search. MinVal = −1 and MaxVal = 1.

2.3. Nonuniform Variation Operation. In the early stage of iteration, the population position is usually far from the optimal solution and needs to be iterated many times to find a feasible search space when the search radius is too small. At the end of iteration, the population position is close to or near the optimal solution. A small range of searches is used as a fine-tuning of the solution vector. It is shown by equation (7), as the traditional mutation operator cannot search the optimal solution efficiently, the nonuniform mutation operator is introduced. The whole population is searched at a large range during the early stage of search. As the number of iterations increases, the search range is gradually narrowed, and the algorithm is effectively balanced to escape from local minimum.

$$m_s = m_{\max} \cdot \left(1 - \frac{P_s}{P_{\max}}\right), \quad (7)$$

where P_s denotes the probability for the S number of species; S denotes the species number; and $m_{\max}$ denotes the maximum variation rate given; $P_{\max} = \max(P_s)$.

The specific operation of nonuniform variation is like uniform variation, but it focuses on searching for small areas near the original individual. Assuming that the variation of the k_{th} component X_k of the t generation (t is the current number of iterations) of Habitat $X_i^t = \{X_1, X_2, \ldots, X_k, \ldots, X_n\}$ is performed, the variation component is as follows:

$$X_k' = X_k + \Delta(t, U_{\max} - X_k), \quad \text{if rand} < 0.5, \quad (8)$$

$$X_k' = X_k - \Delta(t, X_k - U_{\min}), \quad \text{if rand} \geq 0.5, \quad (9)$$

where $U_{\max}$ and $U_{\min}$ are the upper and lower limits of component X_k, respectively. $U_{\max} - X_k$ and $X_k - U_{\min}$ represent random numbers in the range of $[0, y]$ and meet the nonuniform distribution:

$$\lim_{t \longrightarrow \infty} \Delta(t, y) = 0. \quad (10)$$

This feature allows the mutation operator to search the entire space evenly in the initial stage of the algorithm (when t is very small), and the latter part of the algorithm focuses on several local range exact searches. $\Delta(t, y)$ can be defined as follows:

$$\Delta(t, y) = y \cdot \left(1 - r^{(1-(t/T))\cdot b}\right), \quad (11)$$

where r is a random number that fits evenly within the $[0, 1]$ range, T is the maximum number of iterations, and b is the system parameter that determines the degree of dependency on the number of iterations, select here $b = 2$.

Then, the specific steps of NBBO algorithm are as follows: (Algorithm 1)

2.4. Convergence Analysis. By using Markov chain strategy, we can analyze the convergence of the proposed NBBO algorithms [19, 20]. Let $X(t) = \{x_1(t), x_2(t), \ldots, x_n(t)\}$ be a population with improved algorithm algebra of t. We divide $X(t)$ into n subsets based on the fitness of the population. $x_i(t)$ $(i = 1, 2, \ldots, n)$ is a subset of them; next, we use Markov chain to analyze the global convergence of the improved BBO algorithm.

The evolution process of the NBBO algorithm is mainly composed of selection, migration and mutation operations, which are independent of evolution algebra. Considering the number of individuals of BBO algorithm, the evolution process of the NBBO algorithm satisfies the finite homogeneous Markov chain. Population subset $x_i(t)$ $(i = 1, 2, \ldots, n)$ can be equivalent to a state on a finite homogeneous Markov chain, p_i means the probability of being in state $x_i(t)$, and then $p_i > 0$ and $\sum_{i=1}^{n} p_i = 1$.

The transition probability $p_{ij} = p(x_i \longrightarrow x_j)$ of state $x_i(t)$ is as follows:

$$\begin{cases} p_{ij} = 0, & i < j \\ p_{ij} > 0, & i \geq j. \end{cases} \tag{12}$$

Then, the state transition matrix P of the Markov chain is

$$P = \begin{bmatrix} p_{11} & 0 & \cdots & 0 \\ p_{21} & p_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} = \begin{bmatrix} p_{11} & 0 \\ Q_2 & Q_1 \end{bmatrix}, \tag{13}$$

where

$$Q_1 = \begin{bmatrix} p_{22} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ p_{n2} & \cdots & p_{nn} \end{bmatrix}, \quad Q_2 = (p_{21}, p_{31}, \cdots, p_{n1})^T. \tag{14}$$

It can be found by formulas (12) and (13) $Q_2 \neq 0$, $Q_1 \neq 0$. Since the sum of the probabilities of each row in the Markov transition matrix is 1, $p_{11} = 1 \neq 0$. It can be concluded that the state transition matrix P of Markov chain satisfies the reducible random matrix. According to [21], the definition of a reductive random matrix is as follows:

Let matrix Z of order N be a reducible random matrix which can be obtained by the same row transformation and column transformation, namely,

$$Z = \begin{bmatrix} C & 0 \\ R & T \end{bmatrix}, \tag{15}$$

where C is the m ($m \leq N$) order primitive matrix and R and T are nonzero matrices.

Then, we have

$$Z^{\infty} = \lim_{k \longrightarrow \infty} Z^k = \lim_{k \longrightarrow \infty} \begin{bmatrix} C^k & 0 \\ \sum_{i=1}^{k-1} T^i R C^{k-i} & T^k \end{bmatrix} = \begin{bmatrix} C^{\infty} & 0 \\ R^{\infty} & 0 \end{bmatrix}. \tag{16}$$

The above matrix is a stable random matrix, and Z^{∞} satisfies the following conditions:

$$Z^{\infty} = [Z_{ij}]_{N*N}, \begin{cases} Z_{ij} > 0, & 1 \leq i \leq N, 1 \leq j \leq m, \\ Z_{ij} = 0, & 1 \leq i \leq N, m < j \leq N. \end{cases} \tag{17}$$

Therefore, state transition matrix P is a reductive random matrix, then

$$\begin{aligned} P^{\infty} &= \lim_{t \longrightarrow \infty} \begin{bmatrix} p_{11} & 0 \\ Q_2 & Q_1 \end{bmatrix}^t \\ &= \lim_{t \longrightarrow \infty} \begin{bmatrix} p_{11}^t & 0 \\ \sum_{i=1}^{t-1} Q_1^i Q_2 p_{11}^{t-i} & Q_1^t \end{bmatrix} \\ &= \lim_{t \longrightarrow \infty} \begin{bmatrix} p_{11}^{\infty} & 0 \\ Q_2^{\infty} & 0 \end{bmatrix}. \end{aligned} \tag{18}$$

In formula (12) $p_{11}^{\infty} = p_{11} = 1$ and $Q_1^{\infty} = 0$, since the sum of the probabilities of each line in the Markov transition matrix is 1; there must be $Q_2^{\infty} = (1, 1, \ldots, 1)^T$, then

$$P^{\infty} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix}. \tag{19}$$

Formula (19) shows that when the number of iterations $t \longrightarrow \infty$, probability $p_{i1}^{\infty} = 1$ ($i = 1, 2, \ldots, n$), That is, no matter how the initial state can finally converge to the global optimal solution with probability 1, it can be concluded that

$$\lim_{t \longrightarrow \infty} p\{Y_t = f^*\} = 1, \tag{20}$$

where Y_t is the optimal fitness value of population $X(t)$, f^* represents the global optimal fitness value, and $p\{Y_t = f^*\}$ indicates the probability that the optimal individual of the t generation is the global optimal value. It is proved that the NBBO algorithm has global convergence.

3. Simulation Experiments

3.1. Test Function Simulation. In order to test the optimization performance of the proposed IMBBO algorithm, 13 standard test functions were selected for simulation experiments [22, 23]. Table 1 shows the test function expression, the range of values for each independent variable, and the theoretical optimal value. We select the optimal value, average value, and standard deviation of the test results as the evaluation criteria to test the algorithm performance.

The experimental function of Table 1 is numerically tested using the proposed NBBO algorithm proposed in this paper, and the basic BBO algorithm, grey wolf optimization algorithm [24] (DE), elephant herding optimization [25] (EHO), salp swarm algorithm [26] (DEBBO), and particle swarm optimization biogeography optimization algorithm [27] and (BBOPSO) for comparison of results. The experimental parameter settings of the algorithm are shown in Table 2.

To study the performance of the algorithm in different dimensions, the dimensions of test function $f_1 - f_3$ and $f_5 - f_7$ are $D = 10, 30, 50$, the dimension of test functions f_4 is $D = 2$, and the dimension of test functions f_8 is $D = 4$. In order to ensure that the comparison between each method is fair, we ensure that the experimental environment of each method is consistent during the experiment: operating system: window 10, memory: 8 GB, programming language: MATLAB 2016, and ensure that the setting of common parameters of the algorithm is consistent, and the parameters of different algorithms are set according to the parameters of specific references. As the algorithm is running randomly, the six algorithms independently run each test function 30 times to avoid unnecessary errors caused by the random operation of the algorithm.

Tables 3 and 4 show the recorded optimal function value, average value, and standard deviation obtained by the

```
Begin
Algorithmic parameters setting:
Population size = 50,
Maximum number of iterations G = 500,
Maximum immigration rate I = 1,
Maximum emigration rate E = 1,
Mutation probability P_m = 0.05,
Number of elites retained Keep = 2.
    /* Initialization */
Random generation of a set of initial habitats constitutes the initial population.
Calculate the fitness value of each habitat in the population.
While not T /* T is the condition for the end of the iteration */
Sort the habitat in descending order according to the fitness value.
The immigration rate λ_i and emigration rate μ_i of each habitat were calculated according to formula (1).
    /* End of initialization */
    /* Migration */
p_i
  Select H_i according to the immigration probability λ_i
  for j = 1: D /* D is dimension */
    if rand < λ_i
      Select H_j according to the migration probability μ_i
        Randomly select a characteristic variable SIV from H_j
        Replace a random characteristic variable SIV in H_i with SIV
    else
        Perform a local search on the characteristic variable SIV
          according to formula (6) can get SIV'.
        Replace random characteristic variable SIV in H_i with SIV'.
    end if
  end for
end for
    /* End of migration */
    /* Mutation */
for i = 1: Size
  Calculate the probability of mutation P_s according to λ_i and μ_i
  Selection of unmutated habitat H_i based on P_s
  if rand(0, 1) < m_s then
    if rand < 0.5
    Replace H_i (SIV) with a randomly generated SIV by formula (8)
      if rand ≥ 0.5
        Replace H_i (SIV) with a randomly generated SIV by (9)
      end if
    end if
  end if
end for
/* End of mutation */
        Recalculate the habitat fitness values
end while
end
```

Algorithm 1: The main procedure of NBBO.

algorithm. In the iterative process of the algorithm, we record the best fitness of each evaluation, find the average optimal value of each evaluation and draw a curve, which reflects the convergence trend of the algorithm. The convergence curves of some test functions are shown in Figures 2 and 3.

In order to compare the optimization performance of each algorithm more easily and clearly, the algorithm with the best optimization performance of the optimal value, average value and standard deviation of each test function is boldly marked in Tables 3 and 4. From Tables 3 and 4 and the simulation diagram, we find that the convergence speed of the analytical algorithm can be qualitatively and quantitatively analyzed. For the simple multidimensional unimodal function in Figure 2, the NBBO algorithm has faster convergence speed and solution precision in the later iteration although its advantage is not obvious in the initial stage of iteration. For the more complex multidimensional multimodal function in Figure 3, the other five algorithms have the phenomenon of search stagnation or slow search, i.e.,

Table 1: Test function.

Function	Expression	Ranges	Theory optimal
$f1$: Quartic	$f(x_1) = \sum_{i=1}^{n} ix^4 + \text{random}[0,1)$	(−1.28, 1.28)	0
$f2$: Sphere	$f(x_4) = \sum_{i=1}^{n} x_i^2$	{−100, 100)	0
$f3$: Powell	$f(x_5) = \sum_{i=1}^{d/4} [(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4]$	(−4, 5)	0
$f4$: Easom	$f(x_{10}) = -\cos(x_1)\cos(x_2)\exp(-(x_1-\pi)^2 - (x_2-\pi)^2)$	(−100, 100)	−1
$f5$: Alpine	$f(x_6) = \sum_{i=1}^{n} \lvert x_i \sin x_i + 0.1x_i \rvert$	(10, 10)	0
$f6$: Griewank	$f(x_7) = (1/4000)\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(x_i/\sqrt{i}) + 1$	(−600, 600)	0
$f7$: Ackley	$f(x_8) = -20\exp(-0.2\sqrt{(1/n)\sum_{i=1}^{n} x_i^2}) - \exp((1/n)\sum_{i=1}^{n}\cos 2\pi x_i) + 20 + e$	(−32, 32)	0
$f8$: Kowalik	$f(x_{13}) = \sum_{i=1}^{11} [a_i - ((x_1(b_i^2 + b_i x_2))/(b_i^2 + b_i x_3 + x_4))]^2$	(−5, 5)	$3.075E-04$

Table 2: Algorithm parameter settings.

Setting object	Parameter	Value
Common part	Population size	Size = 50
	The maximum number of iterations	$G = 500$
	Elite reserves	Keep = 2
BBO/BBOPSO	Habitat modification probability	$P_{\text{mod}} = 1$
	Maximum immigration	$I = 1$
	Maximum migration	$E = 1$
	Mutation probability	$P_m = 0.05$
DE	Weighting factor	$F = 0.5$
	Crossover probability	CR = 0.5
EHO	The scale factors	$\alpha = 0.5$, $\beta = 0.5$
	The number of clans	$n\,\text{Clan} = 5$
DEBBO	Weighting factor	$F = 0.5$
	Crossover probability	CR = 0.9
	DE mutation scheme	DE/rand/1/bin
The proposed NBBO	Habitat modification probability	$P_{\text{mod}} = 1$
	Maximum immigration	$I = 1$
	Maximum migration	$E = 1$
	Mutation probability	$P_m = 0.05$
	System parameters	$b = 2$

they fall into local optimum and are difficult to jump out. Then, by comparing the optimal value of each test function and the average value of the optimization results, we can see that IMBBO algorithm has achieved a smaller value. Although the optimization results of DE, DEBBO, and BBOPSO algorithms also reach the global optimal value in the optimization of some test functions with different dimensions, it can be seen from the optimization results of single peak function and multi peak function that NBBO algorithm has a great advantage in the optimization ability. Figures 2 and 3 show that the proposed NBBO algorithm can not only improve the convergence of the algorithm but also improve the evolution ability and the ability to jump out of the local optimal solution. It is superior to the other four comparison algorithms in the stability of the optimization result. In summary, when solving the optimization problem of single-peak or multi-peak functions, the proposed NBBO can quickly and effectively find the optimal solution under a short number of iterations, and its optimal solution has higher precision and stability.

Considering the fixed dimension test function has few dimensions to be adjusted, the accuracy and stability of each algorithm are relatively high. For the multidimensional test function, when its dimension D is increased from 10 to 30 or 50, the accuracy and stability of the algorithm have decreased. This is because the objective function is more complicated and the algorithm needs to make more adjustments. However, no matter how the dimension changes, the proposed NBBO algorithm still maintains a stable and high-precision optimization performance for high-dimensional complex functions. In reference [28], derrick proposed that for the evaluation of the performance of the improved evolutionary algorithm, only based on the average value and the standard deviation value to compare the advantages and disadvantages of the algorithm is not rigorous enough, so statistical test should be carried out. The statistical test shows that the improved algorithm has significant improvement advantages over other existing algorithms. Therefore, in order to verify the significant difference between the IMBBO algorithm and other algorithms, a nonparametric estimation method, Wilcoxon rank sum test, is used for statistical analysis. For each test function, the 30 times solution results of IMBBO algorithm are tested with those of BBO, DE, EHO, DEBBO, and BBOPSO algorithms respectively, and H0 is assumed: the solution results of the two algorithms are generally the same; H1: the solution

Table 3: Unimodal function test results.

Function	Dimension	Evaluating indicator	Reference [4]BBO	Reference [24]DE	Reference [25]EHO	Reference [26]DEBBO	Reference [27]BBOPSO	The proposed method
$f_1(x)$ Quartic	$D=10$	Best	$3.40E-03$	$1.70E-03$	$2.10E-03$	$1.81E-04$	$7.92E-05$	**5.78 E − 05**
		Ave|	$1.37E-02$	$4.10E-03$	$6.60E-03$	$9.45E-04$	$1.20E-03$	**4.51 E − 04**
		Std	$9.80E-03$	$1.70E-03$	$3.70E-03$	$6.91E-04$	$8.62E-04$	**3.25 E − 04**
	$D=30$	Best	$7.58E-02$	$3.50E-02$	$8.44E-02$	$1.05E-02$	$6.20E-03$	**1.10 E − 03**
		Ave	$1.47E-01$	$5.46E-02$	$1.33E-01$	$1.46E-02$	$2.23E-02$	**2.70 E − 03**
		Std	$4.41E-02$	$1.03E-02$	$4.13E-02$	$4.20E-03$	$9.10E-03$	**2.00 E − 03**
	$D=50$	Best	$2.51E-01$	$1.21E-01$	$1.98E-01$	$3.47E-02$	$4.63E-02$	**3.00 E − 03**
		Ave	$5.67E-01$	$2.29E-01$	$5.37E-01$	$5.61E-02$	$7.82E-02$	**8.50 E − 03**
		Std	$2.08E-01$	$5.16E-02$	$1.38E-01$	$1.56E-02$	$2.22E-02$	**4.20 E − 03**
$f_2(x)$ Sphere	$D=10$	Best	$1.94E+00$	$1.18E-20$	$2.35E-04$	$6.72E-32$	$2.02E-28$	**3.66 E − 45**
		Ave	$8.38E+00$	$3.52E-20$	$3.74E-04$	$1.69E-29$	$1.37E-27$	**5.79 E − 43**
		Std	$6.12E+00$	$2.77E-20$	$6.88E-05$	$2.06E-29$	$1.49E-27$	**9.78 E − 43**
	$D=30$	Best	$7.13E+01$	$1.20E-03$	$3.00E-03$	$1.40E-10$	$6.73E-07$	**2.09 E − 24**
		Ave	$1.43E+02$	$2.60E-03$	$3.60E-03$	$5.83E-10$	$1.24E-05$	**1.43 E − 22**
		Std	$4.64E+01$	$2.20E-03$	$0.31E-04$	$4.90E-10$	$3.35E-05$	**3.91 E − 22**
	$D=50$	Best	$5.93E+02$	$1.58E+01$	$7.90E-03$	$2.38E-05$	$4.11E-02$	**4.66 E − 18**
		Ave	$7.27E+02$	$2.30E+01$	$9.00E-03$	$1.62E-04$	$1.64E-01$	**6.83 E − 17**
		Std	$1.34E+02$	$4.63E+00$	$1.20E-03$	$1.54E-04$	$9.32E-02$	**8.48 E − 17**
$f_3(x)$ Powell	$D=10$	Best	$2.47E-02$	$2.04E-05$	$4.64E-06$	$3.89E-04$	$2.63E-04$	**8.88 E − 13**
		Ave	$3.01E-01$	$1.52E-04$	$7.51E-06$	$2.40E-03$	$1.45E-03$	**5.16 E − 07**
		Std	$2.54E-01$	$1.04E-04$	$2.22E-06$	$1.80E-03$	$1.24E-03$	**7.21 E − 07**
	$D=30$	Best	$2.47E+00$	$6.19E+01$	$2.36E-04$	$1.21E-01$	$5.35E-02$	**1.13 E − 10**
		Ave	$5.77E+00$	$1.36E+01$	$4.10E-04$	$3.92E-01$	$2.90E-01$	**1.26 E − 06**
		Std	$3.75E+00$	$4.70E+01$	$1.50E-04$	$1.61E-01$	$1.25E-01$	**2.10 E − 06**
	$D=50$	Best	$1.73E+01$	$1.04E+03$	$2.20E-03$	$2.02E+00$	$1.63E+00$	**4.07 E − 09**
		Ave	$5.16E+01$	$1.87E+03$	$3.30E-03$	$4.55E+00$	$2.55E+00$	**4.69 E − 06**
		Std	$2.44E+01$	$4.83E+02$	$6.48E-04$	$2.61E+00$	$9.34E-01$	**6.33 E − 09**
$f_4(x)$ Easom	$D=2$	Best	$-9.13E-01$	**−1.00 E + 00**	$-9.95E-01$	**−1.00 E + 00**	**−1.00 E + 00**	**−1.00 E + 00**
		Ave	$-5.00E-01$	**−1.00 E + 00**	$-9.60E-01$	**−1.00 E + 00**	**−1.00 E + 00**	**−1.00 E + 00**
		Std	$4.45E-01$	**0.00 E + 00**	$3.55EE-02$	**0.00 E + 00**	**0.00 E + 00**	**0.00 E + 00**

Table 4: Multimodal function test results.

Function	Dimension	Evaluating indicator	Reference [4] BBO	Reference [24] DE	Reference [25] EHO	Reference [26] DEBBO	Reference [27] BBOPSO	The proposed method
$f_5(x)$ Alpine	$D=10$	Best	$2.59E-02$	$1.41E-19$	$3.59E-06$	$4.91E-06$	$9.43E-09$	**1.26 E − 26**
		Ave	$5.18E-02$	$2.97E-16$	$4.66E-06$	$8.09E-05$	$1.58E-05$	**8.04 E − 25**
		Std	$2.26E-02$	$5.21E-16$	$8.50E-07$	$5.62E-05$	$1.77E-05$	**1.73 E − 24**
	$D=30$	Best	$3.24E-01$	$7.41E+00$	$5.22E-06$	$4.50E-03$	$4.43E-03$	**9.64 E − 16**
		Ave	$7.83E-01$	$1.11E+01$	$7.12E-05$	$6.21E-02$	$9.06E-03$	**1.05 E − 13**
		Std	$2.94E-01$	$1.97E+00$	$1.75E-05$	$5.85E-02$	$3.07E-03$	**3.65 E − 13**
	$D=50$	Best	$2.32E+00$	$3.12E+01$	$1.68E-04$	$3.33E-01$	$4.26E-02$	**3.55 E − 13**
		Ave	$3.26E+00$	$3.62E+01$	$2.29E-04$	$1.15E+00$	$1.58E-01$	**1.03 E − 10**
		Std	$5.16E-01$	$2.45E+00$	$5.69E-05$	$5.32E-01$	$1.41E-01$	**2.92 E − 10**
$f_6(x)$ Griewank	$D=10$	Best	$1.00E+00$	$1.00E+00$	$1.10E+00$	**0.00 E + 00**	**0.00 E + 00**	**0.00 E + 00**
		Ave	$1.07E+00$	$1.00E+00$	$1.14E+00$	$2.48E-02$	$2.95E-02$	**0.00 E + 00**
		Std	$4.12E-02$	**0.00 E + 00**	$2.33E-02$	$1.57E-02$	$2.54E-02$	**0.00 E + 00**
	$D=30$	Best	$1.88E+00$	$1.00E+00$	$2.90E+00$	$6.03E-10$	$3.66E-07$	**0.00 E + 00**
		Ave	$2.40E+00$	$1.00E+00$	$3.61E+00$	$1.50E-03$	$1.24E-02$	**0.00 E + 00**
		Std	$3.36E-01$	$7.20E-04$	$5.98E-01$	$3.20E-03$	$2.40E-02$	**0.00 E + 00**
	$D=50$	Best	$4.86E+00$	$1.12E+00$	$7.72E+00$	$4.22E-05$	$9.11E-02$	**02.22 E − 17**
		Ave	$8.00E+00$	$1.18E+00$	$9.82E+00$	**1.90 E − 03**	$1.78E-01$	
		Std	$2.26E+00$	$3.33E-02$	$1.01E+00$	**5.00 E − 03**	$6.79E-02$	**7.02 E − 17**
$f_7(x)$ Ackley	$D=10$	Best	$8.70E-01$	$3.43E-11$	$7.20E-03$	**2.66 E − 15**	$6.22E-15$	**2.66 E − 15**
		Ave	$2.05E+00$	$8.75E-11$	$8.10E-03$	**2.66 E − 15**	$8.85E-13$	**2.66 E − 15**
		Std	$6.17E-01$	$6.33E-11$	$6.43E-04$	**0.00 E + 00**	$2.54E-12$	**0.00 E + 00**
	$D=30$	Best	$3.58E+00$	$1.28E-02$	$1.44E-02$	$5.12E-06$	$1.08E-04$	**1.23 E − 13**
		Ave	$4.30E+00$	$1.63E-02$	$1.56E-02$	$6.50E-06$	$2.66E-04$	**1.23 E − 13**
		Std	$4.09E-01$	$2.30E-03$	$7.98E-04$	$2.13E-06$	$1.25E-04$	**1.07 E − 12**
	$D=50$	Best	$5.43E+00$	$2.14E+00$	$1.71E-02$	$8.84E-04$	$2.74E-02$	**1.75 E − 10**
		Ave	$6.18E+00$	$2.57E+00$	$1.88E-02$	$1.70E-03$	$7.18E-02$	**9.85 E − 10**
		Std	$5.51E-01$	$2.98E-01$	$9.30E-02$	$1.20E-03$	$6.75E-02$	**9.57 E − 10**
$f_8(x)$ Kowalik	$D=4$	Best	$6.13E-04$	$3.17E-04$	$3.24E-04$	$3.16E-04$	$4.91E-04$	**3.08 E − 04**
		Ave	$2.70E-03$	$6.10E-04$	$3.52E-04$	$9.99E-04$	$7.38E-04$	**3.23 E − 04**
		Std	$4.30E-03$	$4.01E-04$	$2.27E-05$	$1.20E-03$	$3.14E-04$	**2.26 E − 05**

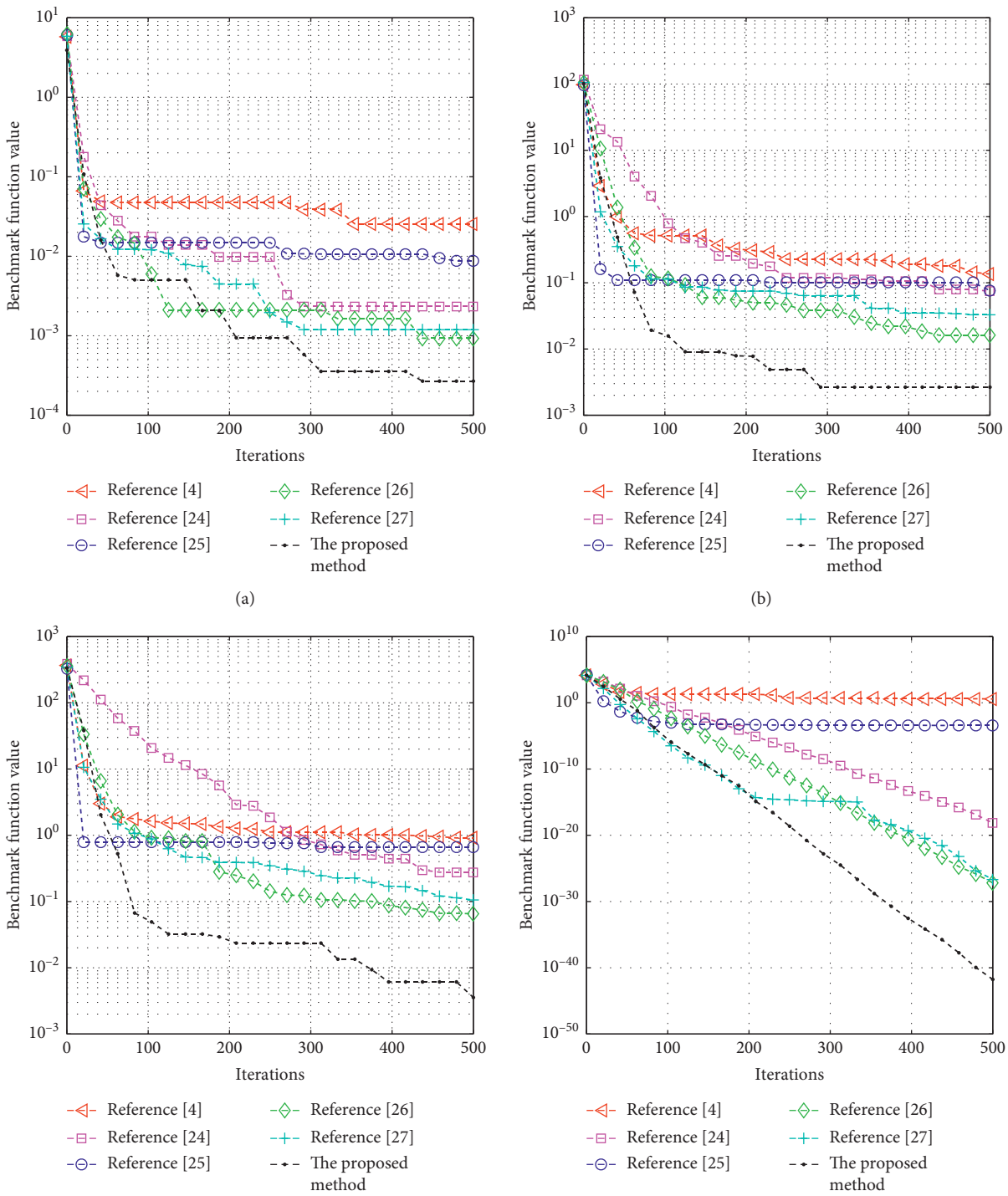

Figure 2: Continued.

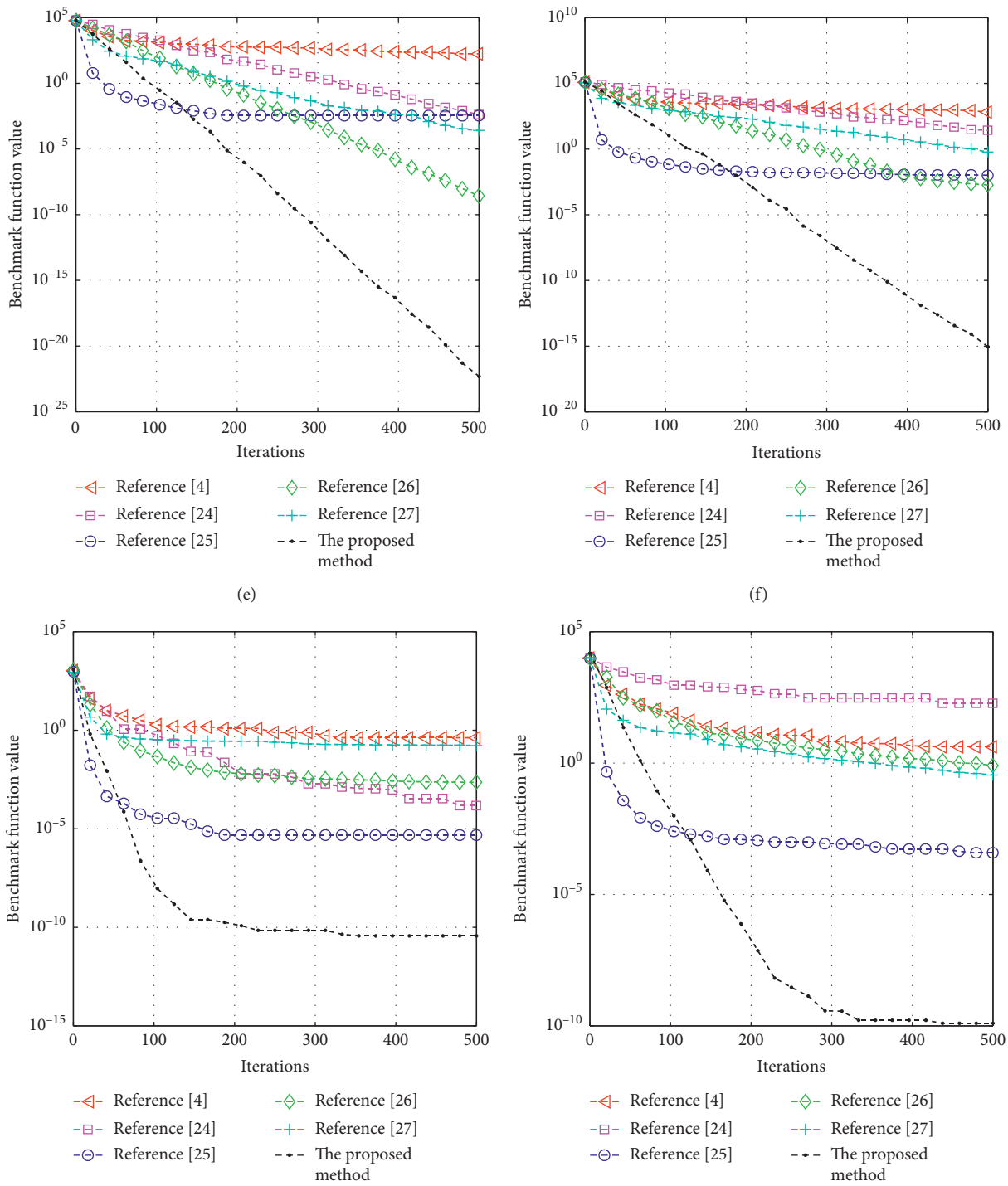

Figure 2: Continued.

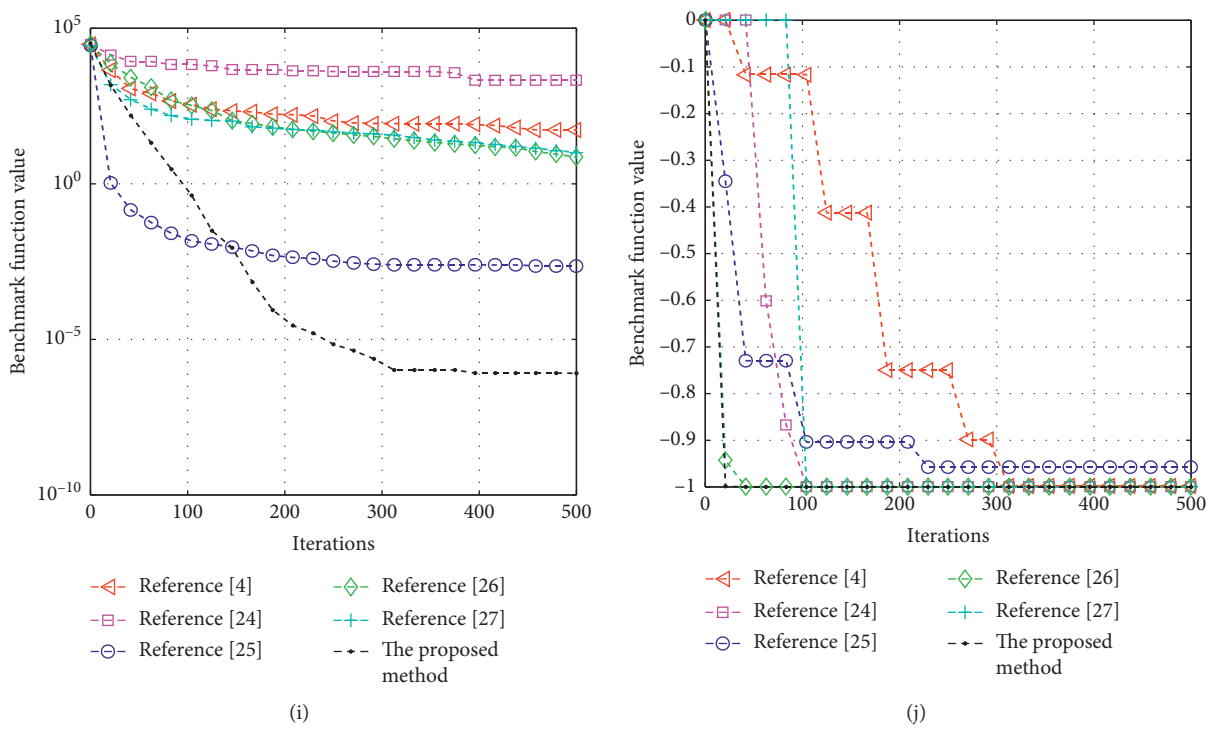

Figure 2: Unimodal function simulation results diagram. (a) f_1: Quartic ($d = 10$), (b) f_1: Quartic ($d = 30$), (c) f_1: Quartic ($d = 50$), (d) f_2: Sphere ($d = 10$), (e) f_2: Sphere ($d = 30$), (f) f_2: Sphere ($d = 50$), (g) f_3: Powell ($d = 10$), (h) f_3: Powell ($d = 30$), (i) f_3: Powell ($d = 50$), (j) f_4: Easom ($d = 2$).

results of the two algorithms are generally different. Set the threshold value of significance level of zero hypothesis test $\alpha = 0.05$. When the test result $P < 0.05$, reject the zero hypothesis; the experimental results of the two algorithms have significant difference; when the test result $P > 0.05$, accept the zero hypothesis; the experimental results of the two algorithms have no significant difference. The specific inspection results are shown in Tables 5 and 6.

Tables 5 and 6 show the test results of test function $f_1 - f_8$. Among them, for the probability p value less than the significance level threshold value of 0.05, bold display; "W" is the significance test judgment result, combined with Tables 3 and 4, use "+" to indicate that NBBO is more accurate and significant than other algorithms, use "−" to indicate that NBBO is less accurate and significant than other algorithms, and use "=" to indicate that NBBO algorithm. There is no significant difference in the accuracy of other algorithms; "N/A" means that significance judgment cannot be made. From the test results, most of the test p values are less than 0.05 and "W" is "+," so zero hypothesis is rejected. Therefore, there are significant differences between the calculation results of IMBBO and BBO, DE, EHO, DEBBO, and BBOPSO, and NBBO is significantly better.

To sum up, the biogeography-based optimization algorithm based on local search and nonuniform variation has good results for most of the benchmark functions, which effectively solves the slow convergence speed of the original BBO algorithm and easy to fall into the local optimization problem in the later stage of the search and greatly improves the convergence speed and search accuracy of the BBO algorithm without adding too many programming steps.

3.2. Iris Classification Based on NBBO Multilayer Perceptron. In order to test the training performance of NBBO multilayer perceptron for iris classification, the mean square error (MSE) of the actual output value of multilayer perceptron and the expected output value is used as an indicator to measure the performance of multilayer perceptron [29]. MSE of all training samples is calculated as follows:

$$\mathrm{MSE} = \frac{1}{s} \sum_{k=1}^{s} \sum_{i=1}^{m} \left(o_i^k - d_i^k\right)^2, \tag{21}$$

where s is the number of training samples, m is the number of outputs, d_i^k is the expected output of the i input unit under k training samples, and o_i^k is the actual output of the i input unit under k training samples. The MSE value is smaller is equal to the performance of the multilayer perceptron is better, and vice versa. The HIS of the i habitat is calculated by

$$\mathrm{HSI}\left(\mathrm{Habitat}_i\right) = E\left(\mathrm{Habitat}_i\right). \tag{22}$$

Figure 4 shows the schematic diagram of a multilayer perceptron based on NBBO algorithm. NBBO algorithm receives the average MSE of all training samples and all expected output samples as the objective function for iterative training and adjusts the link weights and deviations

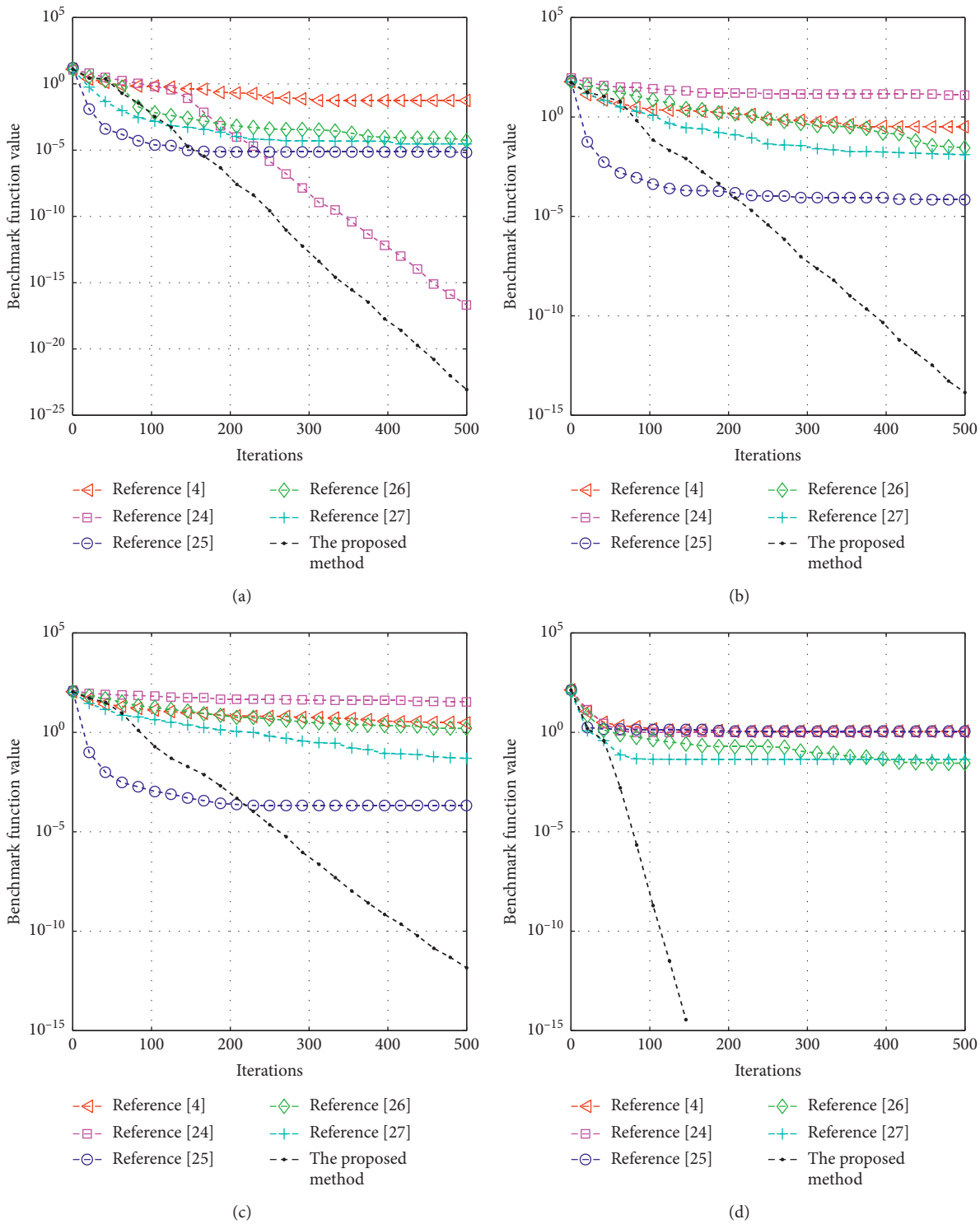

Figure 3: Continued.

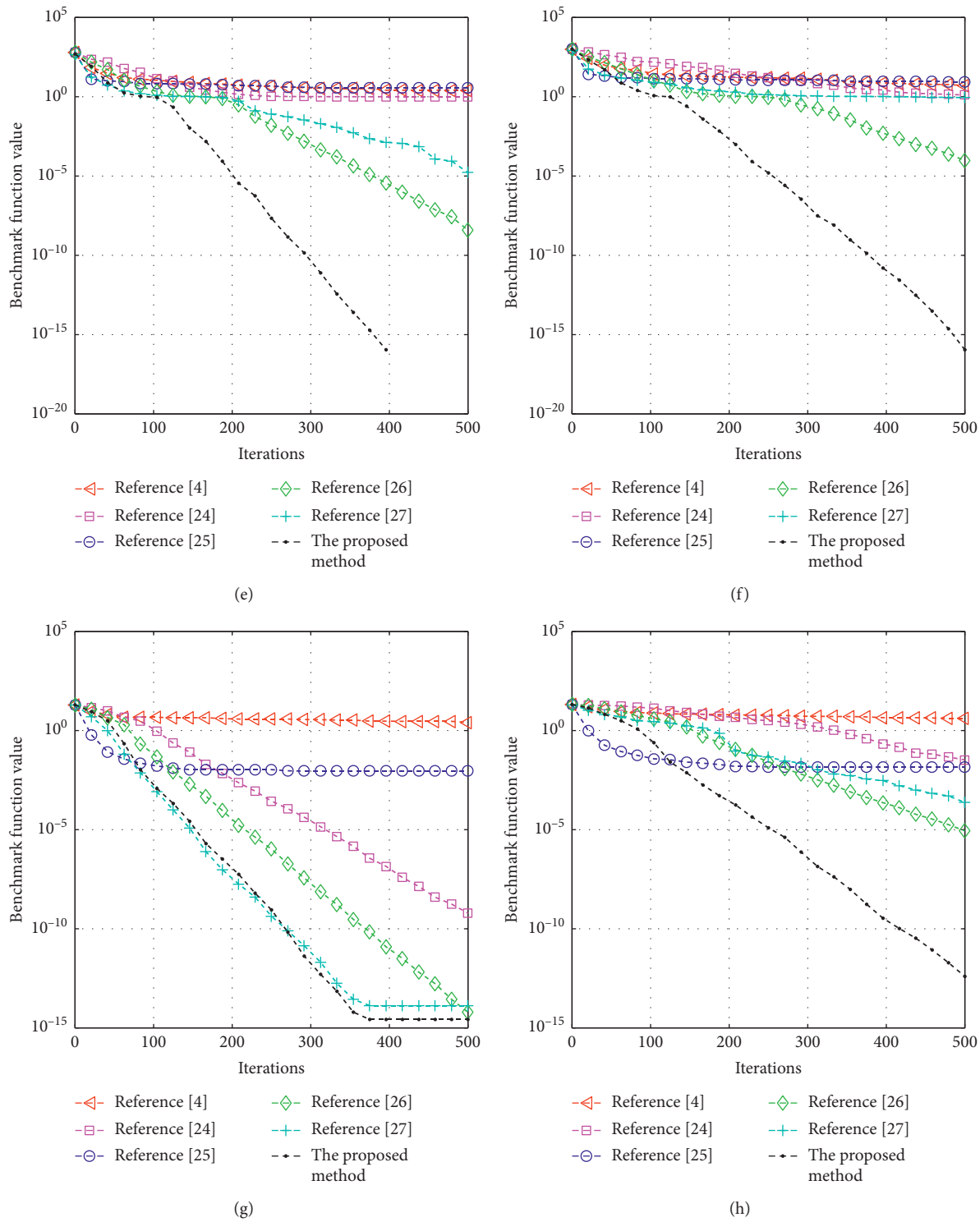

Figure 3: Continued.

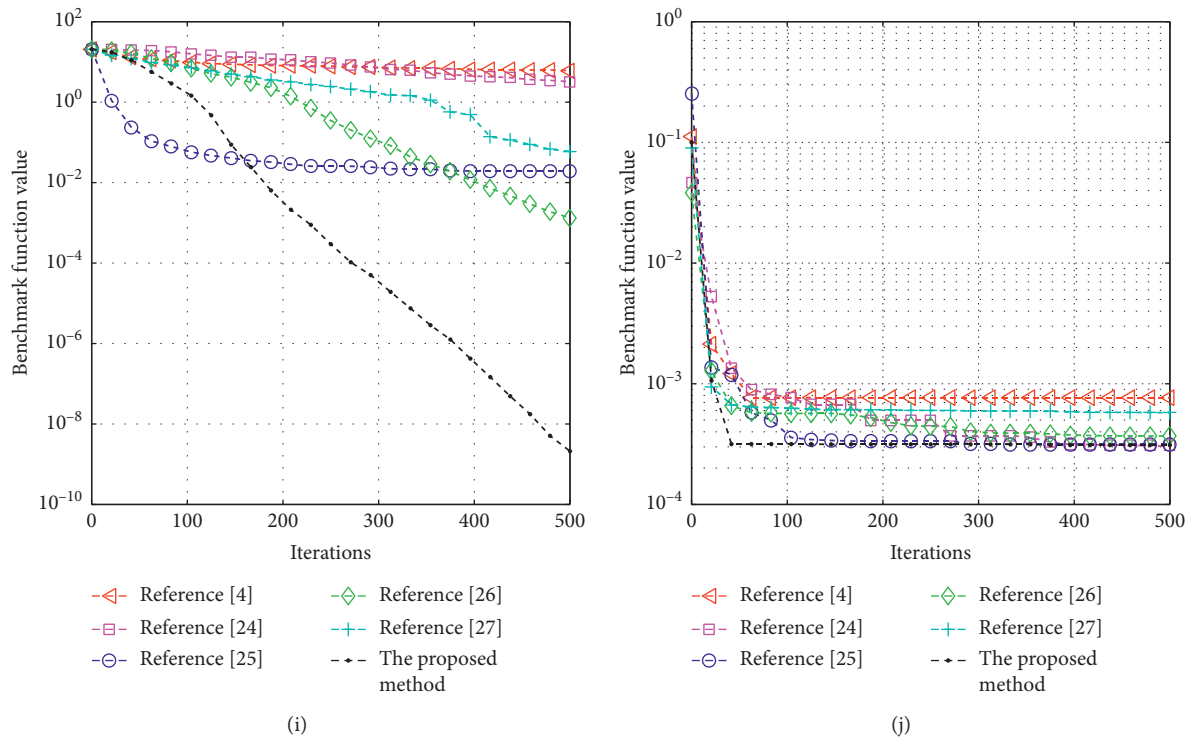

Figure 3: Multimodal function simulation results diagram, (a) f_5: Alpine $(d = 10)$, (b) f_5: Alpine $(d = 30)$, (c) f_5: Alpine $(d = 50)$, (d) f_6: Griewank $(d = 10)$, (e) f_6: Griewank $(d = 30)$, (f) f_6: Griewank $(d = 50)$, (g) f_7: Ackley $(d = 10)$, (h) f_7: Ackley $(d = 30)$, (i) f_7: Ackley $(d = 50)$, (j) f_8: Kowalik $(d = 4)$.

Table 5: Wilcoxon rank sum test results of unimodal function.

Function	Dimension	NBBO vs BBO		NBBO vs DE		NBBO vs EHO		NBBO vs DEBBO		NBBO vs BBOPSO	
		P	W	P	W	P	W	P	W	P	W
$f_1(x)$ Quartic	$D = 10$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$1.40e-03$	+	$1.68e-06$	+
	$D = 30$	$2.82e-11$	+	$2.82e-11$	+	$2.82e-11$	+	$2.82e-11$	+	$6.93e-11$	+
	$D = 50$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.82e-11$	+	$2.87e-11$	+
$f_2(x)$ Sphere	$D = 10$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.78e-11$	+
	$D = 30$	$2.87e-11$	+	$2.87e-11$	+	$2.82e-11$	+	$2.87e-11$	+	$2.87e-11$	+
	$D = 50$	$2.87e-11$	+	$2.87e-11$	+	$2.82e-11$	+	$2.87e-11$	+	$2.87e-11$	+
$f_3(x)$ Powell	$D = 10$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+
	$D = 30$	$2.87e-11$	+	$2.87e-11$	+	$2.93e-11$	+	$2.93e-11$	+	$2.93e-11$	+
	$D = 50$	$2.87e-11$	+	$2.87e-11$	+	$2.78e-11$	+	$2.87e-11$	+	$2.87e-11$	+
$f_4(x)$ Easom	$D = 2$	1.17e − 12	+	N/A	N/A	1.17e − 12	+	N/A	N/A	N/A	N/A

Table 6: Wilcoxon rank sum test results of multimodal function.

Function	Dimension	NBBO Vs BBO		NBBO vs DE		NBBO vs EHO		NBBO vs DEBBO		NBBO vs BBOPSO	
		P	W	P	W	P	W	P	W	P	W
$f_5(x)$ Alpine	$D = 10$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+
	$D = 30$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+
	$D = 50$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+
$f_6(x)$ Griewank	$D = 10$	$1.17e-12$	+	$1.69e-14$	+	$1.17e-12$	+	$5.62e-11$	+	$5.62e-11$	+
	$D = 30$	$1.17e-12$	+	$1.69e-14$	+	$1.17e-12$	+	$1.17e-12$	+	$1.17e-12$	+
	$D = 50$	$3.06e-12$	+	$3.06e-12$	+	$3.06e-12$	+	$3.06e-12$	+	$3.06e-12$	+
$f_7(x)$ Ackley	$D = 10$	$1.17e-12$	+	$1.17e-12$		$1.03e-13$	+	N/A	N/A	$1.15e-12$	+
	$D = 30$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+
	$D = 50$	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+	$2.87e-11$	+
$f_8(x)$ Kowalik	$D = 2$	$2.87e-11$	+	$4.27e-07$	+	$1.68e-06$	+	$2.31e-08$	+	$2.87e-11$	+

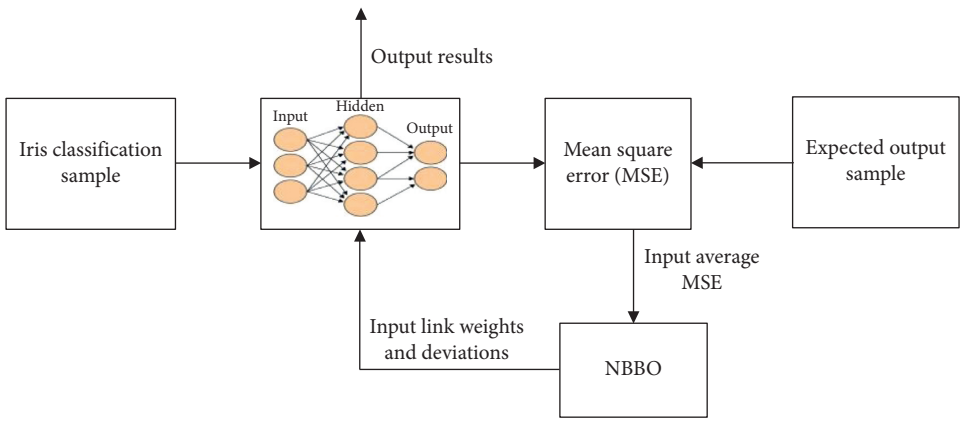

Figure 4: NBBO-based Iris classification diagram.

Table 7: Simulation results of iris classification.

Algorithm	Optimal value	Average value	Standard deviation
Reference [4]	3.15 $E-02$	5.34 $E-02$	1.52 $E-02$
Reference [24]	6.12 $E-02$	1.31 $E-01$	3.48 $E-02$
Reference [25]	2.37 $E-01$	3.92 $E-01$	7.36 $E-02$
Reference [26]	2.73 $E-02$	4.72 $E-02$	1.29 $E-02$
Reference [27]	1.51 $E-01$	2.56 $E-01$	5.34 $E-02$
The proposed method	1.34 E − 02	1.93 E − 02	3.80 E − 03

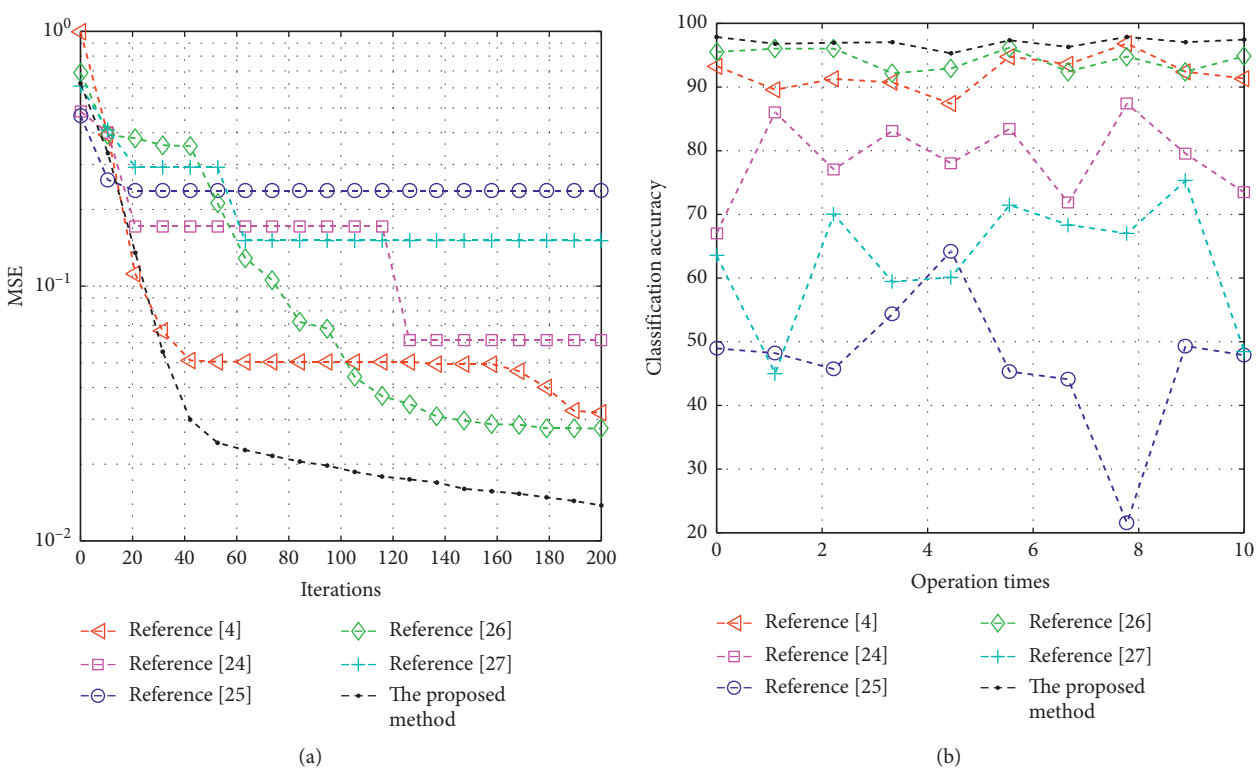

Figure 5: (a) Convergence curve and (b) classification accuracy.

through continuous iterative evolution, so as to provide the optimal connection weight and deviation for iris classification samples after iterative training.

The iris dataset has 150 training samples divided into 3 categories: Setosa, Versicolor, and Virginica, with a total of 4 basic features: sepal length, sepal width, petal length, and petal

width. We used the structure of multilayer perceptron as 4-9-3 to classify the problem. Table 7 shows the experimental results of the data set algorithm. Figure 5 shows the average convergence trend and classification accuracy.

Compared with other five iris classification algorithms in Table 7, we can see that the proposed NBBO has the better optimal value, average value, and standard deviation than the other five algorithms. The optimal and average values of the MSE calculated by NBBO are better than other algorithms, which indicates that the optimization performance of the NBBO is the best among the seven algorithms. At the same time, the minimum standard deviation calculated by NBBO also indicates that the NBBO algorithm inherits the strong robustness of the BBO algorithm and is more robust than other algorithms.

Figure 5(a) shows the classification convergence curves of the six algorithms for the iris classification problem, among which the convergence effect of the DE algorithm is the worst of the six algorithms, and the EHO and BBOPSO algorithms fall into the local optimum in the late iteration. BBO algorithm and DEBBO algorithm have similar convergence precision but the convergence speed is not ideal. The proposed NBBO algorithm is optimal in convergence precision and convergence speed. Figure 5(b) shows that DE, EHO, and BBOPSO show significant fluctuations in the classification accuracy of 10 independent operations. The NBBO algorithm has the best stability and average, and its classification accuracy is the highest among the three stable algorithms BBO, DEBBO, and NBBO.

4. Conclusions and Future Work

This paper proposed a novel biogeography-based optimization algorithm based on local migration strategy and nonuniform mutation for iris classification. Firstly, the hyperbolic cotangent transfer model was used to replace the original transfer model to enhance the adaptability of the algorithm. Secondly, the local transfer strategy was used to improve the global search ability of the algorithm. Finally, the nonuniform mutation operation was used to enhance the development ability of the algorithm in the later stage of iteration for the optimal solution in a small range, to improve the ability of the algorithm to exit the local optimum. The simulation results showed that the proposed NBBO has higher stability, better convergence accuracy, and faster convergence speed than other method for iris classification. How to use NBBO algorithm to solve other multiobjective optimization problem compare with multiobjective optimization method [30–33] and realize better applications for engineering practice will be the focus of our next work.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Natural Science Research Programme of Colleges and Universities of Anhui Province under grant KJ2020ZD39, the Foundation for Talented Young People of Anhui Polytechnic University under grant 2016BJRC008, the Open Research Fund of Anhui Key Laboratory of Detection Technology and Energy Saving Devices, and Anhui Polytechnic University under grant DTESD2020A02.

References

[1] S. A. C. Schuchkers, N. A. Schmid, A. Abhyankar et al., "On techniques for angle compensation in nonideal iris recognition," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 5, pp. 1083–4419, 2017.

[2] S. Gao, X. D. Zhu, and Y. N. Liu, "A quality assessment method of iris image based on support vector machine," *Journal of Fiber Bioengineering & Informatics*, vol. 8, no. 2, pp. 293–330, 2015.

[3] W. W. Boles and B. Boashash, "A human identification technique using images of the iris and wavelet transform," *Pattern Recognition*, vol. 36, no. 8, pp. 1783–1797, 2003.

[4] D. Simon, "Biogeography-based optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702–713, 2008.

[5] X. Zhang, D. Wang, and H. Chen, "Improved biogeography-based optimization algorithm and its application to clustering optimization and medical image segmentation," *IEEE Access*, vol. 7, pp. 28810–28825, 2019.

[6] F. Zhao, S. Qin, Y. Zhang, W. Ma, C. Zhang, and H. Song, "A hybrid biogeography-based optimization with variable neighborhood search mechanism for No-wait flow shop scheduling problem," *Expert Systems with Applications*, vol. 126, no. 6, pp. 321–339, 2019.

[7] M. Kaveh and M. S. Mesgari, "Improved biogeography-based optimization using migration process adjustment: an approach for location-allocation of ambulances," *Computers & Industrial Engineering*, vol. 135, pp. 800–813, 2019.

[8] B. X. Li and K. S. Low, "Low sampling rate online parameters monitoring of DC-DC converters for predictive-maintenance using biogeography-based optimization," *IEEE Transactions on Power Electronics*, vol. 31, no. 4, pp. 2870–2879, 2016.

[9] Y. B. Bhushan, S. U. Bhimrao, and R. D. Koteswara, "An efficient transient stability-constrained optimal power flow using biogeography-based algorithm," *International Transactions on Electrical Energy Systems*, vol. 28, no. 1, pp. 1–15, 2018.

[10] A. Kaveh, A. Dadras, and N. G. Malek, "Buckling load of laminated composite plates using three variants of the biogeography-based optimization algorithm," *Acta Mechanica*, vol. 229, no. 4, pp. 1551–1566, 2018.

[11] J. S. Wang and J. D. Song, "Chaotic biogeography-based optimization algorithm," *IAENG International Journal of Computer Ence*, vol. 22, no. 2, pp. 122–134, 2017.

[12] Q. X. Feng, S. Y. Liu, J. K. Zhang et al., "Improved biogeography-based optimization with random ring topology and Powell's method," *Applied Mathematical Modelling*, vol. 41, no. 6, pp. 630–649, 2017.

[13] H. Li and Y. H. Liu, "Beam pattern synthesis based on improved biogeography-based optimization for reducing

sidelobe level," *Computers and Electrical Engineering*, vol. 60, no. 5, pp. 161–174, 2017.

[14] X. Chen and K. J. Yu, "Hybridizing cuckoo search algorithm with biogeography-based optimization for estimating photovoltaic model parameters," *Solar Energy*, vol. 180, pp. 192–206, 2019.

[15] F. Zhao, S. Qin, Y. Zhang, W. Ma, C. Zhang, and H. Song, "A two-stage differential biogeography-based optimization algorithm and its performance analysis," *Expert Systems with Applications*, vol. 115, pp. 329–345, Jan, 2019.

[16] W. A. Guo, M. Chen, L. Wang et al., "A survey of biogeography-based optimization," *Neural Computing & Applications*, vol. 28, no. 8, pp. 1909–1926, 2017.

[17] A. R. AI-Roomi and M. E. EI-Hawary, "Metropolis biogeography-based optimization," *Information Sciences*, vol. 360, no. 5, pp. 73–95, 2016.

[18] H. P. Ma, "An analysis of the equilibrium of migration models for biogeography-based optimization," *Information Sciences*, vol. 180, no. 18, pp. 3444–3464, 2010.

[19] S. A. Darani and O. Abdelkhalik, "Convergence analysis of hidden genes genetic algorithms in space trajectory optimization," *Journal of Aerospace Information Systems*, vol. 15, no. 4, pp. 228–238, 2018.

[20] B. X. Yue, H. B. Liu, and A. Abraham, "Dynamic trajectory and convergence analysis of swarm algorithm," *Computing & Informatics*, vol. 31, no. 2, pp. 371–392, 2012.

[21] F. A. Attia, "Resolvent operators of Markov processes and their applications in the control of a finite dam," *Journal of Applied Probability*, vol. 26, no. 2, pp. 314–324, 1989.

[22] D. Ebenezer, "Optimum wavelet-based homomorphic medical image fusion using hybrid genetic–grey wolf optimization algorithm," *IEEE Sensors Journal*, vol. 18, no. 16, pp. 6804–6811, 2018.

[23] Q.-K. Pan, H.-Y. Sang, J.-H. Duan, and L. Gao, "An improved fruit fly optimization algorithm for continuous function optimization problems," *Knowledge-Based Systems*, vol. 62, pp. 69–83, 2014.

[24] R. Storn and K. Price, ""Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[25] G. G. Wang, L. Dos Santos Coelho, X. Z. Gao, and S. Deb, "A new metaheuristic optimisation algorithm motivated by elephant herding behaviour," *International Journal of Bio-Inspired Computation*, vol. 8, no. 6, pp. 394–409, 2016.

[26] W. Y. Gong, Z. H. Cai, and C. X. Ling, "DE/BBO: A hybrid differential evolution with biogeography-based optimization for global numerical optimization," *Soft Computing*, vol. 15, no. 4, pp. 645–665, 2010.

[27] D. Li, W. Guo, L. Wang et al., "Particle swarm optimization-based solution updating strategy for biogeography-based optimization," in *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, July 2016.

[28] D. Joaquín, G. Salvador, D. Molina et al., "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm & Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[29] H. Mannila, "Data mining: machine learning, statistics, and databases," in *Proceedings of 8th International Conference on Scientific and Statistical Data Base Management*, June 1999.

[30] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 474–494, 2013.

[31] R. Wang, Z. Zhou, H. Ishibuchi, T. Liao, and T. Zhang, "Localized weighted sum method for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 3–18, 2018.

[32] R. Wang, Q. Zhang, and T. Zhang, "Decomposition-based algorithms using pareto adaptive scalarizing methods," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 6, pp. 821–837, 2016.

[33] K. Li, T. Zhang, and R. Wang, "Deep reinforcement learning for multiobjective optimization," *IEEE Transactions on Cybernetics*, p. 1. In press, 2020.