Network Security Based on Improved Genetic Algorithm and Weighted Error Back-Propagation Algorithm

Junjuan Liang
Henan Polytechnic Institute, Nanyang, 473000, China

***Abstract*—In order to solve the problem of feature selection and local optimal solution in the field of network security, a network security protection model based on improved genetic algorithm and weighted error back-propagation algorithm is proposed. The model combines the dynamic error weight and adaptive learning rate of the weighted error back-propagation algorithm to improve the learning ability of the model in dealing with classification imbalance and dynamic attack mode. In addition, the global search capability of genetic algorithm is utilized to optimize the feature selection process and automatically adjust the hyperparameter settings. The experimental results show that the proposed model has an average accuracy of 96.7%, a recall rate of 93.3% and an F1 value of 0.91 on the CIC-IDS-2017 dataset, which has significant advantages over traditional detection methods. In many experiments, the accuracy of normal data is up to 99.97%, the accuracy of known abnormal behavior data is 99.31%, and the accuracy of unknown abnormal behavior data is 98.13%. These results show that this method has high efficiency and reliability when dealing with complex network traffic, and provides a new idea and method for network security protection research.**

***Keywords*—Genetic algorithm; weighted error back-propagation; multiple strategies; network security**

I. INTRODUCTION

With the speedy prosperity of information technology, network security matters have become more and more notable, and the ways of network attacks have become more complex and diverse. These attacks not only affect the operations of businesses and government agencies, but also pose a significant threat to the privacy and data security of individual users [1]. At the same time, with the rapid growth of network traffic, how to effectively detect and identify potential security threats has become a non-negligible problem that requires urgent handling in the current field of network security. Traditional network security detection methods often rely on matching rules or feature libraries, making it difficult to cope with rapidly evolving attack patterns [2-3].

Thus, a lot of domestic and foreign researchers have promoted in-depth study on network security detection. Wei K et al. focused on optimizing network intrusion detection technology, using a combination of algorithm technology and artificial intelligence to achieve intrusion detection and defense of network security systems. Focusing on the issues of low efficiency and poor detection precision of classical K-means clustering algorithm, an improved K-means clustering algorithm network security detection model was proposed, and experiments were conducted using a dataset. The results indicated that this improvement measure could significantly improve detection efficiency and accuracy [4]. Chen Z optimized traditional network security detection algorithms in response to constantly changing and upgrading network attack techniques. A new network security detection model was constructed using radial basis function neural networks as the main body, optimized through simulated annealing algorithm and hybrid hierarchical genetic algorithm (GA), and relevant experiments were conducted. The results showed that the optimized detection model's predicted values in 15 samples highly approached the true values, which could provide assistance for maintaining network security [5]. Khan M and Ghafoor L explored the special challenges brought by adversarial attacks in the field of network security, and strengthened machine learning-based network security detection systems to address challenges such as the dynamic nature of network environments and the need for real-time decision-making. The experimental results indicated that the research method could provide more powerful and resilient solutions when facing network attacks [6]. Yin L and Zhang D raised a network security detection algorithm grounded on vague reasoning. Firstly, they combined fuzzy reasoning and probability density features to evaluate security data. Then the features of network intrusion data were extracted to achieve safety possibility computation and virus attack detection. The data showed that the raised algorithm had higher calculation precision for network safety possibility, achieved network safety possibility computation and data checking, and enhanced the network's security defense capability [7].

Memon I proposed an authentication key establishment protocol based on IPv6 to protect sensitive information in road network environment. The protocol eliminated duplicate address detection by allowing a moving vehicle to pick up a unique address from a neighboring vehicle or roadside unit, while automatically recycling the address space leaving the vehicle for reallocation. The proposed protocol had the characteristics of anonymous authentication, confidentiality and high efficiency, and the evaluation results showed that the protocol effectively improved the performance of address configuration, and was also very secure in preventing passive and active attacks [8]. Aiming at the growing network demand and massive data traffic management, Farhan N et al. proposed a method based on hybrid clustering to improve the wireless resource management effect of heterogeneous networks [9]. By

analyzing different clustering classifications and existing technologies, they aim to achieve high throughput, low interference and low latency in ultra-dense networks. The results showed that the hybrid clustering technology could effectively improve the wireless resource management in HetNets, increase throughput, and reduce the inter-layer and intra-layer interference, thus enhancing the network performance. Junejo M H et al. addressed the growing security threat in Vehicle ad-hoc networks (VANETs) by proposing cybersecurity solutions that combined machine learning and artificial intelligence techniques to enhance the safety and efficiency of public transportation. In this study, the trust mechanism was compared with cryptography, and its different performance in VANET application and security requirements was analyzed [10]. Although the above methods have high recognition accuracy in the known attack behavior, they may have limitations in the face of new and unknown attack modes. In the actual network environment, the characteristics of network traffic may change with time, and the strategies and techniques of attackers also develop dynamically. Although the feature selection of the current model is optimized based on GA, how to dynamically update and re-select the features to adapt to such changes is still an urgent problem to be solved.

Therefore, a network security protection model based on Weighted Error Back-Propagation (WEBP) and Improved GA (IGA) is proposed in this study. Compared with the existing researches, the main objective of this study is to solve the problem that the efficiency of feature selection is insufficient, resulting in insufficient model learning, and thus affecting the detection accuracy and efficiency. The study is also devoted to solving the problem that the fixed learning rate is difficult to adjust dynamically and cannot adapt to the change of data flow in real time, which is easy to cause the model to fall into the local optimal. Moreover, the study hopes to address the problem that current methods are unable to respond to rapidly evolving attack patterns in a timely and effective manner. The innovation of this model lies in the introduction of a weighted error mechanism, which allows the model to adjust the learning process based on the importance of different samples, thereby improving its adaptability to complex network environments and combining the population search characteristics of GA. Feature selection and hyperparameter optimization are implemented to enhance the global search capacity and robustness of the model. The contribution of the research is to solve the problem of low feature selection efficiency on high-dimensional data, so as to effectively improve model performance and learning efficiency. It provides an effective solution for the field of network security detection, can better deal with the complex network attack mode, and provides a new idea for the practical application of data protection and network firewall.

The research structure mainly includes four parts. The first part constructs the network security detection model based on the WEBP algorithm, analyzes the limitations of the model, and further optimizes it. The second part verifies the performance of the proposed model, and designs experiments to analyze the network security protection capability of the proposed model. The third part is the summary and analysis of the experimental results, and the comparison with the same type of research, emphasizing the advantages of the research method. The last part summarizes the content of the article, and points out the current limitations and future development direction.

II. Methods and Materials

A. Network Security Detection Model Grounded on WEBP Algorithm

WEBP is an improved Back-Propagation Neural Network (BPNN). In traditional back-propagation algorithms, all training samples are assigned the same importance, while the WEBP algorithm allows for adjusting errors during the training process based on the importance or confidence of different samples [11-12]. WEBP essentially introduces error weights in the back-propagation of BPNN to adjust errors. BPNN is a widely studied algorithm in ANN, with the core idea of bidirectional signal transmission [13-14]. Forward transmission refers to the sequential transmission from the input end to the output end. Reverse propagation is when there is a deviation in the results during forward propagation, and the error results are reversed and distributed to each layer structure to correct the weights in the structure [15]. BPNN is usually constructed from three parts: Input Layer (IL), Hidden Layer (HL), and Output Layer (OL). The specific composition model is shown in Fig. 1.

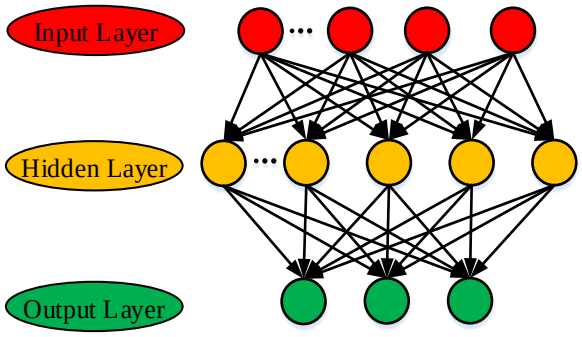


Fig. 1. Example diagram of BPNN structure.

In Fig. 1, after receiving the information data from IL, HL in the BPNN structure will pass the data information to the next layer through a weight function and organize it in the last layer to obtain the final output data. At the same time, BPNN continuously adjusts the weights and thresholds of the structure grounded on the deviation values between the output results and the actual results. The amount of neurons in the IL and OL is defined by the dimensions of the input samples and output results, respectively. However, the selection of neurons in the HL is more sophisticated, and the current intelligence determines it through empirical formulas, as shown in Formula (1).

$$O = \sqrt{n+m} + a \tag{1}$$

1. In Formula (1), 0 represents the HL neuron. n and m represent the number of neurons in the IL and OL separately. a represents a constant, with values ranging from [1,10]. Formula (1) in network security, the strong generalization ability enables the model to handle unseen attack patterns and normal data, effectively identifying potential

threats. In conventional BPNN, using a fixed learning rate makes it difficult to ensure the network has the best learning efficiency in realistic tasks. To solve this issue, research is being conducted on using adaptive learning rates in network structures. Assuming the initial learning rate is $\mu(0)$ and the network fault obtained by the model during the iteration process is denoted as $E(n)$, the change in learning rate is shown in Formula (2).

$$\mu(n)=\begin{cases}\beta\mu(n-1) & E(n)<E(n-1),(1<\beta<1.5)\\ \gamma\mu(n-1) & E(n)>E(n-1),(0.5<\gamma<1)\\ \mu(n-1) & other\end{cases} \quad (2)$$

In Formula (2), β and γ represent constant coefficients, with values of 1.05 and 0.7, respectively. Formula (2) In the dynamic environment of network security, attack patterns and normal traffic are constantly changing, and adaptive learning rates can make the model more flexible in adapting to these changes. In the process of error back-propagation, adaptive adjustment of learning rate can effectively improve convergence speed. Nevertheless, the model continues to disregard the direction of gradient descent throughout the entire process, which renders the model unstable and susceptible to becoming trapped in local optima. The problem of gradient descent direction is solved by introducing error weights, and its description is represented in Formula (3).

$$\Delta w(n)=\alpha\Delta w(n-1)-\mu\frac{\partial E(n)}{\partial w(n)} \quad (3)$$

In Formula (3), α is the momentum term and w represents the weight. Formula (3) can be regarded as a first-order difference formula of $\Delta w(n)$, and its formula is represented in Formula (4).

$$\Delta w(n)=-\mu\sum_{t=0}^{n}\alpha^{n-t}\frac{\partial E(n)}{\partial w(n)} \quad (4)$$

In Formula (4), t represents the time series. The above is the correction of weights, and the specific way to adjust the weights of BPNN is shown in Formula (5).

$$w(n+1)=w(n)-\mu(n)\sum_{t=0}^{N}\alpha^{n-t}\frac{\partial E(n)}{\partial w(n)} \quad (5)$$

Formulas (4) and (5) provide specific methods for weight updating, enabling the model to more accurately correct weights and adapt to new data inputs. In network security protection, it can help detection models quickly adjust to adapt to new threats.

B. Construction of Network Security Protection Model Based on GA-WEBP

Although WEBP improves the performance of BPNN by introducing mechanisms such as weights and adaptive learning rates, the capacity of the model largely relies on the selection and representation of input features. Incorrect or insufficient feature selection may result in the model learning insufficient information, while traditional feature selection methods may be inefficient in processing high-dimensional data. In the field of network security, the feature space is usually large, and anomaly detection is highly sensitive to features. Therefore, the study optimized the model by introducing GA. The logic of GA is represented in Fig. 2.

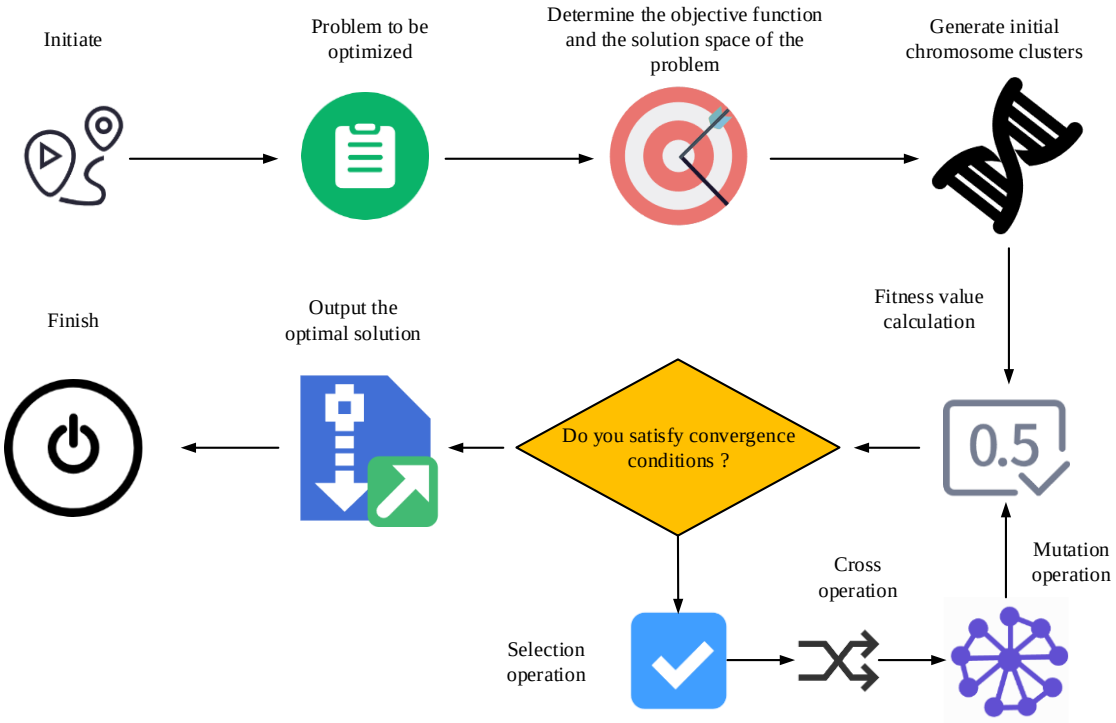


Fig. 2. Schematic diagram of GA process.

In Fig. 2, during the initial stage of GA, the algorithm randomly generates a series of individuals and evaluates each individual based on a predetermined goal, assigning them a fitness value. By comparing these fitness values, individuals who perform well as the new generation are selected. On this basis, screening is conducted to eliminate individuals with poor performance [13-14]. Then, the selected individuals are recombined through crossover and mutation operations to produce the next generation of individuals with excellent traits, gradually approaching the best remedy. Finally, the optimal one is extracted from the last generation population to obtain an approximate optimal solution to the problem. Overall, the process of traditional Gas involves encoding, selection, crossover, and mutation [15-16]. Finally, when the termination conditions are met, that is, when the specified number of iterations or iteration accuracy is reached, as well as the target conditions, the iteration can be stopped. In network security protection, binary encoding is studied as the encoding method for Gas. The second step is the selection operation, which selects high-quality individuals based on their fitness in the environment, in order to use them for generating the next generation. This step is grounded on the law of natural selection, where individuals with greater fitness have a greater possibility of being chosen, while individuals with less fitness have a smaller possibility of being chosen. The study used random sampling as the model selection operation, in which the selection probability of each individual is grounded on their fitness, but the selection process is random. This means that even individuals with lower fitness have the opportunity to be chosen. The possibility of an individual being chosen is shown in Formula (6) [20].

$$F(x_i)=\frac{f(x_i)}{\sum_{i=1}^{Nind} f(x_i)} \tag{6}$$

In Formula (14), $F(x_i)$ points to the fitness of an individual. In network security, fitness usually reflects the effectiveness of an individual under specific security policies or protective measures, such as the ability to detect attacks and reduce false positive rates. $f(x_i)$ points to the possibility of an individual being chosen. In network security applications, this probability determines which individuals (i.e. protection policies or system configurations) can continue to be optimized and evolved, playing a crucial part in improving the comprehensive security of the system. The GA increases the fitness value of individuals by performing crossover operations on different parents in the third step, where the single point crossover structure is shown in Fig. 3.

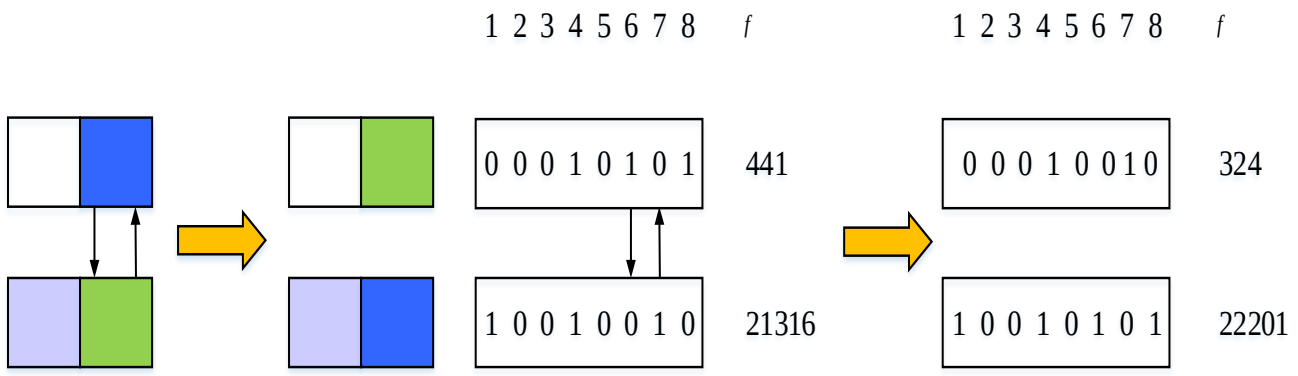


Fig. 3. Single point cross structure.

In Fig. 3, a random crossover point is selected on the chromosome, and the chromosome is divided into front and back parts. Gene exchange is performed in the front or back part of the crossover point to generate new offspring individuals. The mathematical expression for crossover operation is shown in Formula (7) [21].

$$\begin{cases} X_A^{t+1}=\varepsilon X_B^t+(1-\varepsilon)X_A^t \\ X_B^{t+1}=\varepsilon X_A^t+(1-\varepsilon)X_B^t \end{cases} \tag{7}$$

In Formula (7), X_A^t and X_B^t represent the parent individuals performing the crossover operation, and the two parent individuals represent two different protection strategies or configurations. X_A^{t+1} and X_B^{t+1} represent the offspring individuals formed after cross operation, and the newly generated individuals can display different combinations of security policies, helping the network security team find better protection solutions. ε represents the intersection rate. If the intersection rate is high, the algorithm will explore more combinations and may find more effective security protection methods; If it is too low, it may lead to insufficient exploration and inability to adapt to rapidly changing threats. After crossover operation, the GA finally performs mutation operation, as shown in Formula (8).

$$X^{t+1}=X^t-0.5L\Delta \tag{8}$$

In Formula (8), L represents the range of variable values, defining the range within which individuals can vary, ensuring that the mutated individuals remain within the valid parameter space, thereby enhancing the practical applicability of the model. The GA-WEBP network security protection model constructed above, although capable of supporting hyperparameter optimization, may not be able to adapt in a timely manner when facing rapidly changing attack patterns and network traffic. Therefore, research will focus on targeted optimization of Gas from multiple perspectives.

C. Improved Optimization Grounded on GA-WEBP Model

In the optimization of network security protection models based on GA-WEBP, research is conducted from the perspectives of convergence speed, crossover probability, mutation probability, etc. In terms of algorithm convergence speed, the selection of the initial population has a direct correlation. Therefore, the study adopts a real number chromosome as the initial population, and its calculation is represented in Formula (9).

$$X_i = \lceil X_{i\max} - X_{i\min} \times random[0,1] + X_{i\min} \rceil \quad (9)$$

In Formula (9), $\lceil \cdot \rceil$ represents the upward rounding function, X_i represents the value of genes on chromosomes, $X_{i\max}$ and $X_{i\min}$ represent the maximum and minimum values of the values, and $random$ represents any real number within a certain interval range. In network security protection, a good initial population can accelerate the initial exploration of the model, enabling GA to find feasible solutions faster, thereby cutting the amount of subsequent iterations and enhancing algorithm validity. Its expression is shown in Formula (10).

$$\begin{cases} o_c = \begin{cases} \dfrac{s_1(f_{\max} - f_c)}{f_{\max} - f_{avg}} & f_c \geq f_{avg} \\ s_2 & f_c < f_{avg} \end{cases} \\ o_m = \begin{cases} \dfrac{s_3(f_{\max} - f_m)}{f_{\max} - f_{avg}} & f_m \geq f_{avg} \\ s_4 & f_m < f_{avg} \end{cases} \end{cases} \quad (10)$$

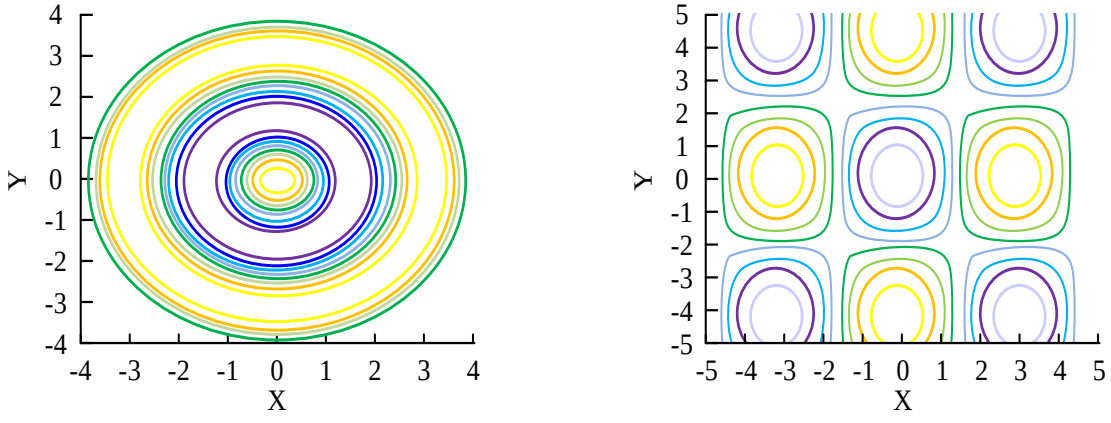
In Formula (10), o_c represents the crossover probability. In network security protection, a higher crossover probability can help discover new attack defense strategies and more effective protection measures. o_m represents the probability of mutation. Enhanced mutation operations can encourage the model to consider non-traditional attack methods and response strategies, thereby improving the adaptability of protection capabilities. s represents the adaptive parameter, and the value of the adaptive parameter is a constant not greater than 1. As the attack and defense strategies continue to evolve, the model can adjust the strategy based on real-time fitness feedback. f represents the fitness value. In Formula (10), the value of the adaptive parameter is greater than the value that should satisfy Formula (11).

$$\begin{cases} s_1 > s_2 \\ s_3 > s_4 \end{cases} \quad (11)$$

In the initial stage of the algorithm, individuals with higher fitness in the population will be retained, but the retained individuals are not the best ones. Therefore, to improve the performance of the algorithm's best remedy, further improvements should be made to crossover and mutation operations. The main purpose of improvement is to ensure that the algorithm can continue to perform crossover and mutation operations even when it has the best individual, as shown in Formula (12).

$$\begin{cases} o_c = \begin{cases} o_{c1} \dfrac{1}{(o_{c1} - o_{c2}) + e^{\frac{f_c - f_{avg}}{f_{\max} - f_{avg}}}} & f_c \geq f_{avg} \\ s_2 o_{c1} & f_c < f_{avg} \end{cases} \\ o_m = \begin{cases} o_{m1} \dfrac{1}{(o_{m1} - o_{m2}) + e^{\frac{f_c - f_{avg}}{f_{\max} - f_{avg}}}} & f_m \geq f_{avg} \\ s_2 o_{m1} & f_m < f_{avg} \end{cases} \end{cases} \quad (12)$$

In Formula (12), the value of $o_{c1} = 0.9$ is adjusted to 0.9, the value range of o_{c2} is [0.5,1], the value of $o_{m1} = 0.1$ is adjusted to 0.1, and the value range of o_{m2} is [0.05,0.1]. For network security protection, continuous crossover and mutation operations can help models adapt to new attack methods and traffic patterns, enhancing their dynamic defense capabilities. The study will use two sets of benchmark functions to testify the solving ability of the improved algorithm, namely the Schaffer function and the Griebank function. The contour plots of the two sets of functions are shown in Fig. 4.



(a) The Schaffer function (b) The Griewank function

Fig. 4. Contour map of two benchmark functions.

Fig. 4 shows the contour plots of the Schaffer function and the Griewank function. Contour plots plot the variation of a function in a given input space by connecting points with the same function values into lines. Through this visualization method, the function value distribution and its fluctuation characteristics in different regions can be intuitively understood. The Schaffer function is often used to test the performance of optimization algorithms because it has multiple local minima and one global minimum, usually near the origin. The contours shown in the figure present a highly concentrated and complex region centered on a smaller function value surrounded by a ring region of larger function values. This distribution shows that the solution of the optimization problem is usually near the origin and the optimization process is complicated. The Griewank function is another classical multi-variable function used to test global optimization algorithms. It is characterized by the existence of a large number of local minima, and the zero point is in a large region. The regular wavy structure in the contour map indicates its various local minima. These local minima are evenly distributed, which indicates the high complexity of the function.

III. Results

A. Performance Testing Based on Improved GA-WEBP Algorithm

The study conducted simulation experiments using MATLAB and analyzed the performance of the IGA through two selected benchmark functions. Firstly, the study analyzed the Schaffer function, which reached its maximum at the origin. Therefore, the study set the parameter values of the GA as follows: the initial population size was 100, the parameter settings satisfied $s_1 = 2s_2 = s_3 = 2s_4 = 1$, and the maximum iteration number of the GA was 100. By improving the GA to solve the Schaffer function, the results are shown in Fig. 5.

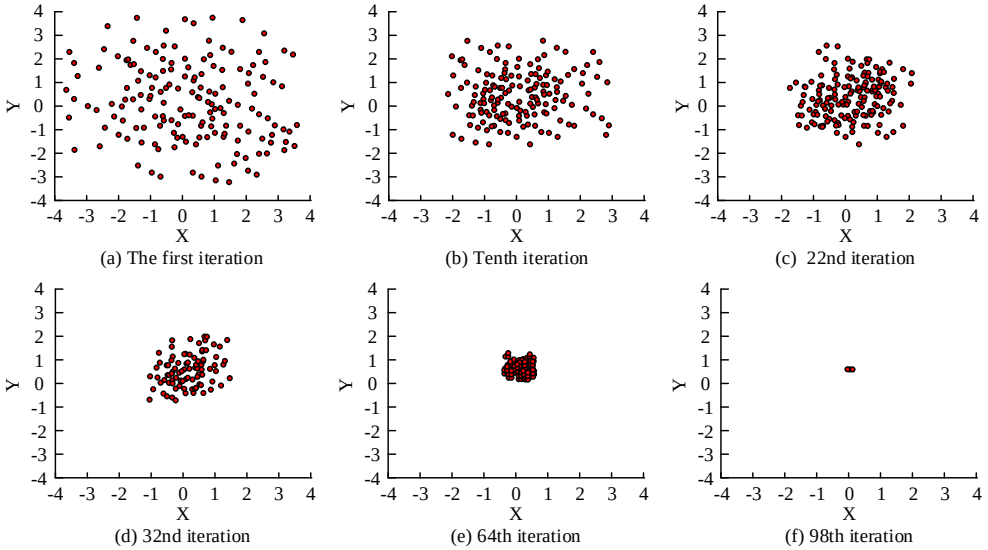
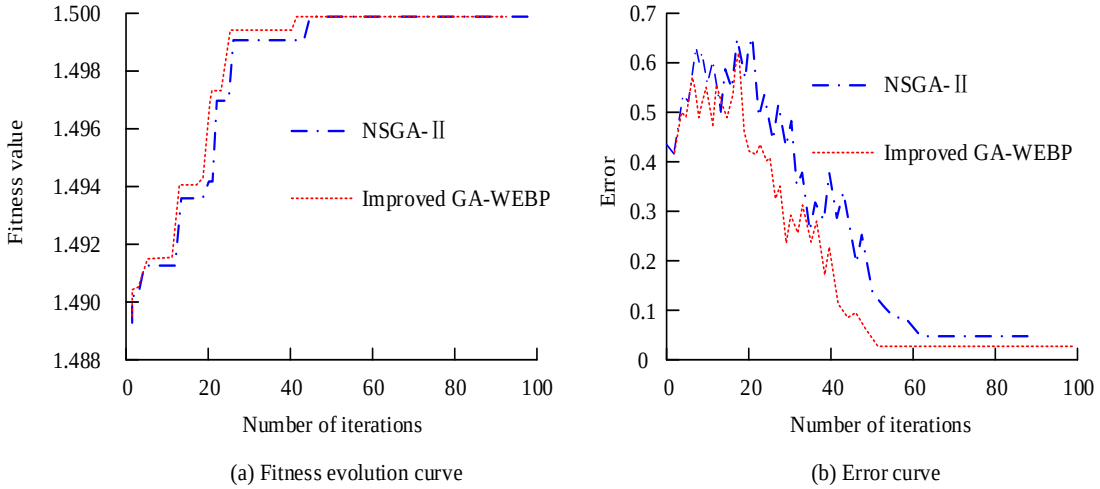


Fig. 5. Partial solution process of Schaffer function.

Fig. 5(a)-5(f) show the results of the algorithm's first, tenth, 22nd, 32nd, 64th, and 98th iterations, respectively. The results showed that the IGA gradually converged at the 32nd iteration, and the algorithm basically completed convergence at the 64th iteration. The results showed the ability of the model to explore the search space efficiently. Compared with the traditional algorithm, the improved GA had a faster convergence rate, which was due to the introduction of adaptive mechanism. As a result, the algorithm can effectively filter out the best solution in the early stage. The research used the fitness curve and iteration error curve of Schaffer function as the evaluation index of algorithm performance, and compared with the Non-dominated Sorting GA II (NSGA-II) to verify the effectiveness and progressiveness of the IGA. The results are shown in Fig. 6.

Fig. 6(a) shows the fitness curve results of the Schaffer function, where the NSGA-II algorithm began to converge at the 48th iteration, the proposed algorithm began to converge at the 42nd iteration, and the algorithm had a solution value of 1.5 for the multivariate unimodal function. Fig. 6(b) shows the iterative error curve of the Schaffer function, where the NSGA-II algorithm reached its minimum error at the 60th iteration, with a minimum error of 4.8%; The IGA proposed in the study achieved the minimum error at the 53rd iteration, and the minimum error result was 2.7%. In the Griebank function, there were two extrema in the domain, and the local minima had a regular arrangement. The study set the population size to 100, the adaptive parameters satisfied $s_1 = 2s_2 = s_3 = 2s_4 = 1$, and the maximum number of iterations was 250. The results showed that the research algorithm could achieve efficient feature selection and optimization in high-dimensional and complex optimization environments [19], especially in the field of network security, where it was crucial to quickly adapt to the dynamic changing environment in the face of complex network traffic data. By improving the GA to solve the Griebank function, the results are shown in Fig. 7.

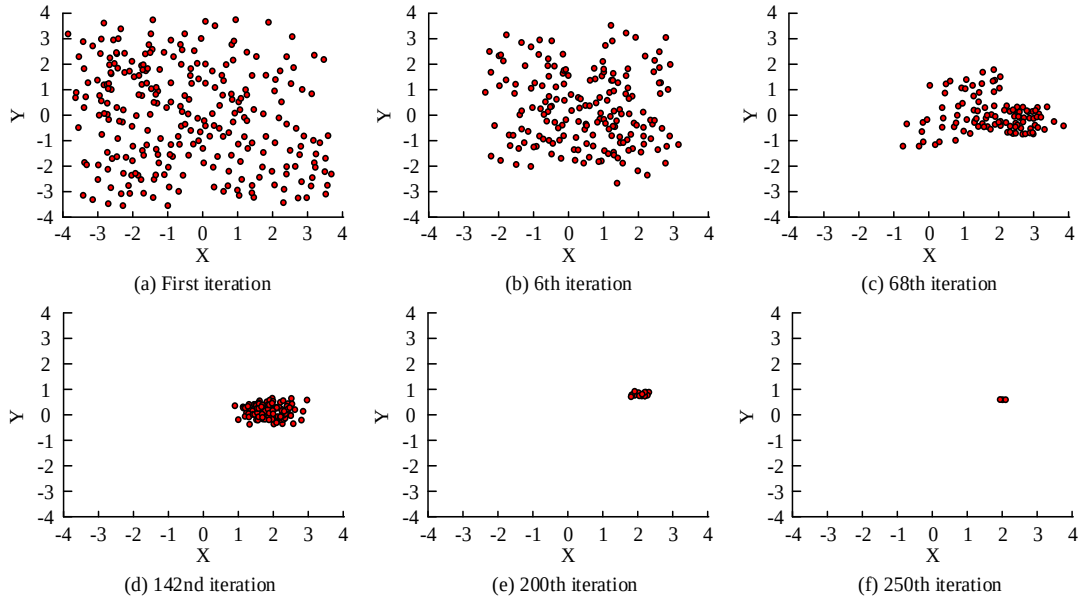


(a) Fitness evolution curve (b) Error curve

Fig. 6. The fitness curve and iteration error curve of Schaffer function.

Fig. 7(a)-7(f) show the results of the first, sixth, 68th, 142nd, 200th, and 250th iterations of the algorithm, respectively. The results showed that the IGA gradually converged at the 68th iteration, and the algorithm basically completed convergence at the 130th iteration. Due to the complexity of Griewank function, the research method showed a good search strategy ability to find the global optimal solution in a large number of local minima, indicating its effectiveness for testing complex optimization problems. The fitness curve and iteration error curve of the Griewank function were studied as evaluation indicators for algorithm performance, and the results are shown in Fig. 8.

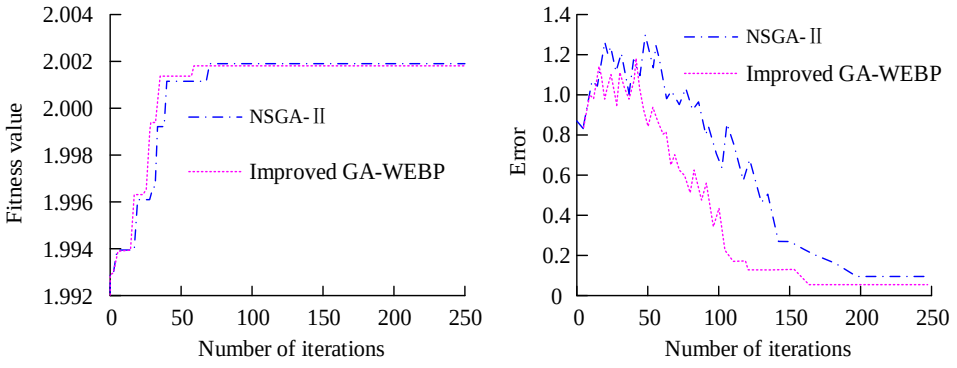
Fig. 8 (a) gives the fitness curve results of the Griewank function, where the NSGA-II algorithm began to converge at the 72nd iteration, the proposed algorithm began to converge at the 65th iteration, and the algorithm had a solution value of 2.0 for the multivariate unimodal function. Fig. 8 (b) gives the iterative error curve of the Griebank function, where the NSGA-II algorithm reached its minimum error at the 200th iteration, with a minimum error of 7.7%. The IGA proposed in the study achieved the minimum error at the 164th iteration, and the minimum error result was 5.2%. In the network attack detection, the attack mode was often varied, and the local minimum represented the misjudgment or omission of certain characteristics. The strong anti-interference ability ensured the robustness of the network security detection system, enabled it to effectively identify various attacks in the real environment, and reduced the false positive rate and false negative rate. Based on the analysis of the results in Fig. 7 and Fig. 8, the proposed IGA had good global optimization ability and fast convergence, and its effectiveness had been verified.



(a) First iteration (b) 6th iteration (c) 68th iteration

(d) 142nd iteration (e) 200th iteration (f) 250th iteration

Fig. 7. Partial solution process of Griebank function.



(a) Fitness evolution curve (b) Error curve

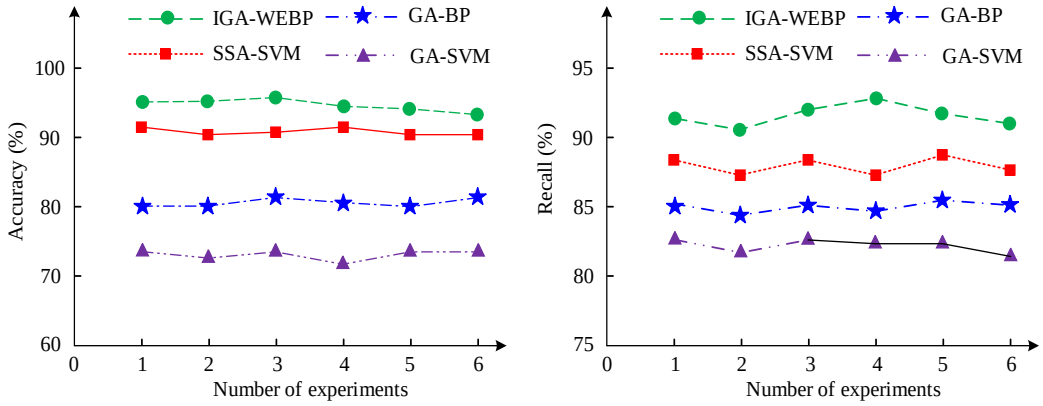
Fig. 8. The fitness curve and iteration error curve of the Griebank function.

B. *Network Security Protection Analysis Based on Improved GA-WEBP Algorithm*

To testify the effectiveness of the proposed algorithm, it was compared and analyzed with the Genetic Back-Propagation Algorithm (GA-BP), Genetic Support Vector Machine (GA-SVM), and Sparrow Search Support Vector Machine (SSA-SVM) algorithms [17]. The algorithm performance was validated using the CIC-IDS-2017 dataset, and precision, recall, F1 value, PR curve, ROC curve, and mean absolute error were used as comparison indicators for performance comparison. The accuracy and recall results of the four algorithms are represented in Fig. 9.

Fig. 9 (a) shows the accuracy comparison results of four algorithms. According to Fig. 9 (a), the accuracy curves of IGA-WEBP were higher than the other three compared algorithms in all four algorithms, with an average accuracy of 96.7%, which was higher than the 91.4% of SSA-SVM algorithm, 81.0% of GA-BP algorithm, and 74.6% of GA-SVM algorithm. Fig. 9 (b) shows the comparison results of recall rates for four algorithms. According to Fig. 9 (b), the recall curves of IGA-WEBP in all four algorithms were higher than the other three compared algorithms, and its average recall rate was 93.3%, which was higher than the 88.6% of SSA-SVM algorithm, 85.4% of GA-BP algorithm, and 83.0% of GA-SVM algorithm. The above results indicated that, in terms of accuracy and recall, the SSA-SVM algorithm outperformed the three compared algorithms in terms of performance. The good balance of accuracy and recall meant that the model could effectively reduce false alarms in cybersecurity, reduce the burden on security operations teams, and improve response efficiency. This helped to enhance the accuracy of decision-making in practical applications, making network protection measures more targeted. The PR curves and F1 values of the four algorithms are shown in Fig. 10.

Fig. 10 (a) shows the comparison results of PR curves for four algorithms. According to Fig. 10 (a), the space under the PR curve of IGA-WEBP in the four algorithms was 0.89, which was higher than the 0.83 of SSA-SVM algorithm, 0.76 of GA-BP algorithm, and 0.53 of GA-SVM algorithm [18]. Fig. 6(b) shows the comparison results of F1 values for four algorithms. According to Fig. 6(b), the F1 value curves of IGA-WEBP in all four algorithms were higher than the other three compared algorithms, and its average recall rate was 0.91, which was higher than the 0.86 of SSA-SVM algorithm, 0.79 of GA-BP algorithm, and 0.70 of GA-SVM algorithm. The above results indicated that, in terms of accuracy and recall, the ISSA-SVM algorithm outperformed the three compared algorithms in terms of performance. The PR curves and F1 values of the four algorithms are shown in Fig. 11.



(a) Accuracy rate comparison results (b) Recall rate comparison results

Fig. 9. Comparison results of accuracy and recall.

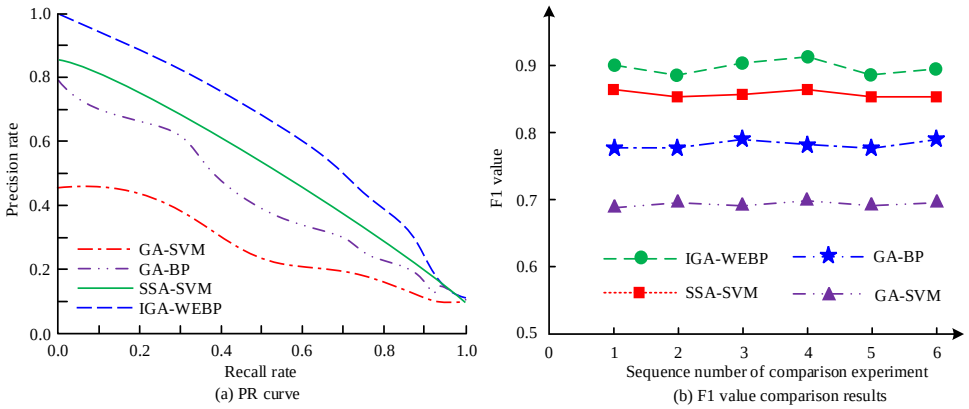


Fig. 10. Comparison results of PR curve and F1 value.

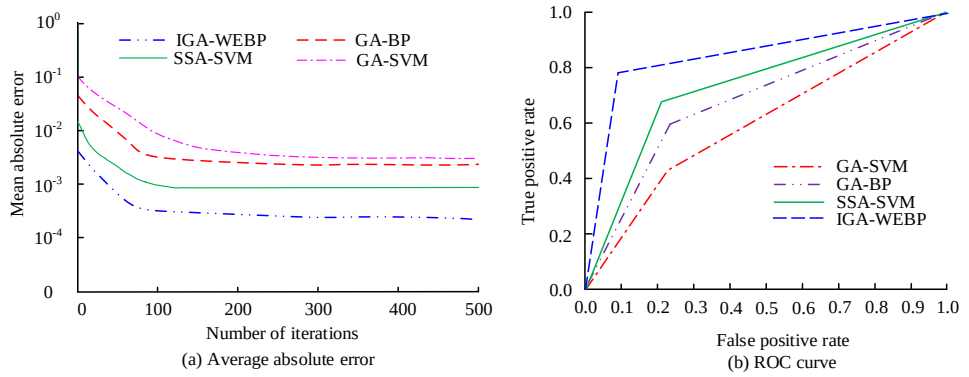


Fig. 11. Mean absolute error and ROC curve.

Fig. 11 (a) shows the comparison results of the average absolute error curves of four algorithms. According to Fig. 11 (a), IGA-WEBP had the lowest average absolute error curve among the four algorithms, and its stable average absolute error was 0.0034. Below 0.0023 for SSA-SVM algorithm, 0.0084 for GA-BP algorithm, and 0.0083 for GA-SVM algorithm. Fig. 11 (b) shows the comparison results of ROC curves for four algorithms. According to Fig. 11 (b), the space under the ROC curve of IGA-WEBP in the four algorithms was 0.88, which was higher than the 0.77 of SSA-SVM algorithm, 0.72 of GA-BP algorithm, and 0.64 of GA-SVM algorithm. The above results indicated that, in terms of average absolute error and ROC curve dimensions, the capacity of the IGA-WEBP algorithm was superior to the three compared algorithms. The results showed that the model had high efficiency in abnormal traffic detection, which indicated that the algorithm could not only handle normal traffic well, but also accurately capture abnormal traffic. The low error rate meant that the system could monitor and analyze the network in a more precise manner, and discover potential threats in time to enhance the security of the network environment. In addition, the actual performance of the network security protection system designed for research was analyzed, and the results are represented in Table I.

TABLE I. VERIFICATION RESULTS OF NETWORK SECURITY PROTECTION PERFORMANCE

Number of experiments	Data type	Input quantity	Correct corresponding quantity	Accuracy of measurement (%)
The first time	Normal data	9000	8974	99.71
	Known abnormal behavior data	3300	3276	99.27
	Unknown abnormal behavior data	560	543	96.96
The Second time	Normal data	11000	10997	99.97
	Known abnormal behavior data	3500	3476	99.31
	Unknown abnormal behavior data	500	486	97.20
The third time	Normal data	10000	9862	98.62
	Known abnormal behavior data	3000	2976	99.20
	Unknown abnormal behavior data	480	471	98.13

Table I shows that the proposed algorithm achieved recognition accuracy of over 95% for normal data, known abnormal behavior data, and unknown abnormal behavior data in each test. In the recognition of normal data, the highest accuracy reached 99.97%; the highest accuracy in identifying known abnormal behavior data was 99.31%; the recognition accuracy of unknown abnormal behavior data reached a maximum of 98.13%. The results indicated that the research method could effectively detect and distinguish normal traffic and potential abnormal behavior when facing different types of data streams, which helped to improve the effectiveness of network security protection in practical applications. These high recognition accuracy data reflected the high sensitivity and adaptability of the algorithm to network traffic changes, and could provide strong support for real-time protection. In network security applications, fast and accurate response capabilities are key to preventing data breaches and system breaches.

IV. Discussion

In the above experiment, when testing the performance of the improved method on two benchmark functions, the minimum error of the improved algorithm in the Schaffer function fitness curve was 2.7%, while NSGA-II was 4.8%; In the fitness curve of the Griebank function, the improved algorithm achieved a minimum error of 5.2% at the 164th iteration, while NSGA-II achieved a minimum error of 7.7% at the 200th iteration. For the analysis of network security protection based on the IGA-WEBP algorithm, the average accuracy of the IGA-WEBP model is 96.7%, the average recall is 93.3%, and the F1 value is 0.91, which is significantly better than the comparison algorithm. The reason why the model proposed by the research has excellent performance is mainly due to its structural advantages. The WEBP used in the IGA-WEBP model dynamically adjusts the weights of each sample. Unlike BPNN which relies solely on error feedback, WEBP allows the model to update based on the importance of the samples during the learning process. This enables samples that are more representative in specific situations to commit a greater effect on the practicing of the model, thereby improving the model's ability to recognize complex attack patterns. The importance of GA lies in its ability to optimize algorithm feature selection and hyperparameter settings through natural selection and genetic operations. Traditional feature selection methods have low efficiency in high-dimensional data processing, while GA can efficiently traverse the search space and find the optimal feature combination. In the same type of research, Unnisa N et al. proposed an intrusion detection system mainly used for detecting threats, and conducted research using various algorithms implemented by machine learning, achieving certain results in network intrusion detection [22]. Kim S et al. conducted a comprehensive review of the security threats faced by cyber physical systems. At the same time, the impact and implementation limitations of typical cyber physical attacks were analyzed, and for each established cyber physical attack, the time response of the physical system using conventional physics-based anomaly detectors was clearly explained [23]. Compared with the model proposed by the research, the methods summarized above exhibit low computational efficiency when the feature space is too large, making them less ideal for time sensitive real-time detection.

V. Conclusion

With the increasing complexity of network attack patterns, the limitations of traditional methods in network security protection have become increasingly apparent. A network security protection model grounded on improved GA-WEBP was proposed to address this issue. This model introduced the dynamic error weight and adaptive learning rate of WEBP, combined with the feature selection and hyperparameter optimization capabilities of GA, to demonstrate the effectiveness of the model in improving network security. Through experiments, the study not only verified the adaptability of the improved GA-WEBP model to dynamic changes in data during feature selection and weight updating, but also demonstrated its ability to effectively reduce false alarm rates and improve detection accuracy in practical applications. Although this research had derived important achievements in improving model performance, there were still some shortcomings. For example, the adaptability of the model to new and unknown attack patterns was still limited, which might affect its security in truly dynamic environments. Therefore, future research can focus on analyzing new feature selection and dimensionality reduction techniques to optimize the performance of the model in real-time environments, in order to further enhance the effectiveness and efficiency of network security protection.

References

[1] Fei R, Guo Y, Li J, Hu B, Yang L. An improved BPNN method based on probability density for indoor location. IEICE TRANSACTIONS on Information and Systems, 2023, 106(5): 773-785.

[2] Saminu S, Xu G, Zhang S, Kader IAE, Aliyu HA, Jabire AH, Ahmed YK, Adamu MJ. Applications of Artificial Intelligence in Automatic Detection of Epileptic Seizures Using EEG Signals: A Review. Artificial Intelligence and Applications, 2023,1(1): 11-25.

[3] Mokayed H, Quan T Z, Alkhaled L, Sivakumar V. Real-time human detection and counting system using deep learning computer vision techniques. Artificial Intelligence and Applications. 2023, 1(4): 221-229.

[4] Wei K, Zang H, Pan Y, Wang G, Shen Z. Strategic application of ai intelligent algorithm in network threat detection and defense. Journal of Theory and Practice of Engineering Science, 2024, 4(01): 49-57.

[5] Chen Z. Research on internet security situation awareness prediction technology based on improved RBF neural network algorithm. Journal of Computational and Cognitive Engineering, 2022, 1(3): 103-108.

[6] Khan M, Ghafoor L. Adversarial Machine Learning in the Context of Network Security: Challenges and Solutions. Journal of Computational Intelligence and Robotics, 2024, 4(1): 51-63.

[7] Yin L, Zhang D. The Calculation Method of the Network Security Probability of the Multi-rail Division Based on Fuzzy Inference. Mobile Networks and Applications, 2022, 27(4): 1368-1377.

[8] Memon I. A Secure and Efficient Communication Scheme with Authenticated Key Establishment Protocol for Road Networks. Wireless Personal Communications, 2015, 85(3):1167-1191.

[9] Farhan N, Rizvi S, Shabbir A, Memon I. Clustering Approaches for Efficient Radio Resource Management in Heterogeneous Networks. VFast Transactions on Software Engineering, 2021, 9(3): 68-77.

[10] Junejo M H, Ab Rahman A A H, Shaikh R A, Yusof K M, Kumar D, Memon I. Lightweight trust model with machine learning scheme for secure privacy in VANET. Procedia Computer Science, 2021, 194: 45-59.

[11] Xie Y, Wang K, Huan H. BPNN based indoor fingerprinting localization algorithm against environmental fluctuations. IEEE Sensors Journal, 2022, 22(12): 12002-12016.

[12] Zhang C, Tian Y X, Fan Z P. Forecasting sales using online review and search engine data: A method based on PCA–DSFOA–BPNN. International Journal of Forecasting, 2022, 38(3): 1005-1024.

[13] Lin J, Yang X, Zhou J, Wang G, Liu J, Yuan Y. Algorithm of BPNN-UKF based on a fusion model for SOC estimation in lithium-ion batteries. IET Power Electronics, 2023, 16(5): 856-867.

[14] Zhang P, Cui Z, Wang Y, Ding S. Application of BPNN optimized by chaotic adaptive gravity search and particle swarm optimization algorithms for fault diagnosis of electrical machine drive system. Electrical Engineering, 2022, 104(2): 819-831.

[15] Gunawan A, Thamrin S, Kuntjoro Y D, Idris A M. Backpropagation neural network (BPNN) algorithm for predicting wind speed patterns in East Nusa Tenggara. Trends in Renewable Energy, 2022, 8(2): 107-118.

[16] An K, Zhang J. Application of Genetic Algorithm in the Innovative Design of Animation Image. Journal of Electrical Systems, 2024, 20(9): 469-475.

[17] Ye F, Doerr C, Wang H, Back T. Automated configuration of genetic algorithms by tuning for anytime performance. IEEE Transactions on Evolutionary Computation, 2022, 26(6): 1526-1538.

[18] Ghezelbash R, Maghsoudi A, Shamekhi M, Pradhan B, Daviran M. Genetic algorithm to optimize the SVM and K-means algorithms for mapping of mineral prospectivity. Neural Computing and Applications, 2023, 35(1): 719-733.

[19] Feng Y, Lan C, Briseghella B, Fenu L, Zordan T. Cable optimization of a cable-stayed bridge based on genetic algorithms and the influence matrix method. Engineering Optimization, 2022, 54(1): 20-39.

[20] Maskooki A, Deb K, Kallio M. A customized genetic algorithm for bi-objective routing in a dynamic network. European Journal of Operational Research, 2022, 297(2): 615-629.

[21] Alhijawi B, Awajan A. Genetic algorithms: Theory, genetic operators, solutions, and applications. Evolutionary Intelligence, 2024, 17(3): 1245-1256.

[22] Unnisa N, Yerva M, Kurian M Z. Review on intrusion detection system (ids) for network security using machine learning algorithms. International Research Journal on Advanced Science Hub, 2022, 4(3): 67-74.

[23] Kim S, Park K J, Lu C. A survey on network security for cyber–physical systems: From threats to resilient design. IEEE Communications Surveys & Tutorials, 2022, 24(3): 1534-1573.