Research on Power Load Prediction Based on Elman Network with Improved Particle Swarm Optimization

Shijian Liu\textsuperscript{1a}, Li Zhang\textsuperscript{2b*}, Huiwen Xia\textsuperscript{1c}

\textsuperscript{1}School of Automation, \textsuperscript{2}School of Mechanical and Electronic Engineering, Wuhan University of Technology, Wuhan 430070, China
\textsuperscript{a}303031@whut.edu.cn, \textsuperscript{b}zliss0520@whut.edu.cn, \textsuperscript{c}674562685@qq.com

Abstract. Because electricity is so difficult to store, the electric power department needs to store enough power in advance, whether customers use it or not. To lower the cost of the electric power department in this aspect, it is essential to realize high-precision power load forecasting. The Elman neural network (ElmanNN) prediction model based on the improved Particle Swarm Optimization (IPSO) algorithm is constructed. The Particle Swarm Optimization (PSO) algorithm is brought in, and the learning factor of the algorithm is optimized for the limitation that the PSO algorithm can easily obtain the local optimal solution. Finally, the improved algorithm is applied to the ElmanNN. The improved ElmanNN solves the problems of slow training speed and is effortless to obtain the local minimum. Finally, simulation experiments verify that the ElmanNN model based on the improved PSO algorithm has a smaller prediction error in power load prediction.

1. Introduction
In recent years, China's economic development has achieved unprecedented achievements in history. As the foundation and important pillar of the national economy, the electric power industry contributes greatly to economic development. However, at present, the contradiction between the supply and demand of electricity is highlighted. The power supply cannot fully meet the needs of all types of users, and even restricts economic development to a certain extent. Since it is difficult to achieve mass storage of electric energy, and the demand for electric power is not constant in all walks of life, it is hoped to provide users with an appropriate amount of electric energy according to the prediction of power load, so that the power generation of the power system is in a dynamic balance with the demand of users, and meets the needs of various users\cite{1-2}.

The key point of power load prediction is to construct a suitable mathematical model. Compared with traditional mathematical models, ElmanNN can learn historical data and approximate arbitrary complex nonlinear relations. However, the Elman network also has obvious deficiencies and limitations, such as slow training speed, low prediction accuracy and easy to obtain local optimum. The large error in load forecasting will lead to a sharp increase in the operating cost of power production and power system \cite{3}. Therefore, accurate load forecasting makes practical sense. In this paper, the ElmanNN is optimized based on the IPSO algorithm, which better solves the time-varying and nonlinear problems of the power load brought by the complexity of the power system, and significantly improves the accuracy of power load prediction.
2. Elman Neural Network with improved Particle Swarm Optimization

2.1. The structure of neural networks
ElmanNN was first proposed for speech processing problems. It consists of four parts, and the relationship between each layer is shown in Fig. 1.

From the perspective of information flow in the operation of a neural network, it usually includes two types: feedforward network and feedback network. The connection of the input layer, hidden layer and output layer of ElmanNN is similar to the feedforward network, and the connection layer is connected to the hidden layer, which can remember the output from the moment before in the hidden layer and automatically link to the input, so as to achieve the effect of storage and delay [4]. Therefore, ElmanNN is equivalent to a BP neural network with feedback, which is more sensitive to historical data and has a stronger ability to facilitate the realization of dynamic modeling.

According to Fig. 1, the mathematical model expression corresponding to ElmanNN is shown as follows.

\[ x_c(k) = x(k - 1) \]  
\[ x(k) = f(w^1 x_c(k) + w^2 (u(k - 1))) \]  
\[ y(k) = g(w^3 x(k)) \]

In (1), (2) and (3), \( u \) is the input vector. \( y \) is the output vector. \( x \) is the output vector of the hidden layer. The output vector of the undertaking layer \( x_c \) is n-dimension. \( w^1 \), \( w^2 \) and \( w^3 \) are connection weights of hidden layer to output layer, input layer to hidden layer and undertaking layer to hidden layer respectively [5]. \( g(*) \) is the activation function of the neuron in the output layer. \( f(*) \) is the activation function of the neuron in the hidden layer.

ElmanNN adopts the BP algorithm to optimize the weights, and the quadratic sum of errors is as follows.

\[ E(\omega) = \sum_{i=1}^{n} (y_k(\omega) - \hat{y_k}(\omega))^2 \]  

In (4), \( \hat{y_k}(\omega) \) is the target input vector[4].
2.2. Particle Swarm Optimization algorithm
The idea of PSO stems from the research on the foraging behavior of birds. Through the information sharing of individuals in the flock, the whole bird flock produces a search process from random to ordered, so as to find the location with the largest amount of food [6]. James Kennedy and Russell Eberhart built a simplified algorithm model based on this, and the normalized formula is as follows.

\[ V_{i_d}^{k+1} = V_{i_d}^k + c_1 r_1 (P_{i_d}^k - X_{i_d}^k) + c_2 r_2 (P_{gd}^k - X_{i_d}^k) \]  
\[ X_{i_d}^{k+1} = X_{i_d}^k + V_{i_d}^{k+1} \]  

(5)

(6)

In (5) and (6), \( c_1 \) and \( c_2 \) are learning factors. \( r_1 \) and \( r_2 \) are random numbers between 0 and 1. \( X_{i_d}^k \) is the position of particle \( i \) in the \( k \)-th evolutionary iteration in the \( d \)-dimensional space. \( V_{i_d}^k \) is the velocity of particle \( i \) during the \( k \)-th evolutionary iteration in the \( d \)-th dimension space. \( P_{i_d}^k \) and \( P_{gd}^k \) are the positions corresponding to individual extreme values of particles in \( d \)-dimensional space and global extreme values in \( d \)-dimensional space respectively [5].

2.3. Improved Elman neural network
For the PSO algorithm, it has an outstanding global search capability, but when particle swarm searches near the global optimal position, its search process will start to slow down, all particles are near the global optimal position and it is difficult to jump out of this state. The gradient descent training method adopted by the BP algorithm will probably lead to the local minimum of ElmanNN [7]. So this paper introduces the PSO algorithm and combines it with the BP algorithm of ElmanNN to fully concentrate the advantages of both.

At the same time, aiming at the shortcomings of PSO easily getting local optimality and slow training speed of BP algorithm, PSO algorithm is further improved. Based on the basic algorithm, make \( c_1 \) and \( c_2 \) change with the increase of the evolutionary algebra [8], in order to encourage particles to move in the whole search space in the early stage of optimization. Moreover, the convergence speed of the algorithm to the optimal solution is accelerated, and the optimal solution is found effectively. \( c_1 \) and \( c_2 \) are shown as follows.

\[ c_1 = R_1 + \frac{R_2 \times t}{T_{max}} \]  
\[ c_2 = R_3 - \frac{R_4 \times t}{T_{max}} \]  

(7)

(8)

In (7) and (8), \( R_1, R_2, R_3 \) and \( R_4 \) are the initial values set. \( t \) and \( T_{max} \) are the current evolutionary algebra and the maximum evolutionary algebra respectively.

The IPSO algorithm is introduced to optimize the threshold and weight of ElmanNN, and then the BP algorithm is combined to predict. The steps are as follows.

Step1: Normalize the sample data to make its value between (0,1).
Step2: The connection weights and thresholds of each layer of ElmanNN are recorded as individual particles and initialized [9]. Then by matrix coding, the connection weights are denoted as a matrix.
Step3: Train ElmanNN, that is, optimize its connection weights with the PSO algorithm.
Step4: Calculate the fitness function value according to the position of the current particle. The calculation formula is (9).

\[ F = k \sum_{i=1}^{n} |y_i - o_i| \]  

(9)
In (9), \( k \) is the adjustable constant coefficient. \( i \) is the \( i \)-th node. \( n \) is the total number of nodes. \( o_i \) is the actual output value. \( y_i \) is the theoretical output value \([10]\).

Step5: Compare the current position of the particle swarm and each particle with the optimal position in the ongoing search process. If the current position is superior to the optimal position, the current position is updated to its current optimal position. Then, the position and velocity of particles are changed according to the velocity update and position update formula of the PSO algorithm to determine whether the fitness function value meets the requirements of the problem. If not, the cycle is continued until the condition is met, so as to obtain the optimal weight and threshold value.

Step6: After the training of ElmanNN is completed, the best position of the output particle swarm is the connection weight of each layer of the optimal ElmanNN. Then, the trained network uses the sum of error squares to calculate the error and update the weight until it meets the requirements of the problem. Finally, the simulation results are obtained \([11]\).

The specific training process of the ElmanNN model based on the IPSO algorithm is displayed in Fig. 2.

3. Simulation and Analysis

According to the improved ElmanNN prediction model, the power load prediction is realized by programming in MATLAB.

1) The used data comes from the power load data of a factory in January 2014 provided in the test questions of the 9th Electrical Engineering Mathematical Modeling Contest organized by the Chinese Society of Electrical Engineering. The factory’s daily electricity consumption is usually on the order of thousands of kilowatt-hours, and peak consumption occurs between 9 and 19 hours a day. This paper obtains the power consumption data from 9 to 19 hours in 9 days. For the first 8 days, the data of
consecutive 3 days were taken as network training sample data, and the data of the fourth day was used as the objective vector so that a total of 6 groups of training samples were obtained [12]. The electricity consumption data of the 9th day was taken as the test samples to calculate the prediction errors.

2) The performance of the neural network is associated with how many the hidden layer neurons are. After many experiments, when the hidden layer neurons are 11, the prediction effect is more ideal.

3) Initialize the particle swarm parameters, including particle population size, particle learning factor, maximum evolution algebra, etc. After reading the sample data in MATLAB, the newelm() function is used to establish ElmanNN to realize the prediction of power load and draw the error graph of prediction results.

In order to facilitate comparison, this paper used the original ElmanNN model, PSO-ElmanNN prediction model and IPSO-ElmanNN prediction model to predict power load respectively. The error of prediction results is displayed in Fig.3, Fig.4 and Fig.5.

To analyze the error of ElmanNN prediction results before and after improvement, MAE, SSE and MSE are used in this paper, and the calculation results of power load prediction errors of the ElmanNN before and after improvement are calculated and analyzed, as shown in TABLE 1.

![Fig.3 Prediction error of ElmanNN](image1)

![Fig.4 Prediction error of PSO-ElmanNN](image2)

![Fig.5 Prediction error of IPSO-ElmanNN](image3)

**TABLE 1. Results of three kinds of prediction errors**

| Prediction Model | MAE   | MSE   | SSE   |
|------------------|-------|-------|-------|
| ElmanNN          | 0.0154| 0.0930| 0.1692|
| PSO-ElmanNN      | 0.0043| 0.0615| 0.0468|
| IPSO-ElmanNN     | 0.0016| 0.0319| 0.0177|

By analyzing the data in TABLE 1, it is found that the MAE, MSE and SSE of power load prediction results after the improved ElmanNN are smaller than those before the improvement. Taking the mean square error as an example, the mean square error of the PSO-ElmanNN model to predict power load is 66.13% of ElmanNN, and the MSE of the IPSO-ElmanNN model to predict power load is 34.30% of ElmanNN, indicating that the ElmanNN based on the improved PSO algorithm has higher prediction accuracy and more stable forecasting ability. The performance index is better than the ElmanNN prediction model before the improvement.

4. Conclusion

The complex structure of the power system endows power load with time-varying, nonlinear and uncertain characteristics. To solve the power load prediction, an ElmanNN model based on the improved
PSO algorithm is established. Since the original PSO algorithm has insufficient local optimization capacity and is prone to premature puberty, this paper optimizes the learning factor of the algorithm and combines it with the traditional ElmanNN. Through subsequent simulation experiments, the ElmanNN based on the improved PSO algorithm is successfully verified to have less error and a better effect on power load prediction. If this model is applied to actual power load forecasting, it will lower the cost of power production and power system operation, which has a certain practical significance.

References

[1] Wang Yue. Research on evaluation model of new and old kinetic energy conversion based on electric power and economic indicators [D]. North China Electric Power University (Beijing), 2020.

[2] Leaf Aster, Zhang Jian, Lei Xiaolan, et al. Research on the driving effect of electric power construction investment on national economy and employment. Investment Research, 2020, 39(7): 97-110

[3] Song Mingda, Zhao Yuhong. Research on Power load Forecasting based on Elman neural network [J]. Science & Technology Wind, 2020(11): 200-201.

[4] Qian Yi-zhao, Chen Liang. Application of Elman Neural Network in Power Load Forecasting [J]. China Electrotechnical Technology, 2019(14): 55-56.

[5] Zhang Qiqi. Improved Particle Swarm Optimization Elman Neural Network for Traffic Flow Prediction [D]. Chang‘an University, 2019.

[6] Jiang Yunteng, Li Ping. Short-term Load Prediction Based on Improved Particle Swarm Optimization Neural Network [J]. Electrical Technology, 2018, 19(2): 87-91.

[7] Gao Jiayao, DING Yuanming. Improved Elman Neural Network for Node Traffic Prediction in Ad Hoc Network [C]// Proceedings of the 3rd China Command and Control Conference (Vol. 1) : National Defense Industry Press, 2015:171-176.

[8] Tian Zhongda. Approach for short-term wind power prediction via kernel principal component analysis and echo state network optimized by improved particle swarm optimization algorithm[J]. Transactions of the Institute of Measurement and Control,2021,43(16).

[9] Bi Ya,Lam Anthony,Quan Huiqun,Liu Hui,Wang Cunfa. A Comprehensively Improved Particle Swarm Optimization Algorithm to Guarantee Particle Activity[J]. Russian Physics Journal,2021,64(5).

[10] Davoud Sedighizadeh,Ellipsis Masehian,Mostafa Sedighizadeh,Hossein Akbaripour. GEPSO: A new generalized particle swarm optimization algorithm[J]. Mathematics and Computers in Simulation,2021,179.

[11] Liu Biao,Zhao Yufei,Wang Wenbo,Liu Biwang. Compaction Density Evaluation Model of Sand-Gravel Dam Based on Elman Neural Network With Modified Particle Swarm Optimization[J]. Frontiers in Physics,2022.

[12] Song Kun,Lv ShengKai,Hu Die,He Peng,Feng Yun-Wen. Software Defect Prediction Based on Elman Neural Network and Cuckoo Search Algorithm[J]. Mathematical Problems in Engineering,2021,2021.