A novel improved butterfly optimization algorithm based on the adaptive inertia weight and the dynamic switching probability

Abstract: Butterfly optimization algorithm (BOA)is widely applied in various complex problems because of its efficient performance. However, similar to other swarm intelligent optimization algorithms, the BOA also has the problem of easily falling into the local optimum. In order tosolve the above problem, this paper proposes an improved BOA (IBOA) by balancing local search and global search.In the IBOA,the adaptive inertia weight and the dynamic switching probability are introduced, which enables the IBOA to significantly improve the exploration ability in the early stage and the exploitation ability in the later stage, so that the algorithm is able to effectively escape fromthe local optimum. The performance of the IBOA is verified on eight benchmark functions. Compared with the other four optimization algorithms, the IBOA has the strongest optimization ability. Finally, a new prediction model is proposed based on that the IBOA is used to determine the hyper-parameters of support vector machine (SVM), which is called the IBOA-SVM model.Prediction experiments are carried out in earthquake magnitude dataset and air quality index dataset to validate the performance of the IBOA-SVM. The comparative experimental results indicate that our proposed model has higher prediction accuracy.

Keywords: Butterfly optimization algorithm, Adaptive inertia weight, Dynamic switching probability, Support vector machine

1 Introduction

Inspired by nature, many bionic intelligent optimization algorithms are proposed, such as particle swarm optimization (PSO) [1], grey wolf optimizer (GWO) [2], grasshopper optimization algorithm (GOA) [3], genetic algorithm (GA) [4], moth-flame optimization (MFO) [5], sine cosine algorithm (SCA) [6], ant colony algorithm (ACA) [7], artificial bee colony (ABC) [8], salp swarm algorithm (SSA) [9], and butterfly optimization algorithm (BOA) [10]. The BOA is originally proposed by Arora and Singh [10]. The algorithm mimics the way butterflies search for food in nature. Butterflies can sense the smell of flowers. Unlike human beings, whose sense of smell depends on the nose, butterflies' sense of smell comes from tentacles. When the flower fragrance irritates the tentacle, the butterfly will move to the honey source to find food. The BOAsearches the global optimal solution according to this characteristic. However, like other swarm intelligent optimization algorithms, the BOA also has two possible problems: slow convergence and susceptibility to fall into the local optimum. Therefore, many researchershave improved and promoted the BOA.

Sharma et al. [11] proposed a variant of theBOA embedded with lagrange interpolation called mlBOA. By introducing adaptive parameters, lagrange interpolation formula, and levy flight, the exploration and exploitation capabilities of the algorithm are balanced. The fragrance generation scheme of the BOA is improved to further improve the performance of the algorithm. The superiority of the mlBOA is verified in experiments. Bendahmane et al. [12] usedthe crossover operator to propose animproved version of the BOA called xBOA. The crossover operator is embedded in the global exploitation phase of the BOA, enabling the algorithm toexplore several areas in the solution space at the same time and find a more accurate solution. The xBOA is used to explore the unknown area of single robot and multi-robot. The results indicate that the xBOA has stronger robustness and higher

exploration rate. In order to effectively prevent the algorithm from falling into local optimum,Long et al. [13]combined the dynamic inertia weight and the opposition learning strategy to solve this problem. The effectiveness of the improved algorithm is verified in several benchmark functions and datasets. Xia et al. [14] proposed a double-mutantBOA called DMBOA. They introduce the dynamic switching probability, which gradually converges from 0.8 to 0.5.Itis beneficial for the algorithmto achieve the balance between the global and local search. In addition, the update equation of the algorithm is modified, and the population reconstruction mechanism is established to improve the ability of the BOA to escape from the local optimum. The effectiveness of the DMBOA is verified on 12 classical benchmark functions, and the DMBOA shows better performance in blind source separation of signals. Zhang et al. [15] put forward an improvement in the update equation of the BOA, and introduced an operator named color perception rule to make the search behavior of the butterfly more in line with objective facts. In addition, the control parameters are updated by chaotic mapping, and the hybrid-flash BOA (HFBOA) is proposed. The processing results of benchmark functions and practical problems demonstrate that the HFBOA has better performance. In order to balance the search of the BOA, Zhou et al. [16] introducedgaussian distribution estimation strategy into the global update equation, and the new variantis called GDEBOA. The comparative experimental results indicate that the GDEBOA has the obvious effectiveness. The GDEBOAis also used to deal with the path planning problem.The simulation results show the superiority of the GDEBOA. Mortazavi et al. [17] proposed a variant of the BOA called FBOA that combines fuzzy decision strategy and virtual butterfly. The comparative experimental results on 21 benchmark functions indicate that the FBOA algorithm has the highest accuracy. Some of the more advanced improvements can be found in [18-22].

The BOA is widely used to optimize the parameters of complex models because of its stable performance, simple operation, and fast convergence. Jin et al. [23] used the improved version of BOA to find the optimal parameters of the depth belief network (DBN), and proposed an IBOA-DBN diagnosis model. In the original BOA, tent mapping is introduced to promote the global optimization ability. In addition, the adaptive inertia weight is added to the global search update equation, which greatly improves the search performance of the algorithm. The experimental results indicate that the IBOA-DBN model has higher recognition accuracy. Tang et al. [24] used an improved BOA to search the optimal hyper-parameters of extreme random forest (ERF) and proposed a fault detection model, which is called IBOA-ERF model. Here are four improvements to improve the BOA. First, chaotic mapping is used to generate the initial population. Second, an adaptive inertia weight is added in the iterative phase to modify the update equation. Third, the pigeon-inspired optimization (PIO) algorithm is used to further modify the local update equation. Fourth, the dynamic switching probability is introduced to achieve a balance between the global and local search. The experimental results indicate that the above improvements greatly enhance the optimization ability of the BOA, and the improved algorithm also shows satisfactory results in finding the optimalparameters of the ERF. Long et al. [25] combined enhanced adaptive BOA (EABOA) to propose a photovoltaic model. They added an adaptive guiding factor to the update equation of the BOA to adjust the exploration and exploitation performance. Finally, the unknown parameters of three PV models are optimized by the EABOA and the other compared algorithm, and the results indicate that the EABOA has superior optimization ability. In order to obtain more accurate structural identification results, Zhou et al. [26] introduced the improved BOA into structural identification. Firstly, chaotic sequence is introduced to optimize the iterative process. The randomness of chaotic sequence can prevent the algorithm from premature convergence. Secondly, the clustering competition learning mechanism is used to renew the population

again. The comparative experimental results indicate that the improved BOA has higher accuracy of actual structural identification.

This paper proposes a novel improved butterfly optimization algorithm called IBOA. We introduce the adaptive inertia weight and the dynamic switching probability to achieve a better balance between the global and local search. The effectiveness of the IBOA is verified on eight benchmark functions. Meanwhile, we use the IBOA to optimizethe hyper-parameters of the support vector machine (SVM), and propose a novelprediction model called IBOA-SVM model. The strongeroptimization capability of the novel model is verified in earthquake magnitude dataset and air quality index dataset.

The rest structure of this article is arranged as follows. The traditional BOA is presented in section 2. In section 3, we introduce the adaptive inertia weight and the dynamic switching probability and propose the IBOA. We discuss the performance of the IBOA in section 4. In section 5, we propose the IBOA-SVM and use the proposed IBOA-SVM to complete prediction tasks on two real datasets. We analyze the experimental data and elaborate on the details of the experiments. The relevant experimental results are analyzed and studied. In the final section of this paper, the significantconclusions are summarized.

2 Butterfly optimization algorithm

The BOA assumes that butterflies can not only perceive fragrance, but also produce fragrance. This fragrance spreads in the air and can be perceived by other butterflies. This forms a network of information exchange within the space. Butterflies can communicate information with each other. When butterflies perceive the information spread by the butterfly with the highest fragrance concentration, they will move to it. This behavior is called global search. When butterflies cannot receive the scent information sent by other butterflies, they choose to wander randomly. This behavior is known as local search. During the iterativeprocess, each butterfly represents a potential solution and updates its coordinates based on both the current solution and the fragrance concentration. The butterfly uses a switching probability p to decide whether to move towards the optimal solution or randomly.

In the BOA, the butterfly is stimulated to sense and produce a fragrance. So the fragrance concentration of the butterfly is an expression that is affected by the stimulation intensity. The expression of the fragrance concentration is as follows:

$$f = cI^a, \quad (1)$$

where f is the fragrance concentration, I is the stimulus intensity, a is the power exponentand c is the sensory modality. The value of c is updated according to the following equation [27] :

$$c^{t+1} = c^t + \frac{0.025}{c^t \cdot T}, \quad (2)$$

where T is the maximum number of iterations.

In the global search phase, the search operator is close to the optimal solution, which can be represented by the following update equation:

$$x_i^{t+1} = x_i^t + \left(r^2 \cdot g^* - x_i^t\right) f_i, \quad (3)$$

where x_i^t is the solution vector for the i th butterfly in the iteration number t and g^* denotes the optimal solution in the current iteration. f_i denotes the fragrance concentration of the i th butterfly. r denotes a random number between [0,1].

During local searchprocess, the butterfly operator can't receive the fragrance from the surrounding butterflies, so it chooses to wander randomly, which isexpressed as the following update equation:

$$x_i^{t+1} = x_i^t + \left(r^2 \cdot x_j^t - x_k^t\right) f_i , \tag{4}$$

where x_j^t and x_k^t are the j th and k th butterflies. r denotes the random number between [0,1] .

The flow-process diagram of the BOA is presented in Figure 1.

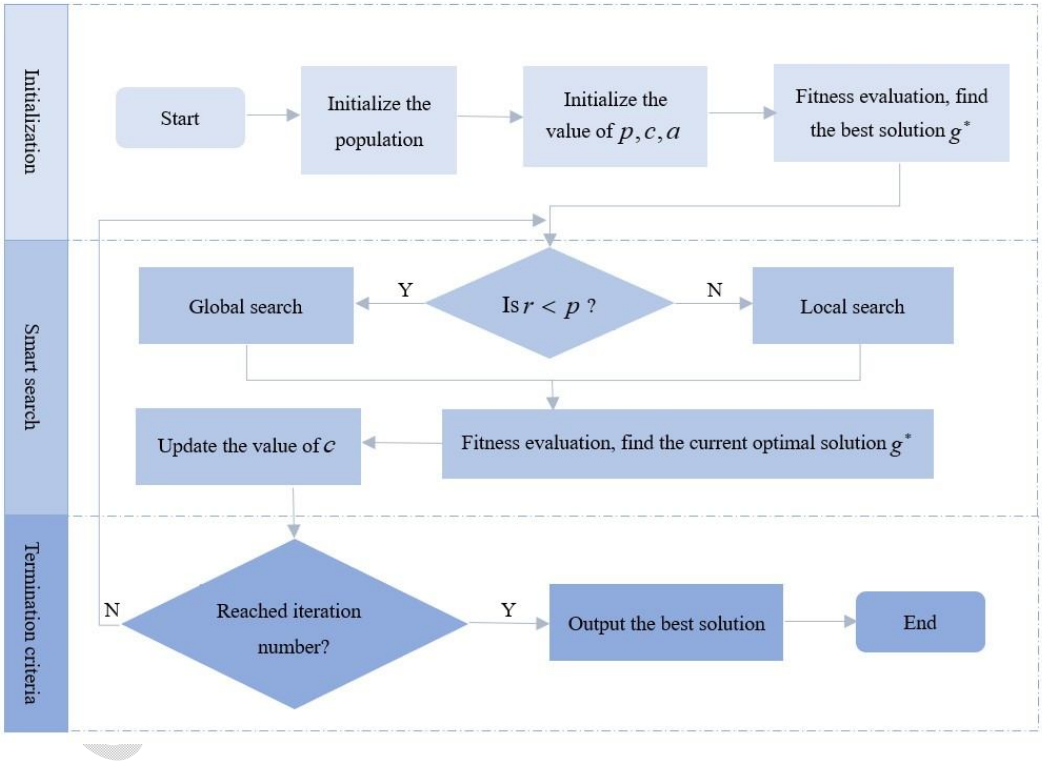


Figure 1 Flow-process diagram of the BOA

3 Improved butterfly optimization algorithm

It is difficult for the traditionalBOA to escapefrom the local optimum. The research shows that effectively balancing the local search and global search can improve the performance of the BOA and effectively solve the optimization problems [28]. This section mainly introduces two improved methods.First, we introducean adaptive inertia weight ω . Second, we introducethe dynamic switching probability p to get a better balance between the local search and global search.

3.1 Adaptive inertia weight

In this section, an adaptive inertia weight ω is proposed. ω is expressed as a function of the number of iterations t. Our goal is to decrease the value of ω as the increase of t.As the value of t increases from 0, the function value decreases from 1 to 0. When the value of ω is large, more butterflies choose local search. When it is small, the algorithm tends to search globally. This enables the algorithm to exhibit high exploration ability in the early stage, allowing it to explore more unknown values in the solution space and thereby increasing the diversity of solutions. The algorithm can quickly move towards the optimal solutionin the later stage. Therefore, achieving a good balance between the global and local search caneffectivelyenhance the optimization performance of the BOA. The expression of ω is as follows:

$$\omega = 1 - \frac{1}{1 + e^{-\frac{t-100}{10}}}, \tag{5}$$

where ω is the adaptive inertia weight and t is the number of iterations.

After introducing ω, the global search update equation is:

$$x_i^{t+1} = \omega \cdot x_i^t + \left(r^2 \cdot g^* - x_i^t \right) f_i . \tag{6}$$

The local search update equation is:

$$x_i^{t+1} = \omega \cdot x_i^t + \left(r^2 \cdot x_j^t - x_k^t \right) f_i . \tag{7}$$

3.2 Dynamic switching probability

In the traditionalBOA, the butterfly operator chooses global search or local search with a fixed probability. The probability is usually set to 0.8, which limits the search behavior of butterfly operators to some extent. This probability causes butterfly operators tend to search globally at any stage of the iterative process, and only a few butterfly operators select local search.Consequently, the BOA is difficult to avoid falling into the local optimum. To obtain a better balance between the global and local search, the paper introducesthe dynamic switching probability p which is expressed as a function of t. The value of p is set to a small value in the early stage, which makes the algorithm tend to local search. As the iteration progresses,the value of p gradually approaches 0.8, thereby biasing the algorithm towards global search. The introduction of p makes the algorithm achieve a relative balance between these two searches, which not only increases the diversity of candidate solutions, but also promotes the convergence speed. This effectively prevents the algorithm from falling into the local optimum.

The mathematical expression of the dynamic switching probability is:

$$p = 0.8 \cdot \frac{t}{t+1} . \tag{8}$$

The flow-process diagram of the IBOA is presented in Figure 2.

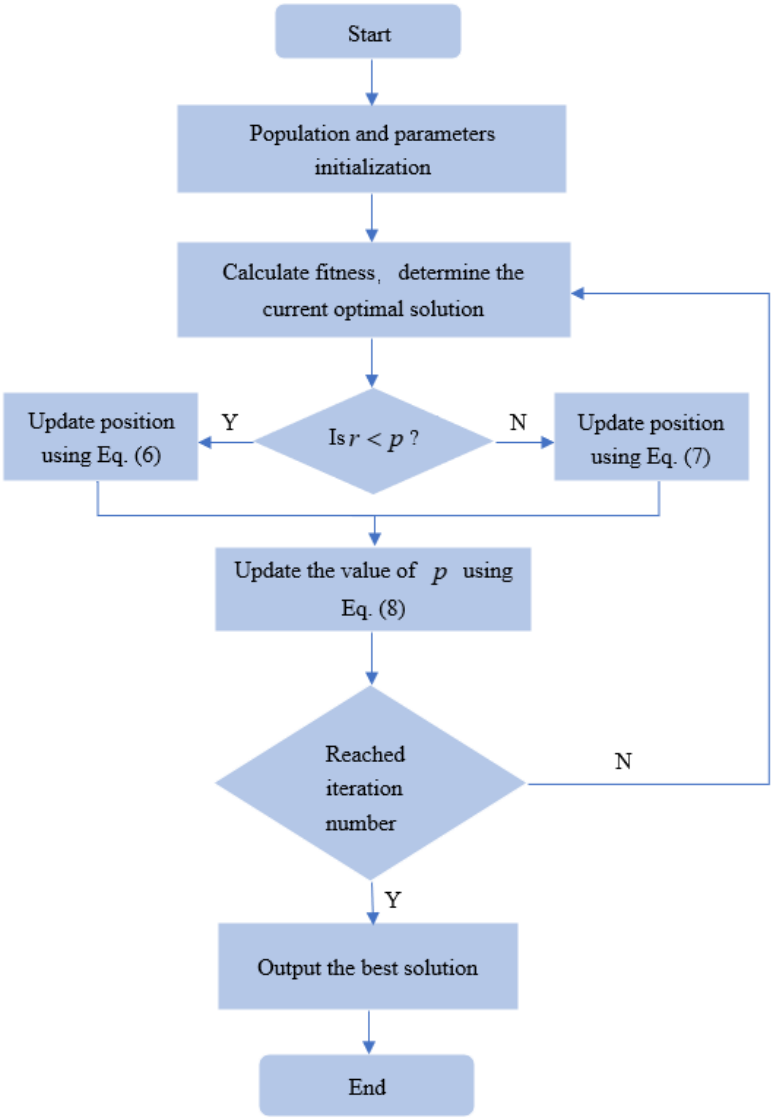


Figure 2 Flow-process diagram of the IBOA

4 Experimental results and discussion

This section presents the simulation results of the IBOA on eight benchmark functions.We comparethe IBOA with the traditional BOA, GWO, SCA, and PSO algorithms and givean analysis of the simulation results. In this experiment, we set that the number of initial population N is 20, the maximum number of iterations T is 200,and the initial value of p is 0. We use the relevant literature about the BOA to select the initial value of c and a , and the result is that c is 0.01, a is 0.1. The eight benchmark functions are listed in Table 1. Table 2 gives the simulation results of the above five algorithms on these functions. Figure 3 shows the convergence curves of five algorithms on the eight benchmark functions respectively.

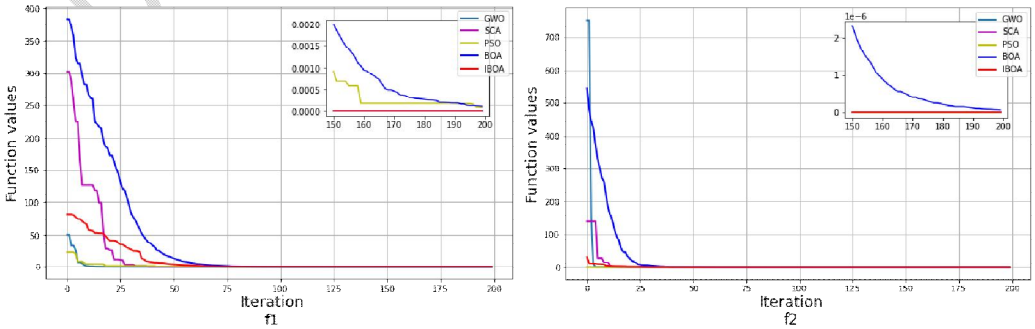
Table 1 Standard benchmark functions used in this study

No.	Equation	Dim	Range	Optima
f_1	$f(x)=\left(x_1^2+x_2^2\right)^{0.25}\cdot\left[\left(50\left(x_1^2+x_2^2\right)^{0.1}+1\right)\right]$	2	[-100,100]	0
f_2	$f(x)=x_1^2+2x_2^2-0.3\cos(3\pi x_1)-0.4\cos(4\pi x_2)+0.7$	2	[-100,100]	0
f_3	$f(x)=\frac{\sin^2\sqrt{x_1^2+x_2^2}-0.5}{\left[1+0.001\left(x_1^2+x_2^2\right)\right]^2}+0.5$	2	[-100,100]	0
f_4	$f(x)=\sum_{i=1}^{n}\left(\sum_{j=1}^{i}x_j\right)^2$	30	[-10,10]	0
f_5	$f(x)=\max\left(\lvert x_i\rvert,1\le i\le n\right)$	30	[-10,10]	0
f_6	$f(x)=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	30	[-10,10]	0
f_7	$f(x)=\sum_{i=1}^{n}x_i^2$	30	[-100,100]	0
f_8	$f(x)=x_1^2+10^6\sum_{i=2}^{n}x_i^2$	30	[-100,100]	0

Table 2 Simulation results of different algorithms

No.	GWO	SCA	PSO	BOA	IBOA
f_1	7.81E-08	3.45E-07	1.28E-04	8.48E-05	**0.00E+00**
f_2	0.00E+00	0.00E+00	1.11E-16	6.42E-07	**0.00E+00**
f_3	9.72E-03	4.84E-03	5.55E-17	3.74E-02	**0.00E+00**
f_4	1.69E-164	9.51E-56	1.00E-21	1.61E-15	**0.00E+00**
f_5	7.06E-08	5.87E+00	4.10E-01	1.15E-05	**1.17E-224**
f_6	4.41E-66	5.82E-23	9.12E-11	1.66E-09	**2.04E-227**
f_7	2.35E-193	1.88E-44	5.36E-24	1.10E-14	**0.00E+00**
f_8	4.75E-27	2.71E-27	2.31E-11	5.53E-08	**0.00E+00**

As shown in Table 2, the IBOA obtains better solutions on each benchmark function and demonstrates significantly higher accuracy.The search results of IBOA reach the theoretical optima for six of these benchmark functions. Only the GWO and SCA find the theoretical optimum of the function f_2, and for other benchmark functions, none of these four compared algorithms find the theoretical optima.The resultsdemonstrate the superior search capability of the IBOA. According to the comparative experimental results between the IBOA and the BOA, it is easy tofind that the adaptive inertia weight and the dynamic switching probability greatly enhance the search performance of the BOA. These two improvements can effectively balance thesetwo search processes of the BOA and achieve our expected results.In addition, these two improvements greatly enhance the ability of the algorithm to escape fromthe local optimum. From the overall simulation results of these eight benchmark functions, the IBOA gets the smallest optimal solution, which indicates that the IBOA not only has superior optimization ability, but also strong robustness.



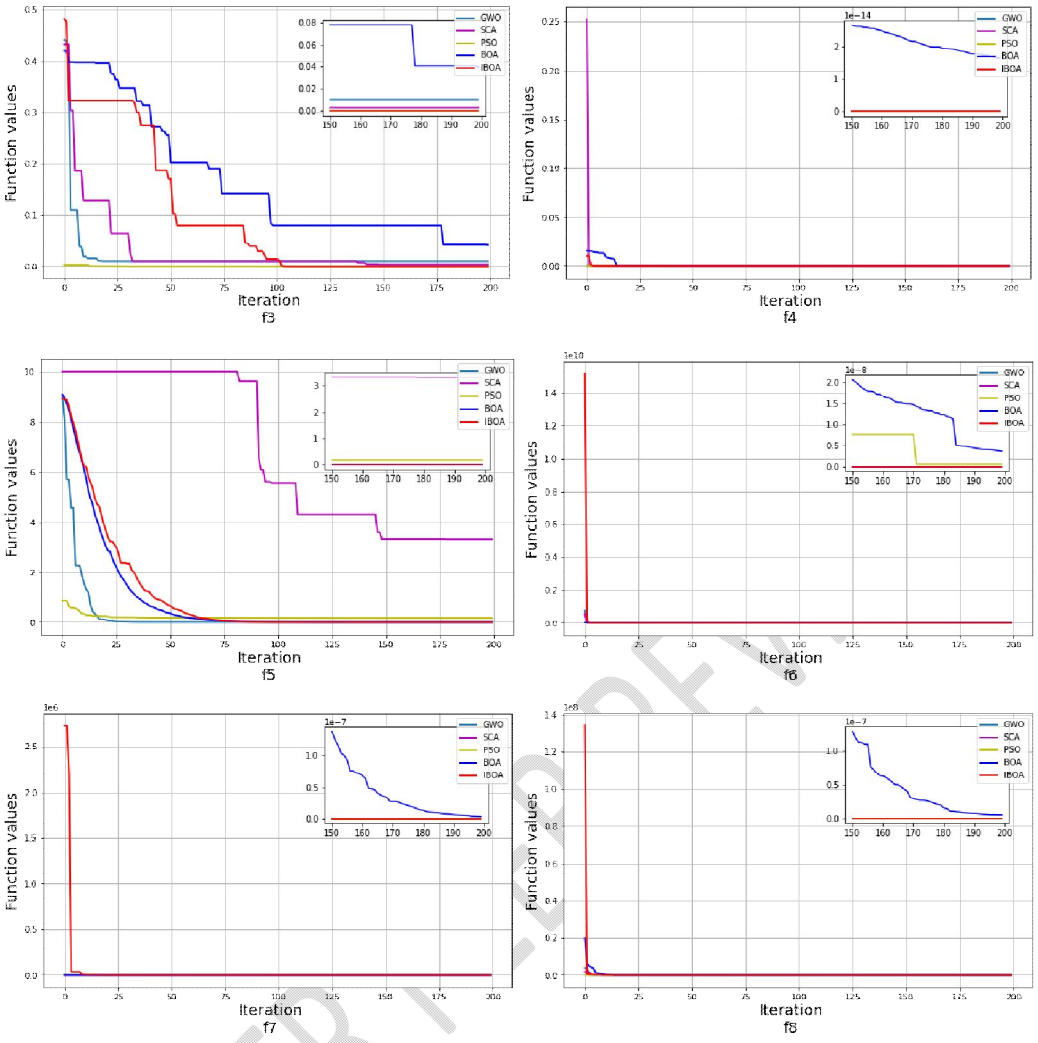


Figure 3 Convergence curves of different algorithms

It is easy to find fromFigure 3 that the optimal solution obtained by the IBOA is closer to the theoretical optima than the other compared algorithms.This result indicates that the IBOA has the highest accuracy. In addition, by observing the convergence curves, we know that the IBOA has a fast convergence speed except a few functions,and for each benchmark function, the IBOA obtains the optimal results, which fully demonstrates the powerful search capability of the IBOA. We can draw a conclusion that the above two improvements play a great role in helping the IBOAescape fromthe local optimum. The IBOA can effectively balance local search and global search, and it outperforms the compared methods in terms of reliability and accuracy.

5 Application of the IBOA-SVM model

5.1 IBOA-SVM model

SVM is a prediction method proposed by Cortes and Vapnik [29], which has high prediction accuracy for most classification and regression problems in low-dimensional space. For the classification problem, the purpose of the SVM is to find a hyperplane, which maximizes the distance between different classes.This means ensuring that the sample points closest to the hyperplane are as

far away from it as possible. In regression problems, we aim to obtain a special hyperplane that minimizes the error between the predicted value and the actual data point, either by fitting the points as closely as possible or having them fall on the hyperplane. The processing model of regression problems is called support vector regression (SVR). The prediction results of the SVM model are often acceptable for linearly separable problems. However, in cases where sample points are linearly inseparable, the prediction results may exhibit significant errors. To address linearly inseparable problems, a kernel function is often introduced to map the data from low-dimensional to high-dimensional space, thereby enhancing the prediction accuracy of the model. The three kernel functions commonly used are as follows:

Polynomial kernel:

$$K_1\left(x_i, x_j\right) = \left(\eta x_i \cdot x_j + \delta\right)^d, \tag{9}$$

Gaussian kernel:

$$K_2\left(x_i, x_j\right) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right), \tag{10}$$

Sigmoid kernel:

$$K_3\left(x_i, x_j\right) = \tanh\left(\eta x_i \cdot x_j + \delta\right), \tag{11}$$

where x_i and x_j denote the sample data, η, δ, d, σ represent parameters in different kernel function.

In this paper, the SVR model is used to deal with regression problems, and the modeling process of the SVR model is introduced below. Assuming a given data set $\{(x_i, y_i)\}_1^n$, (x_i, y_i) as a sample point, SVR is to find a hyperplane that minimizes the distance between the farthest sample point and the plane. The function of the SVR is expressed as:

$$f(x) = w^T \phi(x) + b, \tag{12}$$

where ϕ is a nonlinear mapping function, w is the weight coefficients, and b is a constant. The SVR model is transformed into solving the following optimization problems:

$$Min \frac{1}{2}\left\| w^2 \right\|, \tag{13}$$

$$s.t.: \begin{cases} y_i - w^T \phi(x_i) - b \le \varepsilon \\ y_i - w^T \phi(x_i) - b \ge -\varepsilon \end{cases}, \tag{14}$$

To make the prediction of the SVR model more stable, the slack variables ξ, ξ^* are introduced to optimize the SVR model. The revised optimization problem is expressed as follows:

$$Min \frac{1}{2}\left\| w^2 \right\| + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^*\right), \tag{15}$$

$$s.t.:\begin{cases} y_i - w^T\phi(x_i) - b \le \varepsilon + \xi_i \\ y_i - w^T\phi(x_i) - b \ge -\varepsilon - \xi_i^* \\ \xi_i, \xi_i^* \ge 0, i = 1,2,...n \end{cases}, \tag{16}$$

where C is called the penalty factor, ε is the insensitive loss parameter, and ξ, ξ^* are the slack variables.

Using Lagrange multiplier method and optimal conditions, we find the solution of the above problem by solving its dual problem, and finally obtain the following expression of $f(x)$:

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(x_i, x) + b, \tag{17}$$

where α_i, α_i^* is the lagrange multiplier, $K(x_i, x)$ is the kernel function, and here we use Gaussian kernel.

The flow-process diagramof the IBOA-SVM model is shown in Figure 4.

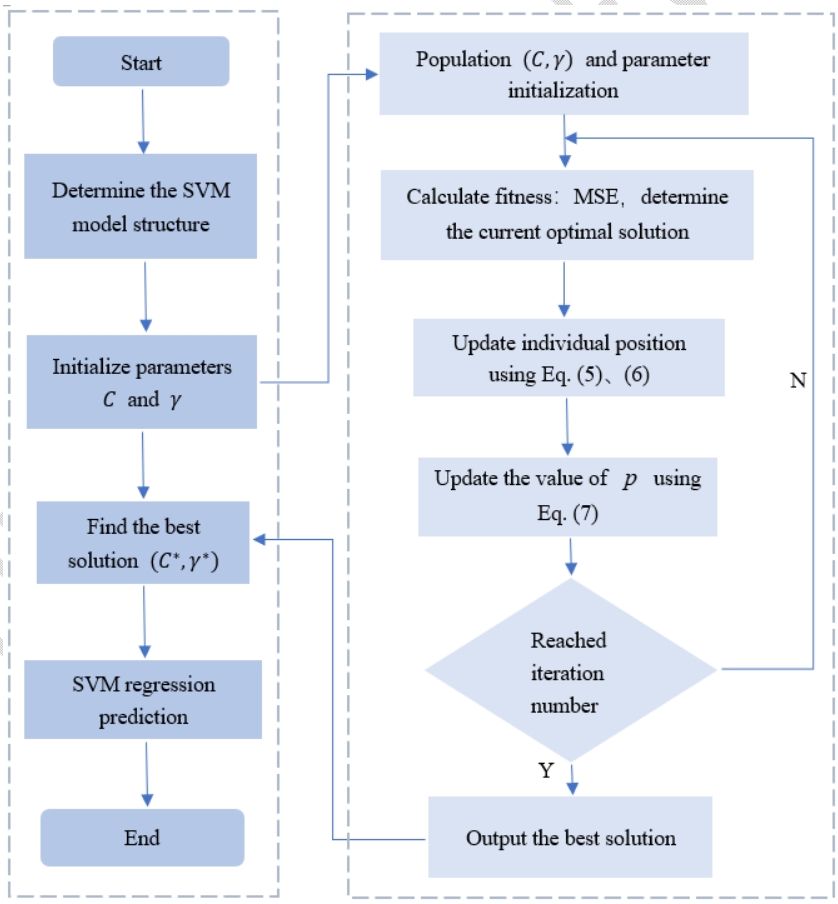


Figure 4 Flow-process diagram of the IBOA-SVM model

5.2 Earthquake magnitude prediction

5.2.1 Earthquake magnitude data

We select seventeen earthquake data in southwest Yunnan, China as the sample data. The data has eight indicators, which are earthquake cumulative frequency, cumulative released energy, *b*-value, abnormal earthquake swarm count, earthquake belt count, activity cycle, correlation zone magnitude, magnitude, respectively. Where, the earthquake cumulative frequency refers to the number of earthquakes with magnitude 3 or above occurring in half a year, the cumulative released energy refers to the accumulated value of energy released in half a year, and the value of *b* is the proportional coefficient in the relationship between magnitude and frequency. The index values of the activity cycle are 1 and 0, indicating whether the regional fault zone is in the activity cycle.The first seven indicators comprehensively reflect the characteristics of earthquake in time, space and intensity, and play a key role in predicting earthquake magnitude. This data is used to verify the earthquake magnitudeprediction performanceof the IBOA-SVM model. In this experiment, we choose70% of the sample data as the training dataset and 30% as the test dataset. The experiments are carried out in the Python3.9.12 environment.

5.2.2 Results of earthquake magnitude evaluation

We choose the mean square error (MSE) and the mean absolute error (MAE) as our evaluation metrics. Lower MSE and MAE values indicate a better alignment between predicted and actual values. Table 3presents the average values of MSE and MAE obtained by running each algorithm 10 times on earthquake magnitude data. The mathematical expressions of MSE and MAE are as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_i\right)^2, \tag{18}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - f_i\right|, \tag{19}$$

where y_i denotes the actual value and f_i denotesthe predicted value, and N denotes the total number of data points.

Table 3 MSE and MAE of different prediction models on earthquake magnitude data

	SVM	GWO-SVM	PSO-SVM	BOA-SVM	IBOA-SVM
MSE	0.28	0.22	0.39	0.40	**0.14**
MAE	0.36	0.36	0.44	0.46	**0.32**

As seen from Table 3, thevalues of MSE and MAE obtained by the IBOA-SVM model are 0.14 and 0.32, respectively,which are the smallest compared with the values obtained by the other four models. There are sufficient reasons to indicate that our proposed IBOA-SVM model has better prediction performance.At the same time, it also demonstrates that the optimization capability of the BOA is greatly enhanced after the adaptive inertia weight and the dynamic switching probabilitycorrection.

To further illustrate the significant efficacy of the IBOA-SVM model, we present a comparative visualization in Figure 5, depicting the actual and predicted earthquake magnitudes.In this visualization, the solid line denotes the actual data and the dashed line denotes the predicted value. By comparing the

predicted value of the five models, it can be seen that the closer alignment of the IBOA-SVM model's predictions to the actual values, which shows that the IBOA-SVMmodel has superior prediction performance.

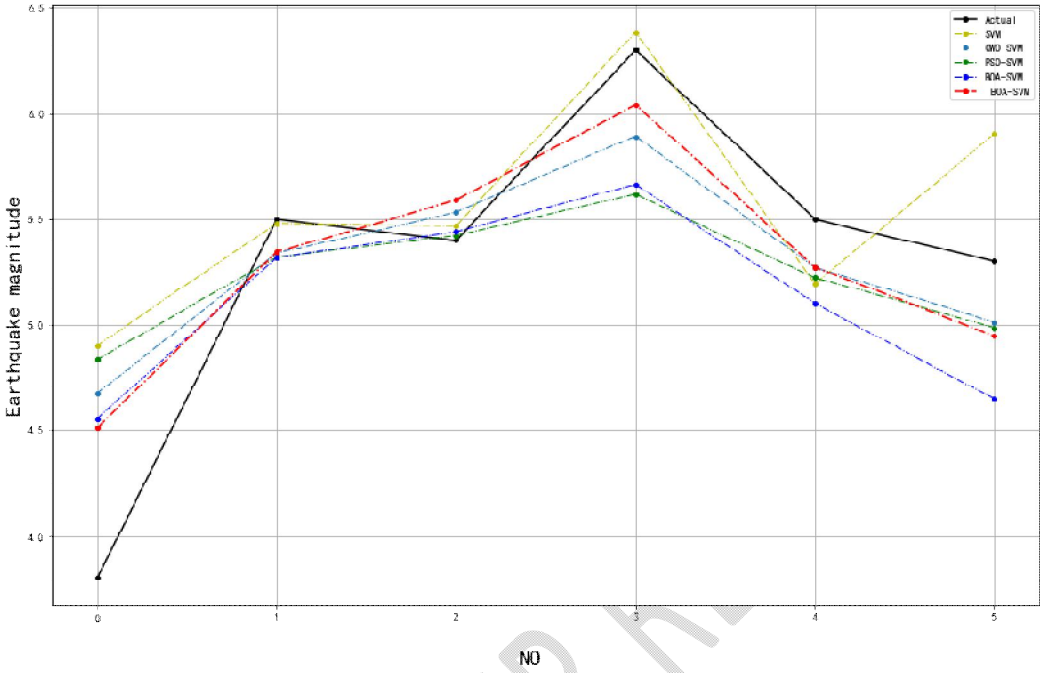


Figure 5 Comparison results of different prediction models on earthquake magnitude data

5.3 Air quality index (AQI) prediction

5.3.1 AQI data

In this paper, we select the monthly AQI and monthly average mass concentration data of six pollutant gases in Chengdu from December 2013 to June 2023 as samples to establish the database. The database contains the monthly average values of $PM_{2.5}$ ($\mu g/m^3$), PM_{10} ($\mu g/m^3$), SO_2 ($\mu g/m^3$), NO_2 ($\mu g/m^3$), CO (mg/m^3), O_3 ($\mu g/m^3$), and AQI. The data is obtained from the China Air Quality Online Monitoring and Analyzing Platform. $PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, and O_3 are chosen as major factors affecting the AQI. In this experiment, we use the data from December 2013 to June 2020 for training, and the data from July 2020 to June 2023 for testing the model's predictive performance.

5.3.2 Results of the AQI evaluation

The values of MSE and MAE are also selected as the test criteria, and the above five models are used to predict the AQI. The experimental results are given in Table 4 and Figure 6.

Table 4 MSE and MAE of different prediction models on AQI data

	SVM	GWO-SVM	PSO-SVM	BOA-SVM	IBOA-SVM
MSE	428.07	409.97	353.40	113.24	**58.06**
MAE	13.12	12.22	10.76	5.43	**4.35**

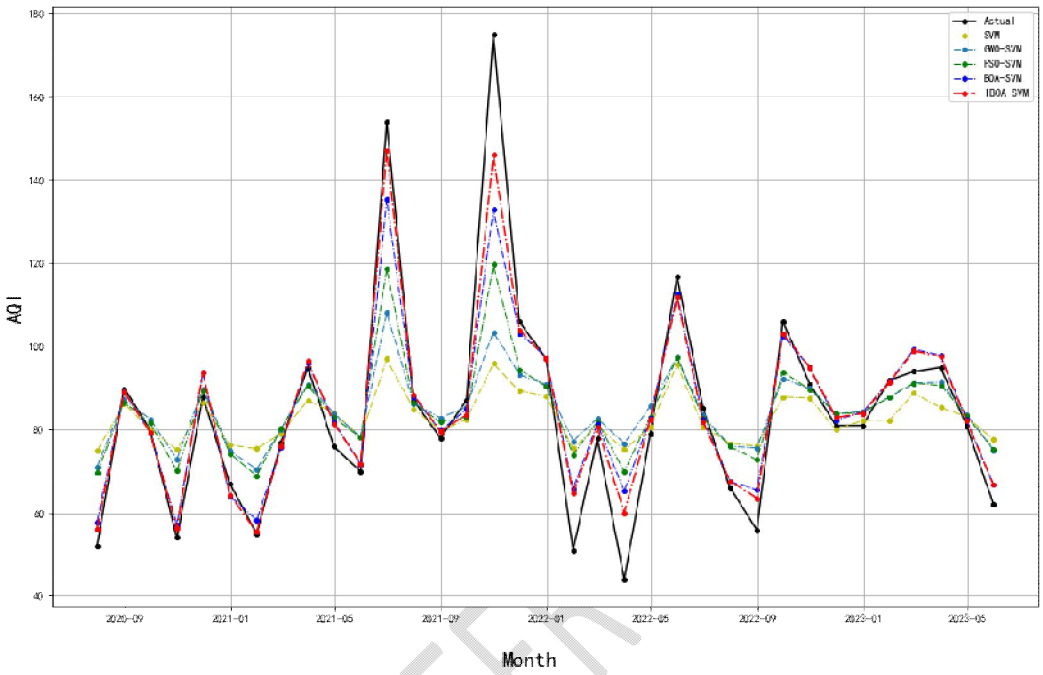


Figure 6 Comparison results of different prediction models on AQI data

Table 4 gives the average values of MSE and MAE obtained by running each algorithm 10 timeson the AQI data. The MSE of the IBOA-SVM is 58.06, and the MSE of the SVM, GWO-SVM, PSO-SVM, and BOA-SVM are 428.07, 409.97, 353.4, 113.24, respectively. The results show that the values of MSE and MAE obtained by the IBOA-SVM model are the smallest, so the predicted value of the AQI is closer to the actual value and the prediction accuracy of the IBOA-SVM model is higher.Figure 6presents a comparison chart of the actual data and the predicted value of the AQI obtained by five algorithms. The predicted value from the IBOA-SVM model exhibits a striking proximity to the actual AQI. This result clearly shows the superior prediction accuracy of the IBOA-SVM model.

6 Conclusions

In this paper, anovel IBOA is proposed, which the adaptive inertia weight ω and the dynamic switching probability p are introduced. The introduction of ω and p can obtain an effective balance between the global search and the local search and promote the capability to escape fromthe local optimum. The optimization capability of IBOA is verified oneight classical benchmark functions. The simulationresults indicate that compared with the GWO, SCA,PSO, and BOA, the IBOA has higher accuracy and stronger robustness. Finally, we combine the IBOA with the SVM model, propose the IBOA-SVM prediction model, and perform prediction experiments in two real datasets to evaluate the performance of the IBOA-SVM. Compared with SVM, PSO-SVM, GWO-SVM, and BOA-SVM, the

IBOA-SVM model has higher prediction accuracy. The experimental results also indicate that the IBOA is a superior hyper-parameter optimization technique for the SVM model.

Data availability

Data will be made available on request.

References

1. Kennedy J (2010). Particle swarm optimization. Encyclopedia of machine learning, Springer-Verlag, New York, 760-766.
2. Mirjalili S, Mirjalili SM, Lewis A (2014). Grey wolf optimizer. Advances in Engineering Software, 69:46-61. https://doi.org/10.1016/j.advengsoft.2013.12.007
3. Saremi S, Mirjalili S, Lewis A(2017). Grasshopper optimisationalgorithm: theory and application. Advances in Engineering Software, 105:30-47. https://doi.org/10.1016/j.advengsoft.2017.01.004
4. Rajeev S, Krishnamoorthy C (1997). Genetic algorithm-based methodologies for design optimization of trusses. Journal of Structural Engineering, 123(3):350-358. https://doi.org/10.1061/(ASCE)0733-9445(1997)123:3(350)
5. Mirjalili S (2015). Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowledge-Based Systems, 89: 228-249. https://doi.org/10.1016/j.knosys.2015.07.006
6. Mirjalili S (2016).Sca: a sine cosine algorithm for solving optimization problems. Knowledge-Based Systems, 96:120-133. https://doi.org/10.1016/j.knosys.2015.12.022
7. Dorigo M, Birattari M (2010). Ant colony optimization. In: Sammut C, Webb GI (Eds) Encyclopedia of machine learning. Springer US, Boston MA, pp 36–39.
8. Karaboga D, Basturk B (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39:459-471. https://doi.org/10.1007/s10898-007-9149-x
9. Mirjalili S, Gandomi AH,Mirjalili SZ et al(2017).Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Advances in Engineering Software, 114:163-191. https://doi.org/10.1016/j.advengsoft.2017.07.002
10. Arora S, Singh S (2019). Butterfly optimization algorithm: a novel approach for global optimization. Soft Computing, 23(3):715-734. https://doi.org/10.1007/s00500-018-3102-4
11. Sharma S, Chakraborty S, Saha AK et al (2022).mLBOA: amodifedbutterfyoptimization algorithm with lagrangeinterpolation for global optimization. Journal of Bionic Engineering, 19:1161-1176. https://doi.org/10.1007/s42235-022-00175-3
12. Bendahmane A, Tlemsani R (2023). Unknown area exploration for robots with energy constraints using a modified butterfly optimization algorithm. Soft Computing, 27:3785-3804. https://doi.org/10.1007/s00500-022-07530-w
13. Long W, Jiao J, Wu T et al (2022). A balanced butterfly optimization algorithm for numerical optimization and feature selection. Soft Computing, 26:11505-11523. https://doi.org/10.1007/s00500-022-07389-x
14. Xia Q, Ding Y, Zhang R et al (2022). Blind source separation based on double-mutant butterfly optimization algorithm. Sensors, 22:3979. https://doi.org/10.3390/s22113979

15.Zhang M, Wang D, Yang J (2022). Hybrid-flash butterfly optimization algorithm with logistic mapping for solving the engineering constrained optimization problems. Entropy, 24:525. https://doi.org/10.3390/e24040525

16.Zhou H, Cheng H, Wei Z et al (2021). A hybrid butterfly optimization algorithm for numerical optimization problems. Computational Intelligence and Neuroscience, 14:7981670. https://doi.org/10.1155/2021/7981670

17.Mortazavi A, Moloodpoor M (2021) Enhanced butterfly optimization algorithm with a new fuzzy regulator strategy and virtual butterfly concept. Knowledge-Based Systems, 228:107291. https://doi.org/10.1016/j.knosys.2021.107291

18.He K, Zhang Y, Wang Y et al (2024). EABOA: enhanced adaptive butterfly optimization algorithm for numerical optimization and engineering design problems. Alexandria Engineering Journal, 87: 543-573. https://doi.org/10.1016/j.aej.2023.12.050

19.Alhasnawi BN, Jasim BH, Bureš V et al (2023). A novel economic dispatch in the stand-alone system using improved butterfly optimization algorithm. Energy Strategy Reviews, 49:101135. https://doi.org/10.1016/j.esr.2023.101135

20.Huang H, Tian M, Zhou J et al (2023). Reliable task allocation for soil moisture wireless sensor networks using differential evolution adaptive elite butterfly optimization algorithm. Mathematical Biosciences and Engineering, 20(8): 14675-14698. https://doi.org/10.3934/MBE.2023656

21.Sureshkumar S, Prasanna Venkatesan GKD, Santhosh R (2023) Adaptive butterfly optimization algorithm (ABOA) based feature selection and deep neural network (DNN) for detection of distributed denial-of-service (DDoS) attacks in cloud. Computer Systems Science and Engineering, 47(1). https://doi.org/10.32604/csse.2023.036267

22.Gonzalez-Sanchez B, Vega-Rodríguez MA, Santander-Jiménez S (2023). A multi-objective butterfly optimization algorithm for protein encoding. Applied Soft Computing, 139:110269. https://doi.org/10.1016/j.asoc.2023.110269

23.Jin Z, Sun Y (2023). Bearing fault diagnosis based on VMD fuzzy entropy and improved deep belief networks. Journal of Vibration Engineering & Technologies, 11:577-587. https://doi.org/10.1007/s42417-022-00595-9

24.Tang M, Cao C, Wu H et al (2022) Fault detection of wind turbine gearboxes based on IBOA-ERF. Sensors, 22:6826. https://doi.org/10.3390/s22186826

25.Long W, Wu T, Xu M et al (2021) Parameters identification of photovoltaic models by using an enhanced adaptive butterfly optimization algorithm. Energy, 229:120750. https://doi.org/10.1016/j.energy.2021.120750

26.Zhou H, Zhang G, Wang X et al (2021). Structural identification using improved butterfly optimization algorithm with adaptive sampling test and search space reduction method. Structures, 33:2121-2139. https://doi.org/10.1016/j.istruc.2021.05.043

27.Singh B, Anand P (2019). A novel adaptive butterfly optimization algorithm. International Journal of Computational Materials Science and Engineering, 7(4):1850026. https://doi.org/10.1142/s2047684118500264

28.Makhadmeh SN, Al-Betar MA, Abasi AK et al (2023). Recent advances in butterfy optimization algorithm, its versions and applications. Archives of Computational Methods in Engineering, 30:1399-1420. https://doi.org/10.1007/s11831-022-09843-3

29.Cortes C, Vapnik V (1995). Support-vector networks. Machine Leaning, 20:273-297. https://doi.org/10.1023/A:1022627411411