Short-term load forecasting of BP network based on bacterial foraging optimization

Yikan Han¹, Hejin Xiong¹, Di Wei²*

¹College of Automation, Wuhan University of Technology, Wuhan, Hubei, 430070, China
²State Grid Xiaogan Electric Power Supply Company, Xiaogan, Hubei, 432000, China
³Corresponding author’s e-mail: 380877327@qq.com

Abstract. The traditional BP neural network search algorithm based on gradient orientation is easy to fall into local minimum, the convergence process is slow, and it cannot guarantee the convergence to the global optimal solution. In order to improve the prediction accuracy of short-term load forecasting, this paper applies the bacterial foraging optimization algorithm to BP neural network, using its unique breeding and eviction operation, can improve the convergence speed of the algorithm, strengthen the ability to search the global optimal solution, and reduce the prediction error.

1. Introduction
By simulating the structure, function and signal sensing mode of the human brain, the neural network algorithm has the advantages of parallel distributed, nonlinear processing, and self-learning[1-2]. Because of its simple structure, small computation, and high feasibility, it has been widely studied and applied in short-term power load forecasting. However, the traditional BP neural network is a gradient-oriented search algorithm, the convergence process is slow, and it is easy to fall into the local minimum, cannot guarantee convergence to the global optimal solution[3].

For intelligent optimization algorithms, experts and scholars at home and abroad have put forward many ideas around them, for example, genetic algorithm (GA), evolutionary programming algorithm (EP), evolutionary strategy algorithm (ES), etc. which are based on evolutionary and genetic theory and the particle swarm (PSO), ant colony optimization algorithm (ACO), etc., which are inspired by the biological behavior of nature[4]. Under this background, Ohio State University scholar Kevin M. Passino proposed Bacterial Foraging Optimization (BFO), an intelligent algorithm for bionic optimization by simulating group foraging behavior of E. coli flora[5]. The unique breeding and eviction operation of the bacterial foraging algorithm can improve the convergence speed of the algorithm and enhance the ability to search for the global optimal solution. In this paper, the BP neural network was optimized by the bacterial foraging optimization algorithm. An improved BP network load forecasting model based on the bacterial foraging optimization algorithm was established and applied to the short-term load forecasting in a certain area.

2. Bacterial foraging optimization algorithm
E. coli is one of the most common microorganisms in nature. When it exists inside a living organism, it always follows the rules of obtaining the most energy per unit time to search for food, and information exchange between bacteria and bacteria is also carried out. In each step of the move, the
current state is also promptly fed back and judged, and then the most efficient foraging path is
developed.

The BFO algorithm truly mimics the four main behavior of *E. coli* foraging: Chemotaxis, Swarming, Reproduction, Elimination and Dispersal[6]:

The main implementation steps of the bacterial foraging optimization algorithm are as follows[7]:

1. Initialize each parameter. \( N_c \) is the number of chemotaxis behavior, \( N_{re} \) is the number of breeding behavior, \( N_{ed} \) is the number of expulsion behavior, \( P_{ed} \) is expulsion probability; \( S \) is the size of the colonies (generally even), \( N_j \) is the maximum step size of the chemotaxis operation, \( C(i) \) is forward The step size of swimming.

2. Perform chemotaxis, reproduction, and expulsion cycles, \( j = j + 1, k = k + 1, l = l + 1 \). The spatial position vector of the bacteria is represented by \( \mathbf{P}_{ijkl} \).

3. The bacteria are initialized, and the initial position is generated by the formula (1), set the fitness \( J \).

\[
X = x_{\min} + (x_{\max} - x_{\min}) \text{rand}
\]

\( \text{rand} \) is a random number uniformly distributed in the range of [0, 1].

4. Simulated bacterial chemotaxis
   a) Flip: Operate according to equation (1-2)
   \[
   P(i, j+1, k, l) = P(i, j, k, l) + C(i)\phi(i)
   \]

   \[
   \phi(i) = \Delta(i)\left(\Delta^T(i)\Delta(i)\right)^{1/2}
   \]

   b) Swimming: If the \( J \) of the bacteria improves after the flipping operation, the movement is performed in the direction obtained by (a), until the \( J \) no longer changes, or the number of steps moved reaches \( N_j \).

5. Carry out the bacterial reproduction cycle, after completing a chemotactic behavior, find the \( J \) of each bacteria and sum up the bacterial energy, and sort out the 1/2 bacteria with less energy in order of the largest to the smallest. The breeding operation is completed for the remaining bacteria.

6. When the operation of step (5) is completed, the generated random probability is compared with \( P_{ed} \). If it is less than \( P_{ed} \) and \( l < N_{ed} \), the eviction operation is performed, and the step (2) is returned, otherwise the optimization is ended.

3. BP neural network model based on bacterial foraging optimization

3.1. Establish model

Firstly, the network structure of the neural network is established, and the connection weights and thresholds are encoded and used to represent a single individual, and the relationship between the expression and the network dimension is expressed. If the neural network contains \( M \) connection weights and thresholds, an \( M \)-dimensional matrix containing \( M \) weights and thresholds is used to represent the individual. The weights and thresholds in the neural network can be expressed by the elements contained in the bacterial body \( \mathbf{B}(i) \).

For example, if a BP neural network contains six connection weights and thresholds, that is, \( \{w_{31}, w_{32}, w_{41}, w_{42}, w_{53}, w_{54}\} \), let \( x_1 = w_{31}, x_2 = w_{32}, x_3 = w_{41}, x_4 = w_{42}, x_5 = w_{53}, x_6 = w_{54} \), so that a single individual can be expressed by equation (4):

\[
\mathbf{B}(i) = \{x_1, x_2, x_3, x_4, x_5, x_6\}
\]

Then the weights and thresholds in the neural network can be expressed by the elements contained in the bacterial body \( \mathbf{B}(i) \).
For a bacterial population, it can be expressed by $BM$ as shown in formula (5), $s$ is the number of bacteria contained in the flora:

$$BM = [B(1), B(2), \cdots, B(s)]$$ (5)

### 3.2. Bacterial initialization

According to the specific composition of the network, the generation of bacterial populations is carried out according to the formulas (4) and (5). Different bacterial individuals in the population represent different weight and threshold combinations in the BP network.

### 3.3. BP neural network training and evaluation of bacterial quality

Bacterial individuals in the colony population represent weights and thresholds in the neural network to form different BP networks. The training samples are substituted into the new network for training, and the iteration is performed according to the evolution of the BP neural network algorithm, and then the mean square error MSE of the BP neural network is set as the objective function.

### 3.4. Iterative cycle for bacterial foraging optimization

1. Training the BP network structure represented by each bacterial individual;
2. The mean square error MSE obtained by the training is taken as the fitness $J$ of the network;
3. Reproduction operation: The fitness calculated by all individuals in the colony is sorted from large to small, and the latter half is eliminated. The remaining individuals are subjected to step propagation;
4. Expulsion operation: Select a part of the individual to reinitialize with probability $P_{ed}$;
5. Iterative looping, after a certain number of repeated training, breeding, and expulsion operations, screens out the optimal fitness and saves the corresponding individual vector, namely the weight and threshold of the BP network.

### 3.5. Termination of the algorithm

The algorithm is terminated when the optimal fitness in the population satisfies the design requirements of the BP network or reaches the highest number of iterations. The optimal solution of the algorithm is decoded into the BP network, and the BP neural network model composed of the optimal weights is used for prediction.

The specific process of BP neural network optimization using BFO algorithm is shown in Figure 1.
Figure 1. Flow chart of BFO algorithm for BP neural network optimization

4. Study simulation

The short-term electric load in a certain area of Hubei province was predicted by the above BFO-BP neural network model. The sample selected in this part of the paper is the historical load data and historical meteorological data of a total of 30 days from July 28 to August 26, 2015 in Hubei province as the training samples, and the data in the training sample is preprocessed to eliminate the bad data.

A three-layer BP network neural network model is established, which adopts a single-output network structure form. The input variable is the load value $L(t-1), L(t-2)$ at two time points before the predicted time, the load value $L(t-T)$ at the same time on the previous day, and load value $L(t-7T)$ at the same time on the same day last week. The daily characteristic meteorological factors include the highest temperature $T_{\text{max}}$, the lowest temperature $T_{\text{min}}$ and the type of rainfall $R$, date type $D$. The number of hidden layer nodes is 5, S-type function is used as activation function, and the output is the load value at the time to be predicted.
At the same time, the BFO-BP neural network model is compared with the standard BP training prediction results. The predicted 96-point load data on August 27, 2015 is compared with the actual load. The comparison of the predicted curves is shown in Figure 2, and the comparison of the prediction error (MAPE) curves is shown in Figure 3.

Similarly, the load on the rest day is also predicted using the EMD-BP model and the standard BP model as a comparison. The prediction results are shown in Figure 4, and the prediction error (MAPE) curve is shown in Figure 5.

The error indexes of the two models are shown in Table 1. The BFO-BP short-term load forecasting model in this paper has less error than the standard BP model in forecasting both workday and rest day, meeting the requirements of practical application of short-term load of power system. It can be seen that the prediction model of this paper is feasible for short-term power load forecasting.

| Table 1. Comparative analysis of prediction results |
|----------------|----------------|----------------|----------------|
| Working day | BP          | BFO-BP       |
| Maximum absolute error (%) | 19.516 | 8.692 |
| Minimum absolute error (%) | 0.054 | 0.021 |
| Average absolute error (%) | 5.460 | 2.414 |
| Rest day | BP          | BFO-BP       |
| Maximum absolute error (%) | 9.789 | 7.528 |
| Minimum absolute error (%) | 0.088 | 0.061 |
| Average absolute error (%) | 4.581 | 2.633 |
5. Conclusion
In this paper, an improved BP network load forecasting model based on bacterial foraging optimization algorithm is established and applied to short-term electric load forecasting in certain area. The prediction results are compared with the standard BP algorithm. The simulation results show that the BP neural network is optimized by the bacterial foraging optimization algorithm, which has higher prediction accuracy and is valuable for further study in the field of short-term power load forecasting.

References
[1] Zhang H T, Xu F Y, et al. (2010) Artificial neural network for load forecasting in smart grid Machine Learning and Cybernetics (ICMLC). In: 2010 International Conference on Machine Learning and Cybernetics. Qingdao. pp. 3200-3205.
[2] Liu Q Dang C, et al. (2013) A one-layer recurrent neural network for real-time portfolio optimization with probability criterion. IEEE transactions on cybernetics., 43 (1): 14-23.
[3] Khwaja A S, Naeem M, et al. (2015) Improved short-term load forecasting using bagged neural networks. Electric Power Systems Research., 125: 109-115.
[4] Al-Hadi I A A, Hashim S Z M, et al. (2011) Bacterial Foraging Optimization Algorithm for neural network learning enhancement. In: 11th International Conference on Hybrid Intelligent Systems (HIS). Melacca. pp. 200-205.
[5] Li Yanan. (2009) Research on bacterial flora optimization algorithm. Harbin Institute of Technology (in Chinese).
[6] Yan Xueting. (2013) Research on multi-objective resource-constrained project scheduling problem based on bacterial foraging optimization algorithm. South China University of Technology (in Chinese).
[7] Ajith Abraham, Aboul-Ella Hassanien, Patrick Siarry, et al. (2009) Foundations of Computational Intelligence Volume 3. Springer-Verlag Berlin and Heidelberg GmbH & Co. K, Springer Berlin Heidelberg.
[8] Lin Yangyu, Yi Wei, et al. (2014) Short-term load forecasting considering meteorological factors and date types. Computer Simulation., 3: 109-112 (in Chinese).
[9] Hong T. (2010) Short term electric load forecasting. North Carolina State University.
[10] Kantardzic M, Gavranovic H, et al. (2015) Improved Short Term Energy Load Forecasting Using Web-Based Social Networks. Social Networking., 4: 119.
[11] Cheng Tao. (2006) Somatosensory Temperature Rolling Forecast System and Its Application in Short-term Load Forecasting of Hunan Power Grid. Hunan Electric Power., 26: 1-4 (in Chinese).
[12] Li Xiaoyan. (2013) Short-term load forecasting of power system considering meteorological factors. South China University of Technology (in Chinese).
[13] Pierrot A, Goude Y. (2011) Short-term electricity load forecasting with generalized additive models. In: 16th Intelligent System Applications to Power Systems Conference (ISAP). Greece. pp. 593-600.