Short, medium and long term load forecasting model and virtual load forecaster based on radial basis function neural networks

Changhao Xia\textsuperscript{a,b,*}, Jian Wang\textsuperscript{b,*}, Karen McMenemey\textsuperscript{c}

\textsuperscript{a}College of Electrical Engineering and Information Technology, China Three Gorges University, Yichang Hubei 443002, China
\textsuperscript{b}School of Mechanical and Aerospace Engineering, Queen’s University, Belfast, Northern Ireland BT9 5AH, UK
\textsuperscript{c}School of Electronics, Electrical Engineering and Computer Science, Queen’s University, Belfast, Northern Ireland BT9 5AH, UK

1. Introduction

Short-term, medium and long-term forecasting of load demand is necessary for the correct operation of electric utilities. Forecasts are required for proper scheduling activities, such as generation scheduling, fuel purchasing scheduling, maintenance scheduling, investment scheduling, and for security analysis [1].

Generally the forecasting methods found in the literature are typically based on the following forms of mathematical analysis: regressive analysis, exponential smoothing, time series, grey box systems, Kalman filtering, expert systems, wavelet analysis, fuzzy system modelling, neural network modelling, etc. Many models for STLF [2–16,21–24] and MLTLF [1,17–20] have been proposed in the literature. In some cases researchers have combined several methods to develop their own hybrid method. For example in [2] a fuzzy linear regression method is used for load forecasting weekend power usage whereas weekday loads are forecast using a general exponential smoothing method. The latest developments in general load forecasting cited in the literature use artificial intelligence-based forecasting (AIBF) techniques, such as ANN [1–4,6,8,13,14,16–19], fuzzy logic [2,12,14,15], and genetic algorithms [9], all of which show promising results for STLF. Less literature is available regarding the use of AIBF for MLTLF, especially when they utilise ANN [17–19].

Electric load demand is often considered as a function of weather variables and human social activities.

Traditional econometric approaches establish functional relationships between weather variables and current load demand for the forecasting function, often assuming a linear relationship. However, as Park et al. (1991) indicate, the econometric approach may not give sufficiently accurate results because of nonlinear and non-stationary relationships between the load data and weather variables. Therefore, an adaptable technique is needed [5]. An ANN can model any complicated nonlinear relationship and since STLF is a nonlinear problem the ANN forecasting method, which combines nonlinear and time series forecasting methods, is widely applied to STLF. However, it is rarely used in MLTLF since the variations within short-term load forecasting can be considered as a stable random process. The variations within medium to long term load forecasting are not usually random and can be attributed to important factors such as governance within a given country. Therefore, accurate MLTLF is a much more difficult problem because it is difficult to describe the forecasting pattern with obvious formula because of the different and various factors which

* Corresponding authors. Address: School of Mechanical and Aerospace Engineering, Queen’s University, Belfast, Northern Ireland BT9 5AH, UK. Tel.: +44 0 28 90974181; fax: +44 0 28 90975576.
E-mail addresses: c.xia@qub.ac.uk (C. Xia), j.wang@qub.ac.uk (J. Wang), k.mcmenemy@ee.qub.ac.uk (K. McMenemy).
can influence it. However, theoretically MLTLF is also a nonlinear problem for which the ANN approach could be utilized to identify any complicated nonlinear relations, assuming enough data can be collected for training purposes. An ANN model for MLTLF will be developed in this paper.

Recently VI has emerged as a means of load forecasting. The efforts of this paper will concentrate mainly on establishing an accurate, stable ANN model suitable for use in VI, and also the design of a VI for load forecasting. The idea is to combine the numerical potentials of MATLAB with the graphical interface of LabVIEW to present an ANN numerical intensive model with a user friendly interface easy to operate.

The paper is organized as follows: Section 2 presents the structures and features of the ANN used in this research. The preprocessing of the weather data applied to the STLF and the determination of training parameters used is outlined in Section 3. A comparison of forecasting results is also documented in Section 3. Section 4 then presents data collection, modelling considerations and forecasting results of the MLTLF based on the RBFNN. The design process of a VI based on the RBFNN model is introduced in Section 5. Section 6 presents a scheme of a multi-purpose electric load forecaster which integrates database, ANN and VI technology. Finally, the conclusion is given in Section 7.

2. Network features

In this section, a brief background is given regarding the different types of neural networks commonly available. The backward propagation neural network (BPNN) is an overall approximation network. It consists of an input layer, a hidden layer and an output layer. In the approximation, weight adjustment is done according to the gradient descent method. The algorithm has a slow convergence speed, it easily converges to local but non-global minimum and can result in over-training to produce uncertain results. Sigmoid neurons in the hidden layer are able to cover a larger range of inputs, but the number of neurons is fixed before training.

In RBFNN, the radial basis neurons only produce responses in a small area. For a large area of input space it is possible to increase the radial basis neurons to adjust the network in order to reach the precision needed. The network scale is generally bigger than BPNN, but it has features such as adaptive structure, the output being independent of the initial weight value, global and optimal approximation, high precision, quick convergence speed, etc. The generalized regression neural network (GRNN) is an important alternative network of RBFNN, which is applied in function approximation.

RBFNN also consists of input layer, hidden and output layers. The input layer consists of signal nodes. The hidden layer consists of radial basis function neurons and the output layer is linear. The neurons in the hidden layer adopt the radial basis function as an activation function which is generally the Gaussian function:

\[ y = \text{purelin}(w_2a + b_2) \]

where, purelin is a MATLAB function, w2 is the weight vector between the hidden layer and the output layer and b2 is the bias of the output neurons. The b1 can be used to adjust the sensitivity of the function, but normally spread density C is used in actual applications. Generally b1 = 0.8326/C. Before training, the input vector, the target vector and the spread density should be supplied. The value C which is too large or small will result in over-adaptation or non-adaptation in function approximation. Therefore, when designing the network design it is necessary to use a trial and error method until the optimum value for C is found.

The structure of a GRNN is similar to that of a RBFNN, with a RBF hidden layer, but has a different linear output layer to the RBFNN. The function of the linear output layer is given by:

\[ y = \text{purelin}(\text{normprod}(w_2, a)) \]

where, normprod is a normalized dot product weight function in MATLAB.
3. STLF models using ANN and the forecasting results

3.1. Data pre-processing

In any type of NN, accurate data is directly related to the adaptability of the model and its forecasting accuracy. In electrical power load forecasting, historical events can have great effects on data. It is important to reduce the amount of abnormal data within the dataset as this tends to interfere with historical data patterns and hence the forecasting accuracy. Therefore, before any data is used to train the networks it is necessary to eliminate any abnormal data in order to recover the true features of the electric load.

In order to increase convergence speed of network, before training, input and output data of NN is processed at [0, 1] as follows. The historical maximum in the local region needs to be considered, e.g. the load in Yichang is less than 1300 MW. Therefore, the hourly load data could be divided by 1300. Similarly, wind speed divided by 15, and precipitation by 100, atmospheric pressure by 1300, humidity by 100 and day number of the week by seven. The temperature data consists of the highest and lowest temperatures in a day (both given by T). Since temperature has a significant influence on load variation and the possible temperature scope in Yichang is –6 to 40 °C, the highest and lowest temperatures are processed as following:

\[ T_1 = \frac{|T - 17|}{23} \]  

(6)

T1 stands for the processed temperature data. The value 17 in formula (6) is the x value computed according to the following formula (7), which is proposed by us, as we consider that 40 °C or -6 °C has the same influence on the load.

\[ |40 - x| = | -6 - x| \]  

(7)

The value 23 in formula (6) is the x value computed by |40 – x| or |−6 – x|. The relationship curve between \( T_1 \) and T is shown in Fig. 1 which is known as a V-shape model for temperature processing.

3.2. Training parameters and forecasting results

The specimen data set used was established based on actual historical hourly load data and weather data of Yichang in September, 2005, which includes wind speed, precipitation, atmospheric pressure, maximum temperature, minimum temperature, humidity and day of the week. The data of the former 21 d in the above specimen set are employed for training and the rest are used for testing, thus, the hourly load forecasting values of the 22nd day was predicted by means of (i) BPNN, (ii) GRNN and (iii) RBFNN. Trial and error methods were utilized to determine the network structure and parameter. The relative errors between the forecasting values and the actual values are shown in Table 1. The forecasting accuracy was evaluated by the average of absolute percentage errors (APE).

\[ \text{APE} = \left| L_a - L_f \right| / L_a \times 100 \]  

(8)

where \( L_a \) and \( L_f \) respectively are the actual and forecasted hourly loads. The mean absolute percentage error (MAPE) is then computed by:

\[ \text{MAPE} = \left( 1/N_h \right) \sum_{n=1}^{N_h} \text{APE} \]  

(9)

where \( N_h \) is the number of hours in the forecasting period.

Structures and parameters of the BPNN model: this paper choose TRAINLM function in MATLAB to train a neural network with five sigmoid neurons in the hidden layer and one linear neuron in the output layer at 50 epochs and 0.0001 training error goal. Structures and parameters of the GRNN model: the function NEWGRNN was used to build the GRNN model. Trial and error methods were utilized to determine the spread parameter for the model. This was then selected to be 0.4. Structures and parameters of the RBFNN model: the function NEWRB (or NEWRBF) was used to build the RBFNN model. Trial and error methods were utilized to determine the spread parameter for the model. This was then selected to be 5.

In order to ensure the statistical significance of the result and reliability of proposed model, Wilcoxon rank sum test was taken in this paper. The function \([p, h, stats] = \text{ranksum}(x, y, alpha) \) was used for the hypothesis test, performed at the 0.05 significance level. The function returns \( p, h \) and stats from the test. The \( h \)-value stands for the test result, and \( p \)-value is the probability of observing the given result. If \( h = 0 \), it indicates there is no evident difference between the two sets of data \( x, y \). The \( h \)-value has a structure with two fields. The field ‘ranksum’ contains the value of the rank sum statistic and the field ‘zval’ contains the value of the normal (Z) statistic. The actual load value in Table 1 is used for the model. This was then selected to be 5.

4. MLTLF model and the forecasting results

4.1. Considerable factors in MLTLF

There are some conventional MLTLF methods such as the output-value-consumption, electric elasticity-coefficient, gross national product (GNP) synthetic-consumption, per capita consumption, load density, relativity analysis, and total-output-allocation methods. Generally, these methods rely on statistical analysis and need some other forecasting data, such as gross domestic product (GDP), population and the area forecasting value.

In reality, many factors (such as time, weather condition, economy and random effects) affect power system load nonlinearly. However it is difficult to determine what essential factors for a local area should considered for load forecasting. For example, the
Chinese economy has developed very rapidly in recent years, and as such relevant policies are in the process of exploration or improvement and there are many indeterminate factors which will affect load forecasting. Therefore, forecasting results produced using the time series model and a relativity analysis to fit historical data, are not satisfactory. It is important therefore to analyze the load features for the local area, the power consumption structure, the trend of historical load, the development of society and economy, the climate changes, etc., in order to design a forecasting model that adapts well to the local area. The ANN forecasting method can consider enough of these factors to give it a good prospect for accurate prediction.

4.1.1. The classifications of factor

Generally, the factors which affect load can be classified into tendency, season and climate, and random factors. Tendency factors refer to historical load data that has varying tendency, e.g. yearly varying long-term power demand has obvious increasing trend. When considering the tendency factors, it is necessary to analyze historical load data in order to make suitable adjustments at special situations, e.g. a new building project will have an affect on power consumption, whilst a new heat and power plant may lead to abnormal load fluctuations. In such cases, it becomes necessary to adjust historical data to eliminate the influence on a normal trend. In addition, since the tendency item is influenced by the macro economy, it is possible to transversely compare the power consumption of each different type of industry. If industry, commerce, daily life, and urban public utilities are considered as the primary factors, gross industrial output value, total volume of retail sales, level of residential consumption and GDP data from recent years could be collected and edited into a specimen set of neural networks. Since the power consumption of a large power network, which changes both monthly and yearly, is influenced by many factors, in addition to the trend of historical load, other synthetic factors should be considered. The primary factors and their trends should be analyzed in detail. Other tendency factors should also be

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Forecasting results of BPNN, RBFNN and GRNN models.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours</td>
<td>Actual values (MW)</td>
</tr>
<tr>
<td></td>
<td>Forecasting values (MW)</td>
</tr>
<tr>
<td>1</td>
<td>754</td>
</tr>
<tr>
<td>2</td>
<td>744.8</td>
</tr>
<tr>
<td>3</td>
<td>734.3</td>
</tr>
<tr>
<td>4</td>
<td>746.1</td>
</tr>
<tr>
<td>5</td>
<td>724.7</td>
</tr>
<tr>
<td>6</td>
<td>711.8</td>
</tr>
<tr>
<td>7</td>
<td>721.5</td>
</tr>
<tr>
<td>8</td>
<td>734.7</td>
</tr>
<tr>
<td>9</td>
<td>737.1</td>
</tr>
<tr>
<td>10</td>
<td>729.5</td>
</tr>
<tr>
<td>11</td>
<td>684.6</td>
</tr>
<tr>
<td>12</td>
<td>687.6</td>
</tr>
<tr>
<td>13</td>
<td>728</td>
</tr>
<tr>
<td>14</td>
<td>741.2</td>
</tr>
<tr>
<td>15</td>
<td>745</td>
</tr>
<tr>
<td>16</td>
<td>730.4</td>
</tr>
<tr>
<td>17</td>
<td>759.3</td>
</tr>
<tr>
<td>18</td>
<td>747</td>
</tr>
<tr>
<td>19</td>
<td>719.6</td>
</tr>
<tr>
<td>20</td>
<td>686.1</td>
</tr>
<tr>
<td>21</td>
<td>670.5</td>
</tr>
<tr>
<td>22</td>
<td>708.6</td>
</tr>
<tr>
<td>23</td>
<td>730.9</td>
</tr>
<tr>
<td>24</td>
<td>688.6</td>
</tr>
<tr>
<td>MAPE (%)</td>
<td>2.56</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Results for Wilcoxon test.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The results of test at 0.05 significance level</td>
<td>Actual values vs. forecasting values by BPNN</td>
</tr>
<tr>
<td>p, h</td>
<td>Zval, ranksum</td>
</tr>
<tr>
<td>0.2990, 603</td>
<td>0.1134, 582</td>
</tr>
<tr>
<td>p, h</td>
<td>Zval, ranksum</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Load data of Hubei in 1989–1997.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peak load (10^4 KW)</td>
<td>342.4</td>
</tr>
</tbody>
</table>
4.2. Data collection and its pre-processing

Data collection is a very important process for accurate load forecasting. The data that can be collected includes: (1) the statistical data of social and economic development including GDP, total population, output value of 1st, 2nd & 3rd industry; (2) power consumption data of each industry, including power consumption of 1st, 2nd & 3rd industry and the whole society; (3) daily load records, including 8760 hourly load data in each year; (4) forecasting data of the developing trends with regards local GDP and population during the plan year; (5) significant strategy intentions, industrial structure adjustment, major objectives of scientific developments relating to the national economy; (6) electricity price; (7) seasonally varying situations of power consumption or load; (8) weather and climate data; (9) power consumption situations of big consumers and (10) increasing situations of power consumption in other districts at home and abroad.

Accurate data is related to the model adaptability and the forecasting accuracy. Historical accidents or other special factors may affect the data. That is, they may interfere with the historical tendency and influence the forecasting accuracy. Therefore, as previously mentioned it is necessary to eliminate the abnormal data as much as possible in order to have a clearer picture regarding load trends, e.g. when network faults result in power loss in a large area, it is possible to estimate the effect of this and then amend the power consumption data.

4.1.3. Forecasting evaluations

A forecasting evaluation is used to evaluate the accuracy of the forecasting model before an actual forecast is used. Historical data can be used to test the model. A reasonable forecast not only relies on scientific theory, reliable data, and an advanced approach, but also on forecasting experience, logical reasoning ability, capacity of comprehensive analysis and, of course, the judgment ability of the forecasting personnel.

4.2. MLTLF models using ANN methods

4.2.1. MLTLF model based on LM algorithm

As mentioned previously, a BP algorithm usually requires a long training time and quite often it can become fixed at a non-optimized solution. An improvement to the BP algorithm is based on Levenberg–Marquardt (LM) algorithm. The regulation of weight adjustment in LM algorithm is as follows:

\[ \Delta w = (J^T J + \mu I)^{-1} J^T e \]

(10)

where, \( e \) is the error vector and \( J \) is the Jacobian matrix of the network error with respect to weights. The Gauss Newton LM algorithm is a transition algorithm between the steepest descent algorithm and the Newton algorithm. It is a common algorithm used to solve nonlinear least square problems.

The network structure and the training parameters may greatly affect the forecasting accuracy. When there are not enough neurons in the hidden layer, the network can not be properly trained. When they are too many neurons in the hidden layer, the training time will increase and other problems can be encountered such as uncoordinated fitting. More training epochs will make it possible to enhance the accuracy of the model, but may result in a longer training time. The choice of the expected error goal needs to match the neuron number chosen. If the error goal is too small then more nodes and longer training time is necessary. A trial and error method was used to determine the final structure of the updated BP network and training parameters used are as follows: five neurons in the hidden layer, one output neuron, 0.0001 training error goal and 2000 epochs. A specimen set was built on the basis of data in Table 3, whilst the forecasting results are shown in Table 4.

4.2.2. MLTLF model based on RBFNN

The RBF network is based on a functional approximation theory with many excellent features such as adaptive structure, the output being independent of the initial weight value, global and optimal approximation, high precision and quick convergence speed. The training of a RBFNN begins from zero neurons. By inspecting output errors, the neurons in the hidden layer are increased automatically, until the error goal is achieved or the maximum number of neurons in hidden layer has been exceeded. The specimen set for training is built according to the data in Table 3. After comparison of the results using different C values, the spread density C was chosen to be 1.5. Compared with BPN, the RBFNN does not encounter the over-training problem, has a high stability and accuracy, and requires less training parameters and less training time. The forecasting results (shown in Table 4) are close to the annual targets of the electric power plan in the Hubei province (Table 5) and the data published by Hubei Electric Power Dispatching Centre in 2005. For instance, the MAPE values between RBF-based forecasting values in Table 4 and the target values in Table 5 for 2000 and 2005, respectively are 7.17% and 1.94%. These results show that the forecasting model based on RBFNN is superior to that based on BPN.

Notes: data in Table 5 are from ninth five-year plan for electric power in Hubei province (corrected edition), March in 1993.

The results show that the neural network forecasting method is feasible for MLTLF. Even though the available data is insufficient and only the tendency factors in the historical load data are taken into account; a higher forecasting accuracy is achieved. As mentioned previously, if all the data representing politics, economy, population, climate factors in recent years are collected and edited into a specimen, then further forecasting accuracy will be achieved.

5. Design of a virtual load forecaster

LabVIEW is graphical programming software which can be used to establish a VI. MATLAB, however, is still thought to be superior with regards numerical analysis and processing. Therefore, MATLAB is integrated into software package LabVIEW to design a VI. Library functions are available within LabVIEW that makes it convenient to connect to software standards such as TCP/IP, SQL.
database, DDE, and Active X. MATLAB also has a script node to link LabVIEW. Using this script node, MATLAB programs were connected into the block diagrams of LabVIEW. The node mode was used to call MATLAB programs in LabVIEW, in order to implement ANN-based computations, data processing and graphical display for forecasting results.

In order to create the VI, it was necessary to design a user front end panel. Seven numeric controls are used to input wind speed, precipitation, atmospheric pressure, maximum temperature, minimum temperature, humidity and day of the week, respectively. Three dialog controls are also used to input day, month and year. Finally two graph indicators are used to indicate a daily load curve and an error curve whilst one numeric indicator is used to output the peak load. The block diagrams are created using standard scripts. The MATLAB code used to implement the previous RBFNN was connected to LabVIEW via a node. The real historical hourly load data and the weather data of Yichang in September in 2005 are contained in the M file. Once the VI is created, the weather data for the forecasting period (from official weather forecasting department) can be inputted. After training the network using data from the former 21 d and inputting the weather forecasting data of the 22nd day, the hourly load forecasting values of the 22nd day was obtained. The output from the VI is a daily load curve for the forecast day (the 22nd day), in addition to the corresponding relative error curve and daily peak load. The results show that the VI is able to display the steady outcomes. Therefore, this test has illustrated the superiority of the RBFNN model. The front panel and block diagram are shown in Figs. 2 and 3.

If we have an ANN forecasting model for MLTLF and follow the example of STLF, we can also establish a relevant VI for MLTLF.

6. Discussion

The main purpose of a forecasting VI is to obtain multiple types of forecasting results (e.g. figure, curve and chart). The flowchart of a multi-purpose virtual load forecaster is shown in Fig. 4, where the data for building a database includes historical daily load, power consumption of each industry, weather data and climate features, GDP, population, industry output values, electricity price and so on. After the database has been established, considering some expert advice, the main factors which have close relationships with the load in local area are used. Then by using the VI the operator can select different forecasting periods and select data from the database according to load features for different areas to build up a relevant specimen set for ANN training. In this paper only actual historical daily load and weather data such as wind speed, precipitation, atmospheric pressure, maximum temperature, minimum temperature and humidity are considered to illustrate a STLF. Concerning MLTLF, even though the tendency factors in the historical load data were only used, higher forecasting accuracy was achieved. The forecasting accuracy of the model is generally evaluated by error between forecasting values and actual loads. In the illustration for STLF, MAPE for hourly load is 2.13%. As for MLTLF, the growing trend was better estimated and an acceptable MAPE (7.17–1.94%) was also obtained. If we have a multi-purpose virtual load forecaster as mentioned above, it would
be possible to acquire a model with higher accuracy. Of course, this should be a very fruitful area for future research.

7. Conclusion

This paper introduces a method that combines neural network models of load forecasting with virtual instrument technology in order to build a virtual forecaster. It also presents the scheme of a multi-purpose virtual load forecaster, to establish a united database for load forecasting and to combine short-term load forecasting with medium and long term load forecasting. When establishing a short-term load forecasting model, we should take the influence of weather factors on load into account. Since a higher temperature or a lower temperature has greater influence on load, a V-shape model to process temperature is proposed. Three neural network models for STLF and two methods in implementing MLTFL are introduced. By comparing results of short-term, medium and long term load forecasting, the result demonstrates that forecasting model based on RBFNN is effective and has high stability. Therefore, RBFNN is more suitable to the applications in design of load forecasting instruments. The virtual load forecaster is easy to implement and simple to operate. In addition, the forecaster is intuitive. Many useful curves, such as daily load curves, their corresponding relative-error curves and the value of daily peak loads can be displayed in the virtual load forecaster.

References