The 2nd International Conference on Complexity Science & Information Engineering

Short-term power load forecasting based on IVL-BP neural network technology

Yongli Wang*, Dongxiao Niu, Li Ji

North China Electric Power University, 2 Beinong Road, Changping District, Beijing 102206, China

Abstract

Accurate forecasting of power load has been one of the important issues in the electricity industry. Recently, along with the privatization and the deregulation, accurate forecasting of power load draws more and more attentions. There are many difficulties in the application of BP neural network which is a very useful tool for the forecasting, such as the defining for the network structure and the local solution which is easy to fall into. To solve these problems, the back-propagation (BP) neural network short-term load forecasting method based on improved variable learning rate back propagation (IVL-BP) is presented in this paper. Though introducing two threshold parameters for the amount of the mean square increasing and decreasing, the learning algorithm is sensitive to the error and convergence speed. Then use genetic algorithm to train network parameters until the error tending to some stable value. Then conduct BP algorithm with the optimized weights to achieve short-term load forecasting. The experimental results have shown that the load forecasting system based on this method has higher accuracy and real-time.

© 2011 Published by Elsevier Ltd. Selection and peer-review under responsibility of Desheng Dash Wu.

Keywords: IVL-BP; neural network; load forecasting; genetic algorithm

1. Introduction

Power load prediction has attracted a great deal of attention from both the practice and academia. The short-term power load forecasting is very significant for the electric network’s reliability and economic development. As short-term power load prediction is of crucial importance to the reliability and economic utilization of electric networks, it is drawing more and more attention from both the practice and academia. The aim of load forecasting is to make the best use of electric energy and relieve the conflict between supply and demand. Inaccurate forecast of power load will leads to a great deal of loss for power companies. Bunn and Farmer pointed out that a 1% increase in forecasting error implied a 10 million increase in operating costs. The short-term forecasts refer to hourly prediction of electricity load demand for a lead time ranging from 1h to several days ahead. In certain instances the prediction of the daily peak load is the objective of short-term load forecasting, since it is the most important load during any given day. The quality of short-term hourly load forecasts has a significant impact on the economic operation of the electric utility since many decisions based on these forecasts have significant economic consequences.

* Corresponding author. Tel.: +86-10-51963870
E-mail address: wyl_2001_ren@126.com
For many years, a lot of researches have been done by domestic and foreign scholars on the theories and methods for the short-term load forecasting. Many methods have been represented such as the chaotic time series, artificial neural network, grey system, small wave analysis and so on. In these years, the BP network is the main neural network used in power system load forecasting. The learning algorithm of basic BP network is based on the error gradient back propagation, the learning convergence is slow, it is easy to fall into local minimum and can not get the global optimal solution, besides it’s sensitive to the selection of initial weights. To solve these problems, many researchers have proposed lots of improved algorithm in recent years, such as modifying the transfer function appropriately in the training process, changing the parameters, improving the accuracy of echelon estimation, improving the momentum coefficient and so on. In this paper, the improved variable learning rate back propagation has been proposed, and it has been improved that the new model is more prominent in the small-sample learning and forecasting.

2. IVL-BP in short-term load forecasting

2.1. The pretreatment of load data

Because of the measuring device itself or a variety of reasons in the data transmission, even the power cuts caused by human beings, some non-real data are contained in the historical load data. If those pseudo-data are used in the load model, the accuracy of forecasting would be influenced greatly, and even the prediction curve would be changed. Therefore, before using these data, it’s necessary to pre-process them and remove the interference factors for historical data.

Assume the load sequence is \((i, n)\), \(i = 0, 1, 2 \cdots, 23\) represents 24 points a day, and \(n = 0, 1, 2 \cdots, N\) is the date series. Then the average load is \(E(i)\) and the variance is \(V(i)\) of the load curve for \(N\) days:

\[
E(i) = \frac{1}{N} \sum_{k=1}^{N} x(i, k) \tag{1}
\]

\[
V(i) = \sigma_i^2 = \frac{1}{N} \sum_{k=1}^{N} [x(i, k) - E(i)]^2 \tag{2}
\]

\[
\rho(i, n) \text{ delimited as the bias ratio:}
\]

\[
\rho(i, n) = \frac{|x(i, n) - E(i)|}{\sigma_i} \tag{3}
\]

If \(\rho(i, n) \geq 1.1\), the load point would be abnormal and need to be removed; If \(\rho(i, n) < 1.1\), it would be the normal one. The abnormal data need to be replaced with \(x(i, n)\) as the following formula:

\[
\hat{x}(i, n) = \frac{x(i, n-1) + x(i, n+1)}{2} \tag{4}
\]

Through the pretreatment of the raw data above, the original series of load have become more reasonable.
2.2. The input data of the network

2.2.1 Factors impacting load

Power system load is a very complex nonlinear system, and there are many factors that impact on the load curve directly or indirectly. Generally, the load can be divided into two categories as periodic load and fluctuant load.

The fluctuant load is influenced by these random factors. For a long load curve, its mean value is zero, but the mean value would be changed greatly in a short period.

The main influential factors would be temperature, holidays or weather conditions. The normalized value of “the characteristics of day” could be considered as a neural network input node to calculate in this paper. The daily maximum temperature and minimum temperature are used as two input of the neural network. Besides, the weather condition which has been fuzzy discreted can be set as another input.

2.2.2 Selection of the network input nodes and the network model

The main influential factors would be temperature, holidays or weather conditions. The normalized value of “the characteristics of day” could be considered as a neural network input node to calculate in this paper. The daily maximum temperature and minimum temperature are used as two input of the neural network. Besides, the weather condition which has been fuzzy discreted can be set as another input.

The network input matrix is as following:

\[
\begin{bmatrix}
    a_1(t-1,h) & a_1(t-2,h) & \ldots & a_1(t-7,h) & T_{1H}(t) & T_{1L}(t) & m(t) & d_{1w}(t) \\
    a_2(t-1,h) & a_2(t-2,h) & \ldots & a_2(t-7,h) & T_{2H}(t) & T_{2L}(t) & m(t) & d_{2w}(t) \\
    \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\
    a_p(t-1,h) & a_p(t-2,h) & \ldots & a_p(t-7,h) & T_{pH}(t) & T_{pL}(t) & m_p(t) & d_{pw}(t)
\end{bmatrix}
\]

While the output matrix of the network is:

\[
\begin{bmatrix}
y_1(t) \\
y_2(t) \\
\vdots \\
y_p(t)
\end{bmatrix}
\]

Where \(a(t-k,h)\) is the load at \(h\) point on the \(k\) days before the predicted \(t_{th}\) day; \(y(t,h)\) represents the actual load at \(h\) point on the predicted \(t_{th}\) day; \(T_{h}(t)\) and \(T_{l}(t)\) are the highest and lowest temperature on the predicted \(t_{th}\) day respectively; \(m(t)\) is as the climate-sensitive factor; and \(d_{w}(t)\) is the coefficient of the characterized day.

The layer of BP network is \(m\), the input of the network is proposed to be R-dimensional vector \(P\), the number of neurons in the \(m^{th}\) layer is \(I^m\), \(W^m = W_{I^m \times I^{m-1}}\) is the weight matrix of \(m\) layer \(n^m\) and \(a^m\) represent the net input and output of \(m\) layers, \(b^m\) and \(f^m\) denote the threshold vector function and the activation vector function of layer \(m\). the output of some layer is the input of the next level, that is:

\[
a^{m+1} = f^{m+1}(W^{m+1}a^m + b^m), \quad m = 0, 1, 2, \ldots, M
\]

The first layer neuron receives the input from the outside, \(a^0 = P\); and the output of the last layer neuron is the output of the whole network, \(a = a^M\). The parameters of the network is adjusted by the algorithm until gets the minimum mean square error, which is calculated by formula(6):

\[
E_{min} = \min(E((t-a)^T(t-a)))
\]

The weight update iterative formula can be defined as:
\[ W^m(k + 1) = W^m(k) + a^m \langle a^{m-1} \rangle^T \]  
(7)

\[ b^m(k + 1) = b^m(k) + a \quad \text{s} \]  
(8)

Where, \( a \) is the learning rate, and \( 0 < a < 1 \).

\[ s^M = -2F^M(n^M)(t - a) \]  
(9)

\[ s^m = F^m(n^m)(W^m_{m+1})^T s^{m+1} \]  
(10)

Where, \( F^m \) is the sensitivity of the activation vector function to the net input \( n^m \) of layer \( m \), which is expressed by the following formula:

\[ F^m = \frac{\partial F^m(n^m)}{\partial n^m} \]  
(11)

The learning rate in normal BP algorithm is fix, but in practice, if the learning rate is too small, the convergence is guaranteed, while learning is too slow; but if it’s too great, it can learn fast, which may cause more oscillations or divergence. So the proper adjustment of the learning rate during the training process is necessary.

The basic idea of variable learning rate back propagation is: in the training process, if the mean square error increases after the update of the weight, and it would be more than the preset value, the learning rate decreases; if less than the preset value, the learning rate would be unchanged; while if the mean square error decreases after the update, the learning rate increases.

Although compared with the normal BP algorithm, the convergence speed of IVL-BP algorithm has become faster, the changes of error is complex and highlight, it’s still difficult for IVL-BP to adapt to this situation, and the convergence is relatively slow. To solve this, two threshold parameters \( \zeta^1 \) and \( \zeta^2 \) are set to measure the increase and decrease of the mean square error. The mean square error converges faster and more smooth along the gradient direction, and easier to reach the global minimum point. The algorithm is as follows:

If the mean square increases after the update weight, then:

\[ W^m(k + 1) = W^m(k), \Delta E > \zeta \]  
(12)

\[ a = a \times \rho, 0 < \rho < 1 \]  
\{ , \Delta E > \zeta \}

\[ W^m(k + 1) = W^m(k) + a^m \langle a^{m-1} \rangle^T \]  
\{, 0 < \Delta E < \zeta \}  
(13)

If the mean square decreases after the update weight, then:

\[ W^m(k + 1) = W^m(k) + a^m \langle a^{m-1} \rangle^T \]  
\{, \Delta E \geq \zeta \}  
(14)
\[ W^m(k+1) = W^m(k) + a^m (a^{m-1})^T, \Delta E < \zeta^2 \] (15)

3. GA for the training of the neural network weight

Because the convergence of BP network training is slow, the genetic algorithm (GA) was proposed to optimize the network parameters in this paper, as a result, it accelerates the convergence speed and avoid the local minimum problem due to the advantage of global optimization of GA.

The so-called network parameters optimization is to find a good network weights and thresholds, to minimize the global error in network. The parameters of BP network was optimized by the modified genetic algorithm, and the steps are as follows:[3]

1) Encode parameters. Take the weights and threshold value of the BP network as genes. In order to shorten the length of the code, the real number coding method was used in stead of binary code. And each weight value and threshold value is expressed by a single real number. As a result, the restructuring only occurs with some weight value or threshold value, which is favorable for evolution.

2) Generate initial population. Since the good network weight is relatively small, in (-1,1), so the range of each gene is (-1,1). The initial population is determined randomly by the probability distribution \[ e^{-\delta} \], in which the range of feasible solutions can be searched with the genetic algorithm.

3) Assess the fitness. The fitness in this paper is calculated though fitness=1/E. it’s obviously that the E is smaller the ability of adaptability would become stronger.

4) The selection is calculated through Roulette Wheel method.

5) The crossover and mutation operations. The size of crossover rate \( P_c \) and mutation rate \( P_m \) which has great influence to the operation performance of genetic. In fact, lower \( P_c \) and \( P_m \) should be taken for the solution with higher fitness, which increase the opportunity to the next generation for the individual. Otherwise, if the fitness is low, then higher \( P_c \) and \( P_m \) should be taken to eliminate the individual. In this paper, the operation of the adaptive crossover rate and mutation rate is as follows:

\[ P_c = \phi / (f_{\max} - \bar{f}), \quad P_m = \varphi / (f_{\max} - \bar{f}) \] (16)

Where, \( \phi \) and \( \varphi \) are the constants less than 1.0, \( f_{\max} \) and \( \bar{f} \) are the largest and average value of fitness respectively.

6) Calculate the fitness of new individuals, and determine whether it reaches the condition for the end, if it is, go to step7, otherwise, go back to step4 and continue iteration.

7) The best individual gene is the optimal solution of the network weights and thresholds, then use them to complete the load forecasting in neural network.

4. Simulation

Use the IVL-BP hybrid model above to predict the load of a regional power grid. Take the June 19, 2009 as forecast date, forecast week is from June 19 to 30, 2009, and the load from June 5 to 16, 2009, as the historical samples. The selected parameters are as follows: the length of code is 48, the number of individuals is 20, the number of evolution generations is 300, \( \zeta^1 = 0.04 \in [0.01, 0.05], \zeta^2 = 0.5 \in [0.5, 0.8], \eta = 1.05, \rho = 0.7 \), learning step in BP
network $n=0.01$. Using MATLAB programming language, the results are shown in Figure 1 and 2 below. Figure 1 shows the daily load curve of April 17, and line 1 and line 2 represent the actual load and forecast load.

Figure 1 The daily load curve of April 17, 2009

It can be seen from Figure 1 that forecasting load curve and actual load curve is very close, and the maximum relative error is 3.24%, the average error is 0.91%, which fully meets the requirements of load forecasting. Besides, the network learning speed has been improved greatly, the statistical average time of learn is 5000, while that of the BP algorithm is 17000. Figure 2 shows the best individual is found in the 200 generations during the evolution, which indicates the evolution is very rapid, and the corresponding average fitness is 1.8135.

Figure 2 The evolution process of GA
5. Conclusion

Short-term load forecasting plays an important role in power system. BP algorithm has been widely used in a number of previous prediction methods. But the BP algorithm has slow convergence, and easily falls into local minimum limitations. Though introducing two threshold parameters for the amount of the mean square increasing and decreasing, the learning algorithm is sensitive to the error and convergence speed. Then it proposed a hybrid algorithm to optimize the network weights though the improved GA algorithm, and apply it to real systems. The experimental results show that this method not only improves learning speed, but also improve the learning accuracy, which has great helpful for the realization of power distribution planning in the power system.

6. Acknowledgements

This research has been supported by “the Fundamental Research Funds for the Central Universities”, It also has been supported by Natural Science Foundation of China (71101048).

7. References


