


Research Article

Optimization of Backpropagation Neural Network under the Adaptive Genetic Algorithm

Junxi Zhang^{1,2} and Shiru Qu ¹

¹School of Automation, Northwestern Polytechnical University, Xi'an City 710072, China

²School of Vehicle Engineering, Xi'an Aeronautical University, Xi'an City 710077, China

Correspondence should be addressed to Shiru Qu; zhangjunxi@xaau.edu.cn

Received 29 April 2021; Revised 20 May 2021; Accepted 21 June 2021; Published 6 July 2021

Academic Editor: Zhihan Lv

Copyright © 2021 Junxi Zhang and Shiru Qu. 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.

This study is to explore the optimization of the adaptive genetic algorithm (AGA) in the backpropagation (BP) neural network (BPNN), so as to expand the application of the BPNN model in nonlinear issues. Traffic flow prediction is undertaken as a research case to analyse the performance of the optimized BPNN. Firstly, the advantages and disadvantages of the BPNN and genetic algorithm (GA) are analyzed based on their working principles, and the AGA is improved and optimized. Secondly, the optimized AGA is applied to optimize the standard BPNN, and the optimized algorithm is named as OAGA-BPNN. Finally, three different cases are proposed based on the actual scenario of traffic flow prediction to analyse the optimized algorithm on the matrix laboratory (MATLAB) platform by simulation. The results show that the average error distribution of the GA-BPNN algorithm is about 1% with small fluctuation range, better calculation accuracy, and generalization performance in contrast to the BPNN. The average output error of the AGA-BPNN fluctuates around 0 and remains in a relatively stable range as a whole in contrast to that of GA-BPNN; the maximum fitness level keeps increasing during the evolution process but approaches the average value in later process, so the population diversity is hard to be guaranteed. The output error of the OAGA-BPNN fluctuates little compared with that of AGA-BPNN, and its maximum fitness continues to increase in the evolution process with guaranteed population diversity. In short, the OAGA-BPNN algorithm can achieve the best performance in terms of calculation accuracy, generalization performance, and population evolution.

1. Introduction

With the development of computer information technology and artificial intelligence (AI) in recent years, a series of new methods and new technologies have also emerged, among which the artificial neural network (ANN) is a typical representative [1, 2]. Currently, this method has been applied in many fields such as predictive decision-making, trajectory positioning, and system scheduling. Due to the applicability and excellent performance of the network model in many fields, it has become a hot spot to be studied by related experts and scholars. Under ANN, the backpropagation neural network (BPNN) is one of the typical representatives. Different from the conventional neural network (CNN) model, the BPNN model is a multilayer

forward neural network that can realize the random nonlinear mapping of corresponding input and output as well as the autonomous learning, so it has emerged in the processing and solution of nonlinear issues [3, 4]. The nonlinear issues are closely related to the actual production and life of human beings, such as vehicle fuel consumption and construction prediction. The genetic algorithm (GA) is a global optimization algorithm, taking code as the calculation object and search information as the objective function, respectively. The selection operation, crossover operation, and mutation operation related to the algorithm are all expanded in the form of probability, so the search process is very flexible [5]. At present, researchers have found that it has greater application potential to combine the BPNN with the GA.

There are many research studies on the application of machine learning in different fields. In view of the precipitation and deposition of the polar components of crude oil in the reservoir, Ahmadi proposed a feedforward neural network-based reservoir asphaltene settlement prediction model, introduced a unified particle swarm optimization (PSO) algorithm to optimize the model, and revealed the effectiveness of the model in this field [6]. Based on the prediction of asphaltene precipitation, Ahmadi and Golshadi proposed a hybrid genetic algorithm and PSO model based on a feedforward neural network, applied it in prediction of the oilfield experimental data by determining the initial weight of the neural network, and verified the validity and good performance of the model by comparison with the measured precipitation data [7]. Ahmadi et al. combined the swarm intelligence and ANN model in the monitoring of reservoir fluid behavior [8]. Shafiei et al. studied the solubility correlation of BPNN and PSO training the ANN in 11 different ionic liquids and found that the optimized model shows a significant effect in the solubility prediction [9]. Ahmadi combined the swarm intelligence and ANN and verified the proposed intelligent model based on the chemical flooding database; and, the result proved that the proposed method shows good reliability, integrity, and robustness, and it could be undertaken as an alternative model to predict chemical flooding efficiency of reservoirs [10]. Ahmadi et al. introduced the application of the least squares support vector machine in predicting the solubility of hydrogen sulfide in various ionic liquids, obtained the hyperparameters based on the genetic algorithm, and embedded in the machine learning model to provide references for correct design of the equipment in the gas desulfurization operation process [11]. Amedi et al. proposed a feedforward multilayer perceptual neural network and analyzed the solubility prediction under different conditions by establishing an adaptive neuro-fuzzy inference system and a radial basis function ANN; the results based on the mean square errors, standard deviations, and average absolute relative errors showed that the proposed ANN intelligence model shows better performance than other computational intelligence models [12]. Ahmadi et al. introduced the least squares support vector machine (SVM) method and completed a high-precision CO₂ brine solubility prediction model with uncertain parameters; after the calculation results obtained based on the model with the relevant CO₂ brine solubility experimental data were compared, it was found that the model shows high prediction accuracy with the experimental deviation less than 0.1% [13]. Moosavi et al. introduced the multilayer perceptron and radial basis function neural network to establish a prediction model for oil reservoirs; and, the prediction results showed the good prediction performance of the development network [14].

Kang et al. proposed a design evaluation system based on the BPNN for public facilities in historic districts and found that the introduction of this model can significantly improve the design efficiency of public facilities in historic districts [15]. Li et al. introduced the BPNN in prediction of tourism volume and proposed a prediction model based on the Baidu index; the BPNN was performed with the global

optimization of weights and thresholds based on the adaptive differential evolution algorithm; and, the results showed that the proposed optimization model shows higher prediction accuracy and was an effective means of predicting the number of tourists [16]. Panda and Panda proposed an optimized ANN model, analyzed and discussed its application in nonlinear system identification and nonlinear channel equalization, and found that the proposed optimized ANN based on the slope shows better performance than the BPNN [17]. Du et al. constructed a system model for the drug inventory management mode based on GA and BPNN and revealed the accuracy and thermal sensitivity of the model [18]. Imakura et al. proposed an alternate optimization method in which linear and nonlinear semi-nonnegative matrix factorization is applied to realize the optimization of the traditional BPNN and found that the optimized method has excellent performance in parallel computing [19]. Aiming at the excessive carbon emissions in climate change, Li and Gao constructed an optimized model based on BPNN, which achieves good applicability in formulating effective carbon emission policies [20]. Therefore, there are many research results for the optimization of BPNN, and it has been applied in specific fields. At present, there have been some research studies on the combination of GA and BPNN, but the optimization of BPNN based on the AGA is less reported relatively.

Based on this, AGA is introduced into the optimization of BPNN, and the AGA is optimized and improved for the optimization of BPNN in this study. In addition, the nonlinear issue of vehicle fuel consumption is combined for simulated analysis of the optimized network model. This study aims to provide a reliable reference for the optimization of BPNN and its application in complex nonlinear issues.

The main contributions of study are reflected in the following two levels: an optimized AGA-BPNN algorithm and OAGA-BPNN algorithm are proposed and applied to the analysis of vehicle fuel consumption problems; the neural network model is applied in the analysis of nonlinear problems based on the actual situation, achieving effective results.

2. Methods

2.1. Backpropagation Neural Network. The concept of ANN is based on the relevant biological activities of the human brain. The intelligent processing of knowledge can be realized through the simulation of the neuronal activities of human brain. At present, neural networks have shown excellent performance in many aspects such as face recognition, market prediction, and risk assessment [21, 22]. Among the ANN models, the BPNN belongs to a multilayer feedforward network. The realization of the neural network training process is based on the error backpropagation algorithm. It is composed of the input layer, hidden layer, and output layer [23]. The basic structure of BPNN is shown in Figure 1.

The BP algorithm is the key in the BPNN. The BP algorithm can be adopted to obtain the error through the

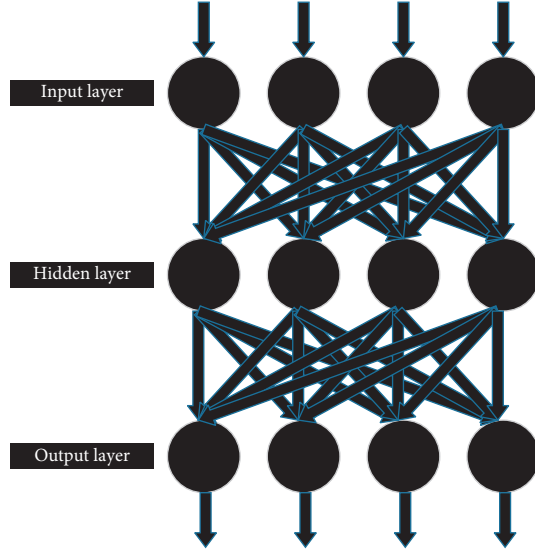


FIGURE 1: The basic structure of BPNN.

forward transmission of the input signal; then, the network can be updated through the reverse transmission of the obtained error [24]. Applying BPNN can reduce the error, showing excellent performance in the processing and solving of relatively simple nonlinear issues [25, 26]. In the forward transmission of the signal, if the signal is expressed as

$$S = (s_1, s_2, L, s_n)^T, \quad (1)$$

then when S enters the hidden layer from the input layer, the output value of the neuron corresponding to the layer structure can be obtained based on the neuron threshold and the activation function. The output value h_j can be expressed as follows:

$$h_j = f_1 \left(\sum_{i=1}^n w_{ji} s_i - \theta_j \right). \quad (2)$$

In the above equation, i and j represent the i^{th} and the j^{th} neuron in the hidden layer structure, w_{ji} refers to the connection weight between the i^{th} and the j^{th} neuron, θ_j represents the threshold of the j^{th} neuron, and f_1 stands for the activation function of the hidden layer.

Thus, the output y_l of the neuron in the output layer can be expressed as

$$y_l = f_2 \left(\sum_{j=1}^m v_{lj} h_j - \vartheta_l \right). \quad (3)$$

In the above equation, v_{ji} refers to the connection weight between the j^{th} neuron and the k^{th} neuron, ϑ_l corresponds to the threshold of the l^{th} neuron, and f_2 represents the activation function of the output layer.

The error function in the forward propagation E can be defined as follows:

$$E = \frac{1}{2} \sum_{l=1}^L (y_l - o_l)^2. \quad (4)$$

In equation (4) above, o refers to the expected output.

In the BP of the acquisition error, the correction of the corresponding neuron threshold and the corresponding connection weight of the BPNN can be expressed in the following equations:

$$v_{lj} = v_{lj} + \Delta v_{lj},$$

$$\vartheta_l = \vartheta_l + \Delta \vartheta_l,$$

$$\Delta v_{lj} = -\eta \frac{\partial E}{\partial v_{lj}} = -\eta \frac{\partial E}{\partial y_l} * \frac{\partial y_l}{\partial v_{lj}} = \eta (o_l - y_l) y_l (1 - y_l) h_j. \quad (5)$$

The correction error D_l of the output layer in the BPNN can be written as

$$D_l = (o_l - y_l) y_l (1 - y_l). \quad (6)$$

At this time, the correction error e_j of each unit in the hidden layer can be expressed as

$$e_j = \left(\sum_{l=1}^L v_{lj} * D_l \right) h_j (1 - h_j). \quad (7)$$

In short, the learning process of the BPNN is actually the reciprocating cycle of the signal forward transmission and the error BP process. The error between the actual and the expected output is reduced to reach a state of convergence by adjusting and correcting the connection weight and the threshold value. In summary, it can be found that BPNN shows a good nonlinear performance, so it has good applicability in the simulation of nonlinear systems, and it is also suitable for parallel processing of large quantities of information. However, the algorithm shows a slower convergence speed and greater volatility in the training process, so it is easy to fall into a local optimum [27]. In view of this, an AGA is introduced in this study to improve and optimize the BPNN.

2.2. Adaptive Genetic Algorithm and Its Optimization. GA is an adaptive global optimization probabilistic search algorithm tool. Based on the initial population, GA can be used to search multiple points simultaneously, which cannot only effectively reduce the search range but also avoid the local optimum [28, 29]. GA can obtain the fitness values corresponding to different individuals in the population by using the fitness function, so as to evaluate the fitness of individuals. Based on the performance of the initial population and the degree of fitness as well as the operation of the selection operator, crossover operator, and mutation operator, it can promote the generation of new populations, so as to narrow the search range to an appropriate area. After a series of iterative processes, the fitness of the population can be improved so that the optimal solution can be finally obtained.

The crossover rate of the corresponding standard GA crossover strategy and the mutation rate of the corresponding mutation strategy are both fixed constants, which

weaken the search ability of the algorithm to a reliable extent. For GA, a too large crossover rate will reduce the stability of the algorithm, while a too small crossover rate will slow down or even stagnate the evolution of the population; a too large mutation rate will bring difficulty to control the evolution of the population, while a too small mutation rate will cause the premature population, making the search for the optimal solution impossible. Based on this, AGA is introduced in this study [30], in which the mutation rate and crossover rate can be adjusted appropriately. Specifically, the crossover rate and mutation rate can be appropriately increased when the fitness value corresponding to the individual in the population is low, so as to speed up the evolution process. In addition, it can reduce the crossover rate and mutation rate appropriately when the fitness value of the individual in the population is high. The adaptive crossover rate P_C based on the AGA can be expressed as follows:

$$P_C = \begin{cases} \frac{P_{C1}(F_{\max} - F')}{F_{\max} - F_{\min}}, & F' \geq F_{\text{avg}}, \\ P_{C2'}, & F' < F_{\text{avg}}. \end{cases} \quad (8)$$

The adaptive mutation rate P_V can be calculated with the following equation:

$$P_V = \begin{cases} \frac{P_{V1}(F_{\max} - F)}{F_{\max} - F_{\min}}, & F \geq F_{\text{avg}}, \\ P_{V2'}, & F < F_{\text{avg}}. \end{cases} \quad (9)$$

In the above two equations, F' refers to the maximum fitness value of the parent individual based on the crossover operation, F refers to the fitness value of the individual based on the mutation operation, F_{\max} corresponds to the maximum fitness value of the population individual, F_{\min} refers to the minimum fitness value of a population individual, F_{avg} represents the average fitness value of the population, and P_{C1} , P_{C2} , P_{V1} , and P_{V2} are constants within the interval.

Based on the above equation, the calculation process of the crossover rate and mutation rate can slowdown the speed of population evolution. For this reason, the population evolution process is divided into different stages, and the staged evaluation parameters are introduced. The AGA can be optimized by adjusting the adaptive crossover rate, mutation rate, and the number of mutation points.

In the process of crossover calculation, individuals in the population are randomly paired. At this time, the larger fitness value can be calculated with the following equation:

$$P_{Cm} = \begin{cases} \frac{P_{C1}}{1 + e^{-IC}} * P, & F' \geq F'_{\text{avg}}, \\ P_{C2'}, & F' < F'_{\text{avg}}. \end{cases} \quad (10)$$

Then, the calculation equation of adaptive crossover rate k_c can be written as follows:

$$k_C = \frac{F' - F'_{\text{avg}}}{F_{\max} - F'_{\text{avg}}}. \quad (11)$$

In the above two equations, F' refers to the maximum fitness value of the parent individual based on the crossover operation, F_{\max} corresponds to the maximum fitness value of the population individual, and F'_{avg} represents the average fitness value of the individuals whose fitness is higher than the average value of the population, and P_{C1} and P_{C2} are constants.

In the mutation operation, fitness P_m of each individual in the population can be calculated with the following equation:

$$P_m = \begin{cases} \frac{P_{m1}}{1 + e^{-lm}} * P, & F \geq F'_{\text{avg}}, \\ P_{m2}, & F < F'_{\text{avg}}. \end{cases} \quad (12)$$

The mutation rate and the number of mutation points can be calculated with the following equations:

$$k_m = \frac{F - F'_{\text{avg}}}{F_{\max} - F'_{\text{avg}}}, \quad (13)$$

$$\text{NUM} = \text{num} * P. \quad (14)$$

In the above equations, F represents the fitness value of individuals in the population based on mutation operation, F_{\max} corresponds to the maximum fitness of individuals in the population, F'_{avg} corresponds to the average fitness value of the individuals whose fitness is higher than the average value of the population, P_{m1} and P_{m2} are two constants, and NUM refers to the largest number of mutation points.

The above optimization operation can elevate the overall fitness value of the population to a higher level quickly, and the excellent genes will not be destroyed in the later evolution so that the maximum fitness value can be continuously improved.

2.3. Optimization of Backpropagation Neural Network.

The standard BP algorithm is easy to fall into the local optimum, which can be avoided by introducing GA [31, 32]. In view of this, GA is introduced into the optimization of BPNN in this study. There are many parameters involved in the whole optimization process, so single point mutation has little effect on the evolution of the population. This is the reason why a number of mutation points are introduced for optimization of the AGA. The overall optimization mainly includes three levels of content: the improvement of connection weights and thresholds, the improvement of the neural network structure, and the improvement of the neural network learning parameters. The optimized algorithm is mainly composed of the GA part and the BPNN part. The GA part uses an improved AGA to determine the GA coding rules based on the structure of the BPNN, and the long string of parameters generated on this basis compose

the initial population. The population fitness can be improved until the GA algorithm meets the requirements of the termination condition by adjusting the crossover rate and mutation rate with the improved AGA. After the GA algorithm reaches the end, the optimal individual is decoded into a set of connection weights and thresholds corresponding to the BPNN according to the coding rules, which are defined as the initial parameters of the neural network. Then, the algorithm enters the BPNN. Based on the initial parameters, the output error can be calculated using the training dataset. If the termination condition is not met, the error will be treated with the BP, and the weight value and threshold will be adjusted and corrected accordingly. When the output error meets the relevant conditions, the network learning is terminated, so the final calculation model can be obtained. In this study, the optimized BPNN is denoted as OAGA-BPNN. The BPNN is optimized based on the optimized AGA, and the specific implementation process is shown in Figure 2.

2.4. Experimental Setting and Case Simulation. As for the prediction of the traffic flow, the traffic flow, speed, and lane occupancy rate are the most common features. The traffic flow reflects the continuous traffic flow distribution on the road, which is complex and volatile. It is not only highly correlated with time but also affected by the location and status of other traffic flows. Generally, the urban traffic flow is characterized by nonlinearity, time dynamics, spatial correlation, and uncertainty, among which the nonlinearity is embodied in the number of vehicles, driving behaviour, and traffic conditions. The time dynamics mainly reflects the temporal correlation of the traffic flow at different locations. The spatial correlation reflects the interaction between traffic flows in different spatial locations. Uncertainty reflects many external factors, such as human factors and environmental factors. The nonlinear characteristics of the traffic flow are emphasized here. To ensure the applicability of the BPNN model in nonlinear issues, the BPNN obtained based on the optimized AGA is applied to analyse the changes in vehicle fuel consumption.

To solve this problem, the input of the optimized BPNN model is the attribute data corresponding to the traffic flow, including average vehicle speed, vehicle flow, and lane occupancy rate, while the output of the model is the prediction result. Based on this, the corresponding optimized BPNN includes 18 inputs and 1 output. A single hidden layer is adopted for BPNN in this study.

A total of 100 sets of sample data are selected for this experiment, of which 80 sets are for network learning and 20 sets are for simulated analysis. The simulation experiment is implemented in the MATLAB platform [33, 34]. Two sets of cases are set up for the simulated analysis. A BPNN can be obtained with each learning through the training dataset, which refers to the traffic flow prediction model in this case. The average output of the algorithm after 5 trainings is calculated and undertaken as the output of the model.

Case 1 : The standard BPNN and GA-BPNN are compared in terms of error analysis; it is found

that they are consistent in the BPNN part; the difference lies in that the initial parameters of the latter have been optimized.

Case 2 : The GA-BPNN and AGA-BPNN are compared in terms of fitness value and average error.

Case 3 : The AGA-BPNN is compared with the OAGA-BPNN in terms of fitness value and average error.

3. Results and Discussion

3.1. Results for Simulated Analysis of Case 1. 20 sets of data for simulated analysis are adopted to compare the average output error of the BPNN and GA-BPNN, and the results are illustrated in Figure 3.

Analysis on the above figure reveals that the fluctuation trend based on the standard BPNN model is relatively large, while the average error of the GA-BPNN model fluctuates within a smaller range. In other words, it fluctuates up and down at 0, the number of points approaching 0 is more, and the average output error is distributed around 1%. Thus, the GA-BPNN model shows a better calculation accuracy than the standard BPNN model. Compared with the standard BPNN model, the GA-BPNN model shows a better generalization performance and a significantly increased calculation accuracy. It indicates that the BPNN optimized with GA can effectively avoid the tendency to fall into local optimum and overfitting in the learning process of the standard BPNN. In addition, it can explain why the GA-BPNN model shows better calculation accuracy and generalization performance in contrast to the standard BPNN model. There are also many studies on performance comparison between BPNN and GA-BPNN neural network. Wang et al. conducted corresponding analysis in the field of wave height prediction; based on the comparison of algorithm generalization performance and predictability level, they found that the GA-BPNN algorithm has better generalization performance and predictive ability and can alleviate local optimization to a certain extent [35]. Similarly, Ling et al. applied the optimized GA-BPNN algorithm to the prediction and assessment of accident consequences; after comparison on the simulation results obtained by the BPNN algorithm and the GA-BPNN algorithm, they found that the latter cannot optimize the existence of BPNN with the improved calculation speed and accuracy [36]. Based on the above analysis, it can be seen that GA-BPNN has a better performance than the BPNN algorithm, which further verifies the effectiveness of the research work in this study.

3.2. Results for Simulated Analysis of Case 2. Figure 4 illustrates the changes and distribution curves of the maximal fitness and average fitness for the GA-BPNN and AGA-BPNN during the 200 generations of population evolution.

Figure 4 illustrates that the maximal fitness of the GA-BPNN algorithm continues to increase during the initial 42 generations of evolution and decreases for several times with different degrees at the about 62 generations. When the

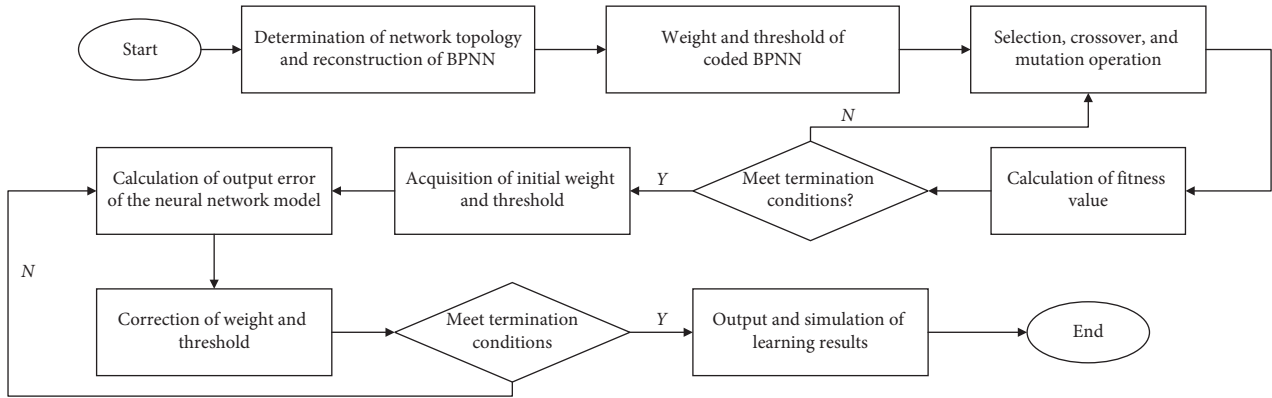


FIGURE 2: The specific implementation process of optimized BPNN.

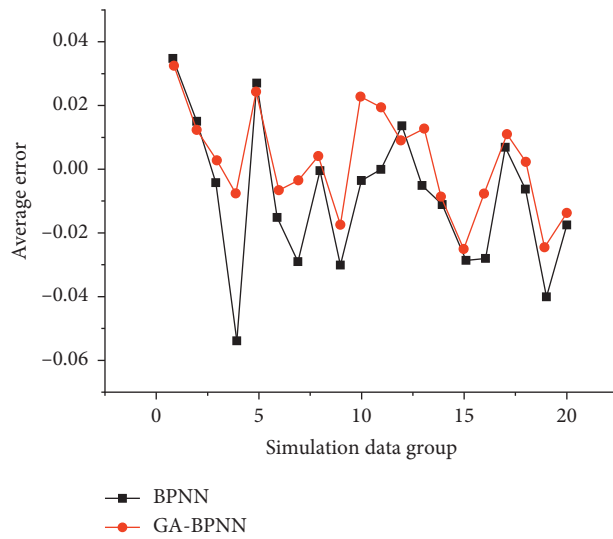


FIGURE 3: Comparison on average output error of the standard BPNN and GA-BPNN.

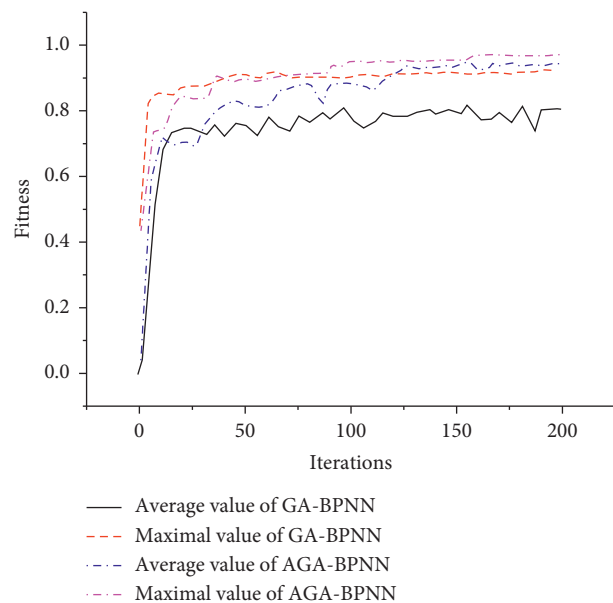


FIGURE 4: The changes and distribution curves of the maximal fitness and average fitness for the GA-PNN and AGA-BPNN.

evolution continues to the 200th generation, even the fitness value corresponding to the best individual in the population is also at a lower level. In contrast, the maximal fitness of the population in the AGA-BP algorithm is constantly increasing, and the optimal individual obtained after 200 generations of evolution is also better than that of the GA-BPNN algorithm.

Thus, the GA-BPNN algorithm with the fixed crossover rate and mutation rate is not good for the storage of good individuals in the population, and the optimal individual is even destroyed during the evolution. In contrast, the AGA-BPNN algorithm can store the good individuals perfectly, avoiding the degradation of the population during the evolution. However, comparisons on the maximal fitness curve and the average fitness value curve reveal that the final maximal fitness is close to the average value, indicating that the diversity of individuals has not been presented. Thus, it suggests that the AGA-BPNN algorithm considers the diversity poorly when it is adopted to adjust the crossover rate and mutation rate of individuals in the population.

20 sets of data for simulated analysis are adopted to compare the average output error of the GA-BPNN and AGA-BPNN, and the results are illustrated in Figure 5.

Figure 5 reveals that the average output error of the AGA-BPNN algorithm fluctuates around 0, and the overall fluctuation remains within a relatively stable range compared with the GA-BPNN algorithm, while the GA-BPNN algorithm shows a large fluctuation range for average output error, with the maximal value at 4% and the minimal value at -10.95%. Compared with Case 1, it is obvious that the AGA-BPNN algorithm shows better generalization ability and calculation accuracy than those of the GA-BPNN algorithm. Regarding the performance of the GA-BPNN algorithm and the AGA-BPNN algorithm, Han et al. applied the algorithms to the test of compensating temperature drift and found that the AGA-BPNN algorithm shows a better performance than the BPNN algorithm, thereby effectively avoiding the local optimal value [37]. This provides support for the research results obtained in this study.

3.3. Results for Simulated Analysis of Case 3. Figure 6 shows the changes and distribution curves of the maximal fitness and average fitness for the AGA-BPNN and OAGA-BPNN during the 200 generations of population evolution.

Figure 6 discloses that compared with the GA-BPNN algorithm, the average output error of the AGA-BPNN algorithm fluctuates around 0, and the overall fluctuation remains within a relatively stable range, while the GA-BPNN algorithm shows a large fluctuation range of average output error, with the maximal value at 4% and the minimal value at -10.95%. In contrast to Case 1, the AGA-BPNN algorithm has obviously better generalization ability and calculation accuracy than those of the GA-BPNN algorithm.

Thus, the OAGA-BPNN algorithm can protect excellent individuals and maintain the diversity of the population in the process of population evolution, thereby avoiding the premature compared with the AGA-BPNN algorithm. It indicates that the optimized OAGA-BPNN algorithm based

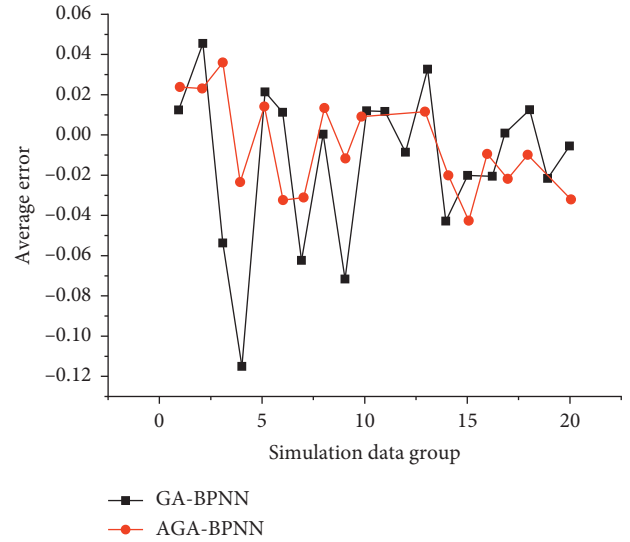


FIGURE 5: Comparison on average output error of the standard GA-BPNN and AGA-BPNN.

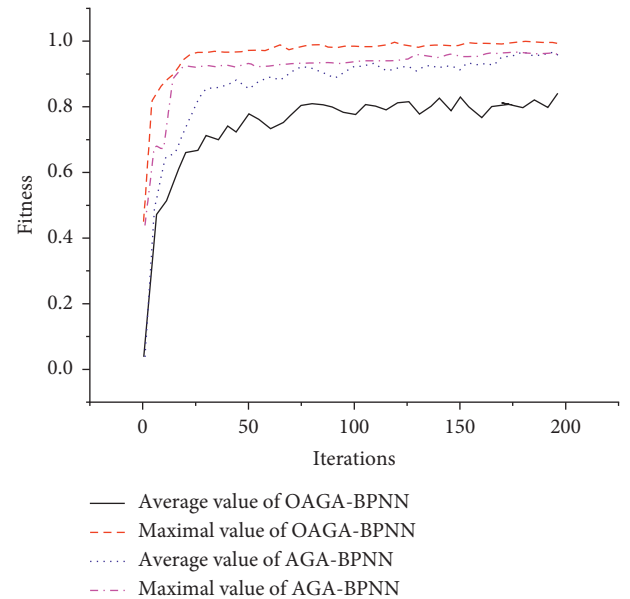


FIGURE 6: The changes and distribution curves of the maximal fitness and average fitness for the AGA-BPNN and OAGA-BPNN.

on the consideration of population diversity has wider applicability in the processing and solving of nonlinear issues.

20 sets of data for simulated analysis are adopted to compare the average output error of the AGA-BPNN and OAGA-BPNN, and the results are illustrated in Figure 7.

Figure 7 discloses that the output error fluctuation range of the OAGA-BPNN algorithm is significantly smaller than that of the AGA-BPNN algorithm, showing a smaller overall error, while the error fluctuation range of the AGA-BPNN is relatively large. The AGA-BPNN algorithm and the OAGA-BPNN algorithm are not extremely different from each other in terms of error, but greatly different from each other in the

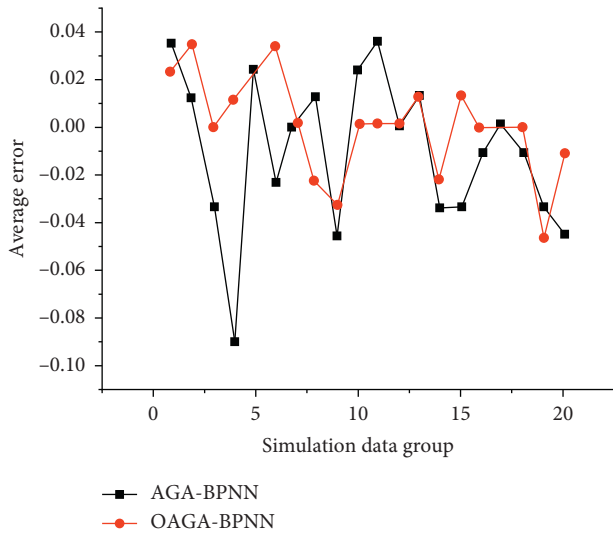


FIGURE 7: Comparison on average output error of the standard AGA-BPNN and OAGA-BPNN.

overall accuracy. In short, the OAGA-BPNN algorithm optimized on the basis of AGA not only improves the search ability of the calculation model but also enhances the generalization ability and calculation accuracy of the model with the significantly reduced error in the whole process. In this study, the AGA-BPNN is improved and optimized into the OAGA-BPNN algorithm. The former combines the excellent performance of the BPNN algorithm and the GA algorithm, and the latter introduces the crossover rate and mutation rate on this basis so that the optimized algorithm can be better applied to the actual situation.

4. Conclusions

The diversity of the individual in the populations is included in this study, and the BPNN model is improved and optimized based on the AGA algorithm. In addition, performance of the neural network model is simulated and verified in combination with traffic flow prediction. Simulated analysis results of three different cases reveal that the BPNN model based on the optimized AGA algorithm shows excellent performance with significantly improved calculation accuracy and good generalization ability. Besides maintaining the optimal individual, it can effectively improve the maximal fitness of the individual in the population, guarantee the diversity of individuals, and significantly reduce the output error of the entire calculation process. This study provides an experimental reference for the performance optimization and improvement of the BPNN model.

In the case experiment setting, traffic flow prediction under consideration is a relatively ideal situation in this study, but the actual scene is more complicated, so it is obviously difficult to analyse and calculate the abnormal data. In addition, it mainly considers from the perspective of AGA and the adjustment of the crossover rate and mutation rate, as well as the diversity of the population for optimization of the BPNN. The consideration of parameters' optimization is not comprehensive enough due to limitation of

the article length. In the follow-up research, more influencing factors will be considered into the calculation model, more parameters will be included based on the AGA to discuss the optimization scheme further, and other methods will be considered to improve the BPNN further.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] B. Suffoletto, P. Gharani, T. Chung, and H. Karimi, "Using phone sensors and an artificial neural network to detect gait changes during drinking episodes in the natural environment," *Gait and Posture*, vol. 60, pp. 116–121, 2018.
- [2] A. H. A. Al-Waeli, K. Sopian, H. A. Kazem et al., "Comparison of prediction methods of PV/T nanofluid and nano-PCM system using a measured dataset and artificial neural network," *Solar Energy*, vol. 162, pp. 378–396, 2018.
- [3] L. Wu, Y. Yang, M. Maheshwari, and N. Li, "Parameter optimization for FPSO design using an improved FOA and IFOA-BP neural network," *Ocean Engineering*, vol. 175, pp. 50–61, 2019.
- [4] J. H. Alcantara and J.-S. Chen, "Neural networks based on three classes of NCP-functions for solving nonlinear complementarity problems," *Neurocomputing*, vol. 359, pp. 102–113, 2019.
- [5] H. Aziza and S. Krichen, "Bi-objective decision support system for task-scheduling based on genetic algorithm in cloud computing," *Computing*, vol. 100, no. 2, pp. 65–91, 2018.
- [6] M. A. Ahmadi, "Neural network based unified particle swarm optimization for prediction of asphaltene precipitation," *Fluid Phase Equilibria*, vol. 314, pp. 46–51, 2012.
- [7] M. Ali Ahmadi and M. Golshadi, "Neural network based swarm concept for prediction asphaltene precipitation due to natural depletion," *Journal of Petroleum Science and Engineering*, vol. 98–99, no. 6, pp. 40–49, 2012.
- [8] M. A. Ahmadi, M. Ebadi, and A. Yazdanpanah, "Robust intelligent tool for estimating dew point pressure in retrograded condensate gas reservoirs: application of particle swarm optimization," *Journal of Petroleum Science and Engineering*, vol. 123, pp. 7–19, 2014.
- [9] A. Shafiei, M. A. Ahmadi, S. H. Zaheri, A. Baghban, A. Amirfakhrian, and R. Soleimani, "Estimating hydrogen sulfide solubility in ionic liquids using a machine learning approach," *The Journal of Supercritical Fluids*, vol. 95, pp. 525–534, 2014.
- [10] M. A. Ahmadi, "Developing a robust surrogate model of chemical flooding based on the artificial neural network for enhanced oil recovery implications," *Mathematical Problems in Engineering*, vol. 2015, pp. 1–9, 2015.
- [11] M.-A. Ahmadi, B. Pouladi, Y. Javvi, S. Alfkhan, and R. Soleimani, "Connectionist technique estimates h₂s solubility in ionic liquids through a low parameter approach," *The Journal of Supercritical Fluids*, vol. 97, pp. 81–87, 2015.
- [12] H. R. Amedi, A. Baghban, and M. A. Ahmadi, "Evolving machine learning models to predict hydrogen sulfide

- solubility in the presence of various ionic liquids,” *Journal of Molecular Liquids*, vol. 216, pp. 411–422, 2016.
- [13] A. M. Ahmadi and A. Alireza, “Applying a sophisticated approach to predict CO₂ solubility in brines: application to CO₂ sequestration,” *International Journal of Low-Carbon Technologies*, vol. 11, no. 3, pp. 325–332, 2016.
- [14] S. R. Moosavi, D. A. Wood, M. A. Ahmadi, and A. Choubineh, “Ann-based prediction of laboratory-scale performance of co₂-foam flooding for improving oil recovery,” *Natural Resources Research*, vol. 28, no. 4, pp. 1619–1637, 2019.
- [15] L. Kang, W. Zhu, and G. G. Yoon, “VR Design of public facilities in historical blocks based on BP neural network,” *Neural Processing Letters*, 2020.
- [16] S. Li, T. Chen, L. Wang, and C. Ming, “Effective tourist volume forecasting supported by pca and improved bpnn using baidu index,” *Tourism Management*, vol. 68, pp. 116–126, 2018.
- [17] S. Panda and G. Panda, “Performance evaluation of a new BP algorithm for a modified artificial neural network,” *Neural Processing Letters*, vol. 51, no. 4, 2020.
- [18] M. Du, J. Luo, S. Wang, and S. Liu, “Genetic algorithm combined with BP neural network in hospital drug inventory management system,” *Neural Computing and Applications*, vol. 32, no. 7, pp. 1981–1994, 2020.
- [19] A. Imakura, Y. Inoue, T. Sakurai, and Y. Futamura, “Parallel implementation of the nonlinear semi-NMF based alternating optimization method for deep neural networks,” *Neural Processing Letters*, vol. 47, no. 3, pp. 815–827, 2018.
- [20] W. Li and S. Gao, “Prospective on energy related carbon emissions peak integrating optimized intelligent algorithm with dry process technique application for China’s cement industry,” *Energy*, vol. 165, pp. 33–54, 2018.
- [21] A. Amo Baffour, J. Feng, and E. K. Taylor, “A hybrid artificial neural network-GJR modeling approach to forecasting currency exchange rate volatility,” *Neurocomputing*, vol. 365, pp. 285–301, 2019.
- [22] A. Saeed and A. Rashid, “Development of core monitoring system for a nuclear power plant using artificial neural network technique,” *Annals of Nuclear Energy*, vol. 144, Article ID 107513, 2020.
- [23] Y. J. Liang, C. Ren, H. Y. Wang, Y. B. Huang, and Z. T. Zheng, “Research on soil moisture inversion method based on GA-BP neural network model,” *International Journal of Remote Sensing*, vol. 40, no. 5–6, pp. 2087–2103, 2019.
- [24] B. S. Alpert, D. Quinn, M. Kinsley, T. Whitaker, and T. T. John, “Accurate blood pressure during patient arm movement,” *Blood Pressure Monitoring*, vol. 24, no. 1, pp. 42–44, 2019.
- [25] X. Tao, H. Zhang, Y. Wang et al., “VVBP-tensor in the FBP algorithm: its properties and application in low-dose CT reconstruction,” *IEEE Transactions on Medical Imaging*, vol. 39, no. 3, pp. 764–776, 2020.
- [26] Z. Asghar Sayed and B. P. Ng, “Aperiodic geometry design for DOA estimation of broadband sources using compressive sensing,” *Signal Processing*, vol. 155, pp. 96–107, 2019.
- [27] F. He and L. Zhang, “Prediction model of end-point phosphorus content in BOF steelmaking process based on PCA and BP neural network,” *Journal of Process Control*, vol. 66, pp. 51–58, 2018.
- [28] H. Prasanchum and A. Kangrang, “Optimal reservoir rule curves under climatic and land use changes for Lampao Dam using Genetic Algorithm,” *KSCE Journal of Civil Engineering*, vol. 22, no. 1, pp. 351–364, 2018.
- [29] C. Pragadeesh, R. Jeyaraj, K. Siranjeevi, R. Abishek, and G. Jeyakumar, “Hybrid feature selection using micro genetic algorithm on microarray gene expression data,” *Journal of Intelligent and Fuzzy Systems*, vol. 36, no. 3, pp. 2241–2246, 2019.
- [30] A. K. Shukla, “Multi-population adaptive genetic algorithm for selection of microarray biomarkers,” *Neural Computing and Applications*, vol. 32, no. 15, pp. 11897–11918, 2020.
- [31] M. Zhang, J. Li, L. Kang et al., “Machine learning-guided design and development of multifunctional flexible Ag/poly (amic acid) composites using the differential evolution algorithm,” *Nanoscale*, vol. 12, no. 6, pp. 3988–3996, 2020.
- [32] L. Jiao, Y. Li, M. Gong, and X. Zhang, “A novel hybrid meta-heuristic algorithm based on the cross-entropy method and firefly algorithm for global optimization,” *Entropy*, vol. 21, no. 5, p. 494, 2019.
- [33] G. Silvia, H. P. Christian, and J. G. Nagy, “IR tools: a MATLAB package of iterative regularization methods and large-scale test problems,” *Numerical Algorithms*, vol. 81, pp. 773–811, 2018.
- [34] M. Karpov, M. H. P. Pfeiffer, J. Liu, A. Lukashchuk, and T. J. Kippenberg, “Photonic chip-based soliton frequency combs covering the biological imaging window,” *Nature Communications*, vol. 9, no. 1, p. 1146, 2018.
- [35] W. Wang, R. Tang, C. Li, P. Liu, and L. Luo, “A BP neural network model optimized by mind evolutionary algorithm for predicting the ocean wave heights,” *Ocean Engineering*, vol. 162, pp. 98–107, 2018.
- [36] Y. Ling, C. Chai, W. Hou, D. Hei, S. Qing, and W. Jia, “A new method for nuclear accident source term inversion based on GA-BPNN algorithm,” *Neural Network World*, vol. 29, no. 2, pp. 71–82, 2019.
- [37] Z. Han, L. Hong, J. Meng, Y. Li, and Q. Gao, “Temperature drift modeling and compensation of capacitive accelerometer based on AGA-BP neural network,” *Measurement*, vol. 164, Article ID 108019, 2020.