Short-term power load forecasting based on improved back propagation neural network

Qirui Wang*, Cheng Peng, Yujun Zhou
Xinjiang Normal University, Urumqi, Xinjiang, China

ABSTRACT

To solve the problem of short-run power load forecasting, this article proposes a model using particle swarm optimization (PSO) to adjust the parameters of the backpropagation (BP) neural network, namely the PSO-BP model. Based on this, the GPSO-BP-NN short-term power load forecasting model is constructed. For the sake of verifying the performance of GPSO-BP-NN, actual data from a certain region in China is selected for experimentation. In view of the analysis of the fitness function outcome, by comparing the prediction results of GPSO-BP-NN, PSO-BP-NN, and BP-NN models, it is found that the mean absolute error of the GPSO-BP-NN model is 2.21%, which is lower than the 2.39% of the PSO-BP-NN and the 3.53% of the BP-NN. Through the analysis of prediction accuracy, algorithm comparison, and time cost, GPSO-BP-NN is superior to the other two prediction models, proving the efficiency of the improved algorithm.

Keywords: Neural network, particle swarm optimization algorithm, power load forecasting, time cost

1. INTRODUCTION

Along with the process of economic development, the problem of accurate load forecasting in power grid construction has attracted more and more attention. Short-Term Load Forecasting (STLF) is intended to predict the load of the system in advance, which is expressed by the sum of all the consumer loads at the same time. Forecasting error has a significant impact on profit, market share and shareholder value. The STLF's forecast step is usually one hour or less, and the forecast range is usually limited to tomorrow, but the forecast horizon is usually limited to a week. Nevertheless, the current research results on electric load forecasting have become more and more abundant[1-3]. Artificial Neural Network (ANN), due to the fact that its excellent nonlinear mapping ability, generalization ability and automatic learning ability, has been proved to be extensively practical in the field of engineering. At present, one of the most extensively utilized Network structures is BP Neural network (Back Propagation Neural Network)[4]. However, due to the BP method is very exquisite to the option of network topology and scale, the calculation results may spill, or produce fluctuations near the optimal solution. At the same time, the convergence speed of BP is also very exquisite to the option of initial weights. If the incipient weight is not selected properly, the optimization result may fall into the local optimal solution[5,6].

In this article, an adaptive PSO is proposed to take full advantage of the BP for short-run load forecasting. The BP model is used to shine different charge order components to various dimension, and the similarity coefficient is given based on the maximum size and minimum size load of the approximate part. The conclusions display that the proposed PSO-BP can predict the power load more accurately.

2. BASIC THEORY

2.1 Back propagation neural network

The back spread neural network is the kind of feed forward network with three or more plies, including an output layer, hidden layer(s), and input layer[7]. The structure of the back propagation neural network separated into two main parts: feed forward and back-propagation. In the feed forward process, entering data from the input layer, passes through one or more hidden layers, after that propagates to the output layer. In the backward propagation process, when there is an issue among the real output and the expected output, the issue is propagated backwards through each layer's weights. During

*461091557@qq.com

International Conference on Optics, Electronics, and Communication Engineering (OECE 2024), edited by Yang Yue, Proc. of SPIE Vol. 13395, 133952R · © 2024 SPIE · 0277-786X · Published under a Creative Commons Attribution CC-BY 3.0 License · doi: 10.1117/12.3046151

this process, the weights of each layer's neurons are modified to reduce the error[8]. Figure 1 displays the basic structure of the back-propagation neural network.

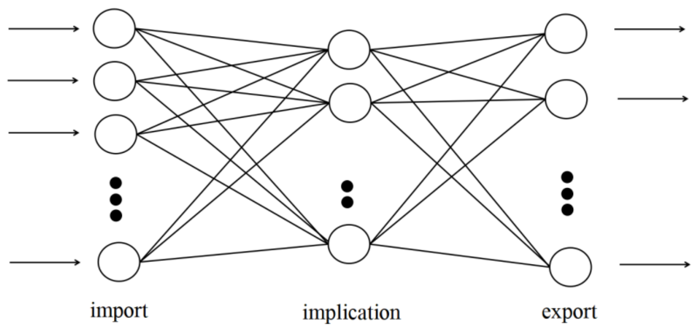

Figure 1. Structure of BP neural network.

2.2 Particle swarm optimization algorithm

Particle swarm optimization (PSO) is n best-of-breed technique that simulates the social behavior of animal groups such as flight or shoal or group to search the optimal solution to a problem. The core idea of this algorithm is to utilize the cooperation and information sharing among individuals in the bunch to iteratively search for the best-of-breed solution[9]. The particle position formula update as follows:

$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1) \tag{1}$$

t and $X_{ij}(t)$ represent the iteration count and the current location of the granules at iteration t, respectively. $V_{ij}(t)$ means the velocity of the granules at iteration t. When the calculation reaches the utmost times of iterations, it will stop running regardless of whether the global optimum has been found or not.

In the PSO algorithm, the inertia weight w has a part in equipoise the partial and holistic search abilities[10]. Specifically, the inertia weight w introduces randomness into the velocity update of particles, which helps particles escape local optima and increases the possibility of finding the global optimum. If the value of w is too large, particles will focus too much on the global optimum, resulting in the loss of local search ability and the algorithm getting trapped in local optima prematurely. On the other side, if the value of w is too small, particles will focus too much on local optima, leading to the loss of global search ability and the algorithm being unable to find the true global optimum. The formula for counting the inertia weight is shown in equation (2), where f_{ave} displays the fitness function.

$$\omega = \begin{cases} \omega_{\max} - \dfrac{(\omega_{\max} - \omega_{\min})(f - f_{ave})}{f_{\max} - f_{ave}} & f \geq f_{ave} \\ \omega_{\max} & f < f_{ave} \end{cases} \tag{2}$$

2.3 Ant colony algorithm optimizes particle swarm algorithm

Ant colony optimization (ACO) is a novel simulated evolution algorithm that simulates the behavior of ants searching for paths in nature[11]. By using ACO to improve the granules swarm best-of-breed (PSO) algorithm in finding the first-best solution, the search strategy can be continuously adjusted according to the current search situation to avoid getting trapped in local optima. Combining ACO with PSO allows for the utilization of the collective intelligence and information sharing mechanism of ACO to enhance the performance of the PSO algorithm, while also leveraging the fast convergence and global search capability of PSO to compensate for the shortcomings of ACO in handling complex problems[12]. The steps for combining the algorithms are as follows:

Step 1: Design the basic parameters of the hybrid algorithm, including the maximum number of iterations, minimum error, population size, etc.

Step 2: Determine the fitness values based on the fitness function.

Step 3: Sort the fitness values and retain particles.

Step 4: If the general number of maintained granules equals, proceed to the next step; or else, return to Step 2.

Step 5: Update the particle information based on equations (1) and (2).

Step 6: If the error is less than or the number of iterations reaches, output the collective extremum and particle information; otherwise, return to Step 3.

3. POWER LOAD FORECASTING MODEL OF GPSO-BP NEURAL NETWORK

3.1 Data preprocessing

In the GPSO-BP prediction model, preconditioning of historical data is an important step, including the following steps: In data cleaning, handling and removing abnormal values, missing values, and erroneous data in the historical load data. In data normalization, converting all data to a range of 0-1. The normalization for the primitive power charge data is as follows:

$$X_i^{'} = \frac{X_t - X_{\min}}{X_{\max} - X_{\min}} \tag{3}$$

$X_{\max}, X_{\min}$ represents the maximum size and minimum size power load, X_i, t represents the data to be normalized, and the charge time. The load prediction results are evaluated utilizing the relative mistake formula, which is:

$$error = \frac{|y_i - y_k|}{y_k} \tag{4}$$

error represents the relative error, y_i, y_k represents the predicted and actual load values.

3.2 The neural network model of GPSO-BP

The GPSO-BP-NN model combines the advantages of particle swarm optimization (PSO) and error backpropagation (BP). The characteristics of the GPSO-BP-NN include: PSO algorithm can quickly find the optimal solution, while BP neural network can achieve high-precision prediction; PSO algorithm has adaptability, automatically adjusting the search strategy to find the optimal solution; Both PSO algorithm and BP neural network have good robustness and can handle noise and abnormal data.

The GPSO algorithm continuously updates the optimal velocity $(gbest)$ and position $(pbest)$ of particles in the space. The optimal solution obtained by this algorithm is the best-of-breed weights and thresholds of the BP. Multiple iterations of training are performed to minimize the sum of squared mistakes in the output. The particles update their velocity and position as follows:

$$v_{i+1} = \omega * v_i + c_1 r_1 (pbest_i - x_i) + c_2 r_2 (gbest_i - x_i) \tag{5}$$

$$x_{i+1} - x_i + v_{i+1} \tag{6}$$

ω is the inertia weight factor, c_1, c_2 is the learning factor, and r_1, r_2 is a equivalently distributed randomized number between 0 and 1. The fitness function is calculated based on the squared absolute randomized among the predicted and exportation meanings of the samples.

$$f = \frac{1}{M} \sum_{k=1}^{M} \|Y_k - T_k\|^2 \tag{7}$$

f is the fitness function, M is the number of training samples, Y_k is the predicted value, and T_k is the actual output value.

The topological structure adopted by the GPSO-BP includes an input layer, a hidden layer, and an output layer. The input layer gets signals, the hidden layer handles and transforms the signals, and the output layer manufactures the output results. Each node represents a neuron in the network. In the input layer, each node corresponds to an input signal, while in the hidden and output layers, each node corresponds to a neuron. The calculation process are as follows:

Step 1: Design the essential parameters of the hybrid algorithm, including the maximum number of iterations M, the minimum error ε, the number of individuals in the population N, c_1, c_2, etc.

Step 2: Determine the fitness value based on the fitness function.

Step 3: Sort the fitness values and keep the top $N/2$ particles.

Step 4: If the total number of retained particles is equal to N, proceed to the next step; or else, go back to Step 2.

Step 5: Update the particle information based on equations (1) and (2).

Step 6: If the error is less than ε or the number of iterations reaches M, output the global extremum $Gbest$ and particle information; or else, go back to Step 3.

Step 7: Output the granule information to determine the initial weights and thresholds.

Step 8: Calculate the error value.

Step 9: If the minimum error is met, proceed to the next step; or else, go back to Step 7.

Step 10: End the training.

The optimization algorithm flow of the GPSO-BP is shown in Figure 2.

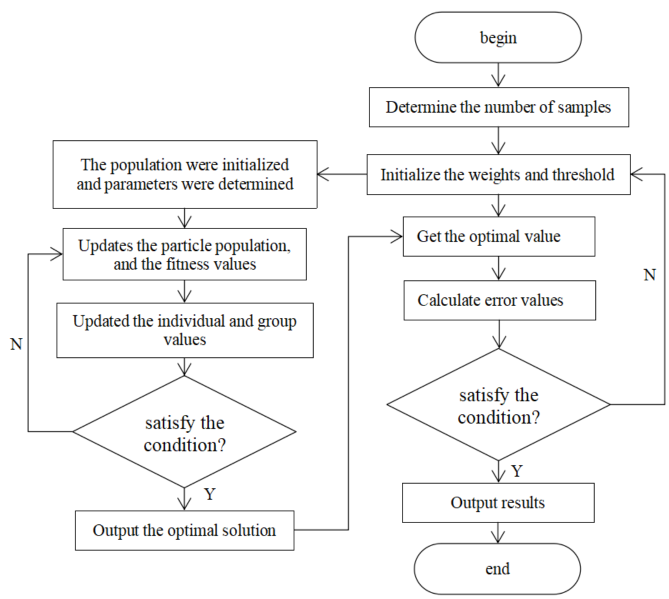

Figure 2. Optimization algorithm flow of GSO-BP

4. EXPERIMENTAL ANALYSIS

The data used in this study was gained from the historical electricity load from a city, comprising the maximum daily load from June 14th to June 19th, 2023. The recommended neural network model was utilized to predict the power load for the 24 time periods on June 19th. After the training is completed, the PSO-BP model is tested using the training samples, and the measurement is a check of the differences between all output prediction loads and actual power loads.

4.1 Parameter selection

The main arguments settings for this case are as follows: the group size is 20, the inertia weight factor varies rectilinear from wmax = 1.2 to, and c1=c2=2, the maximum value of iterations is set to 1000. The maximum daily load from June 14th to June 17th is used as the input data for the exercising samples, and the load on June 18th is used as the output data for the training samples; the maximum daily load from June 12th to June 18th is used as the input data for the test samples, and the load on June 19th is used as the output data for the test samples. The training times of the BP are set to 1000, and the structure of the BP is 11-1-1, with 11 nodes in the input layer, hidden layer and output layer are all 1. All samples are normalized using Equation (3). The sample data is trained and predicted using the BP-NN, PSO-BP-NN, and GPSO-BP-NN load prediction models, respectively.

4.2 Experimental result

The fitness function value change process for power load prediction using the proposed method is shown in Figure 3.

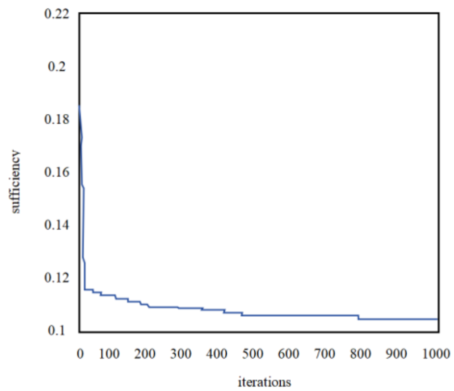

Figure 3. Change of fitness function value.

According to the fitness function value change results in Figure 3, the GPSO-BP-NN proposed in this study obtains fitness function values higher than 0.18. In this stage, the value decreases rapidly until the iteration reaches 20. Although the fitness function value still shows a decreasing trend, the rate of decrease slows down noticeably until the iteration reaches 480. The adapted function value gradually stabilizes and tends to be below 0.1. As the adapted function value piecemeal reduces, it indicates that the predicted values are similar to the actual values. This demonstrates that the proposed GPSO-BP-NN method for short-term power load prediction is in accordance with this requirement.

From Table 1, it can be drawn from the GPSO-BP-NN model has higher prediction accuracy than the PSO-BP-NN and BP-NN models. The average absolute error of the GPSO-BP-NN model is 2.21%, while the average absolute errors of the PSO-BP-NN and BP-NN models are 2.39% and 3.53%, respectively.

Table 1. Prediction results of PSO-BP model and BP model.

Time	Relative value	GPSO-BP-NN		PSO-BP-NN		BP-NN	
		Forecast load/MW	Relative error/%	Forecast load/MW	Relative error/%	Forecast load/MW	Relative error/%
0:00	625.2419	624.3501	-0.12672	617.4682	1.684287	627.2808	1.8018
1:00	605.259	619.0672	2.333876	615.3411	-0.13572	620.9755	4.1559
2:00	595.322	616.2902	3.594333	610.0582	2.324876	616.974	5.2385
3:00	592.5026	615.3676	3.936692	607.2812	3.585333	613.3444	5.125
4:00	600.2531	616.1787	2.711613	606.3586	3.927692	611.6205	3.4554
5:00	613.1934	618.6766	0.925543	607.1697	2.702613	616.3394	2.0208

Time	Relative value	GPSO-BP-NN		PSO-BP-NN		BP-NN	
		Forecast load/MW	**Relative error/%**	**Forecast load/MW**	**Relative error/%**	**Forecast load/MW**	**Relative error/%**
6:00	655.1069	643.8039	-1.73145	609.6676	0.916543	630.9072	-2.3422
7:00	740.2196	728.271	-1.6161	634.7949	-1.74045	665.8921	-8.924
8:00	753.9465	714.3182	-5.30176	719.262	-1.6251	703.3772	-5.5701
9:00	811.5071	801.5475	-1.22309	705.3092	-5.31076	798.0059	-0.5508
10:00	824.618	834.2122	1.194337	792.5385	-1.23209	788.2253	-3.3485
11 :00	833.3377	848.4457	1.850785	825.2032	1.185337	800.5829	-2.8717
12:00	768.6996	768.6693	0.01402	839.4367	1.841785	737.5659	-2.9034
13:00	728.2479	709.9882	-2.52079	759.6603	0.00502	711.4781	-1.07
14:00	798.5049	810.2254	1.502574	700.9792	-2.52979	793.0869	0.4638
15:00	822.1891	827.8802	0.717868	801.2164	1.493574	808.4227	-0.576
16:00	812.7684	833.7944	2.633984	818.8712	0.708868	806.165	0.3083
17:00	869.3486	876.7122	0.873899	824.7854	2.624984	873.1146	1.4939
18:00	974.0697	898.3209	-7.8312	867.7032	0.864899	866.2595	10.2469
19:00	931.7149	896.4888	-3.79973	889.3119	-7.8402	863.0005	-6.4617
20:00	925.3952	895.5388	-3.24009	887.4798	-3.80873	864.4546	-5.6581
21:00	875.8925	887.9343	1.407101	886.5298	-3.24909	848.8419	-2.0722
22:00	784.1658	818.0268	4.386318	878.9253	1.398101	749.6089	-3.2869
23:00	682.8979	669.4692	-1.97473	809.0178	4.377318	642.272	-4.6828
Mean absolute error/%		2.21		2.39		3.53	
Maximum relative error/%		7.8492		8.2946		10.2379	

From the experimental results in Figure 4, it can be drawn from the power load forecast results of the proposed method are very close to the actual power load output results. The actual output results are highly consistent with the predicted output curve, demonstrating the high accuracy of the proposed method for power load prediction.

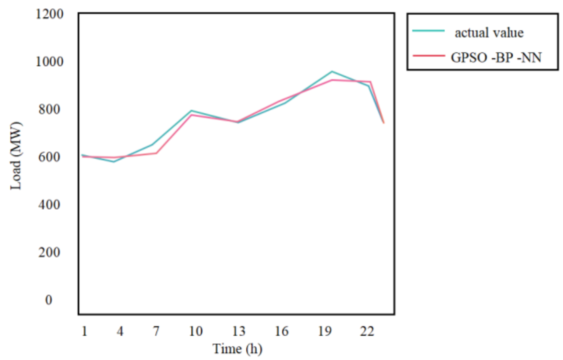

Figure 4. The prediction result of GPSO-BP-NN

To further compare the effectiveness of the GPSO-BP-NN, PSO-BP-NN, and BP-NN neural networks in power load prediction, the power load prediction for the 24 time periods on June 19th, 2023, by using the proposed method is compared with the random distributed method and the chaotic optimization method. The results are exhibited in Figure 5. From the experimental results in Figure 5, it can be found that the GPSO-BP-NN method for power load prediction achieves higher accuracy compared to the PSO-BP-NN and BP-NN neural network prediction methods. Specifically, when different time points are selected, it is found that the GPSO-BP-NN has better prediction results at all time points except 1 o'clock and 4 o'clock. This also indicates the high effectiveness of the proposed method for power load prediction.

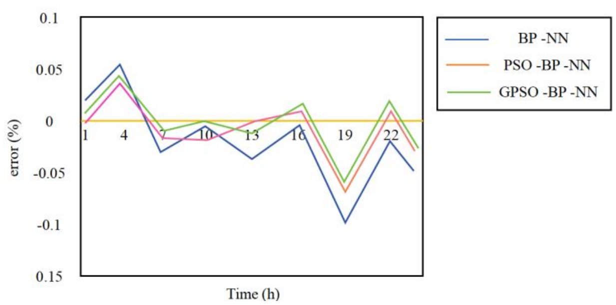

Figure 5. The prediction result of model.

To further compare the effectiveness of the GPSO-BP-NN, PSO-BP-NN, and BP-NN neural networks in power load prediction, the time consumption is analyzed based on the above prediction results. The selected time period is from June 14th to June 19th, 2023. The comparison results of the three algorithms are shown in Figure 6.

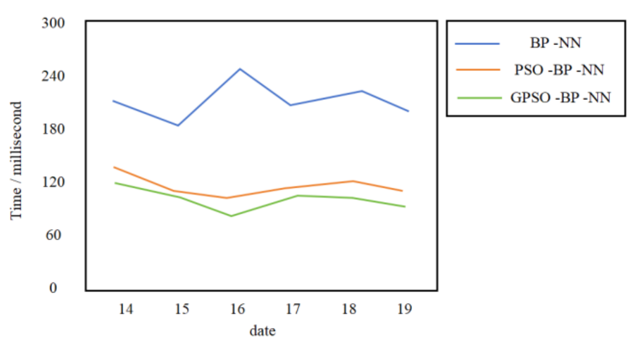

Figure 6. The prediction result of model.

From the comparison results of the time consumption in Figure 6, it can be concluded that the time consumption of the GPSO-BP-NN model for power load prediction is below 120ms, indicating the real-time performance of power load prediction, which can improve the management performance of power companies. The experimental results effectively verify the high real-time performance of the proposed method for power load prediction. Combining the training results of fitness function value change, prediction accuracy, relative error, and time consumption, it can be found that the proposed GPSO-BP-NN neural network structure can accurately predict power load.

5. CONCLUSION

Accurate prediction of power load is important for the sustainable development of power companies and has significant implications for national economic development. In this study, a short-run power load forecast method based on ameliorated PSO-BP is proposed, applying granule swarm optimisation algorithm to optimize the scales and sills of the neural network. Due to the fast convergence speed of the particle swarm algorithm, the proposed GPSO-BP-NN improves the optimization impact of the simple PSO-BP. The results of fitness function value change show that the GPSO-BP-NN algorithm is close to the actual values. Among the prediction results of GPSO-BP-NN, PSO-BP-NN, and BP-NN models, the average absolute error of the GPSO-BP-NN model is 2.21%, which is lower than the other two short-term power load prediction models. Finally, through the analysis of prediction accuracy, algorithm comparison, and time consumption, the GPSO-BP-NN algorithm is superior to the other two algorithms, reasoning the effectiveness of the recommend improved algorithm.

REFERENCES

[1] Jin, Y., Acquah, M. A., Seo, M., et al., "Short-term electric load prediction using transfer learning with interval estimate adjustment," Energy and buildings, 258 (2022).

[2] Mountassir, F. and Bousmah, M., "Short term electricity load forecasting using smart Grid AI-prediction simulator (SGAIPS)," Foundation of Computer Science (FCS), NY, USA, 22 (2021). DOI:10.5120/IJCA2021921589.

[3] Nassif, A. B., Soudan, B., Azzeh, M., et al., "Artificial intelligence and statistical techniques in short-term load forecasting: a review," (2021), DOI:10.48550/arXiv.2201.00437.

[4] Ng, R. W., Begam, K. M., Rajkumar, R. K., et al., "An improved self-organizing incremental neural network model for short-term time-series load prediction," Applied Energy, 292 (2021). DOI: 10.1016/j.apenergy.2021.116912.

[5] Memarzadeh, G. and Keynia, F., "Short-term electricity load and price forecasting by a new optimal LSTM-NN based prediction algorithm," Electric Power Systems Research, 192(4 October), 106995 (2021). DOI: 10.1016/j.epsr.2020.106995.

[6] Subbiah, S. S. and Chinnappan, J., "Deep learning based short term load forecasting with hybrid feature selection," Electric Power Systems Research, (2022).

[7] Yi, L., Zhu, J., Wang, Y., et al., "Short-term power load forecasting based on orthogonal PCA-LPP dimension reduction and IGWO-BiLSTM," Recent Patents on Mechanical Engineering, (2023).

[8] Ozer, I., Efe, S. B. and Ozbay, H., "A combined deep learning application for short term load forecasting," Alexandria Engineering Journal, 60(4), 3807-3818 (2021). DOI: 10.1016/j.aej.2021.02.050.

[9] Mamlook, R., Badran, O. and Abdulhadi, E., "A fuzzy inference model for short-term load forecasting," Energy Policy, 37, 1239-1248 (2009).

[10] Amjady, N. and Keynia, F., "Short-term load forecasting of power systems by combination of wavelet transform and neuro-evolutionary algorithm," Energy, 34, 46-57 (2009).

[11] Casolari, S. and Colajanni, M., "Short-term prediction models for server management in Internet-based context," Decision Support Systems, 48, 212-223 (2009).

[12] Taylor, J. W., "Triple seasonal methods for short-term electricity demand forecasting," European Joumal of Operational Research, 204, 139-152 (2010).