Research on Short-term Load Forecasting of the Thermoelectric Boiler Based on a Dynamic RBF Neural Network

Wei-Bao DAI, Ping-Hua ZOU, Cheng-Xian, YAN
School of Municipal and Environmental Engineering
Harbin Institute of Technology, Harbin, 150001, China
E-mail: dwb750424@126.com

Abstract: As thermal inertia is the key factor for the lag of thermoelectric utility regulation, it becomes very important to forecast its short-term load according to running parameters. In this paper, dynamic radial basis function (RBF) neural network is proposed based on the RBF neural network with the associated parameters of sample deviation and partial sample deviation, which are defined for the purpose of effective judgment of new samples. Also, in order to forecast the load of sample with large deviation, sensitivity coefficients of input layer is given in this paper. To validate this model, an experiment is performed on a thermoelectric plant, and the experimental result indicates that the network can be put into extensive use for short-term load forecasting of thermoelectric utility.

Key words: dynamic RBF neural network; load forecasting; partial sample deviation; input layer sensitivity coefficient; thermoelectric boiler

1. INTRODUCTION

For a running thermoelectric utility, because of its thermal inertia and the various factors for determining the quantity of heat and power supplied to consumers, its regulation becomes very complicated, so its short-term load must be forecasted accurately according to associated parameters.

Short-term load forecasting is aimed at predicting a system load for a period of minutes, hours, days or weeks. Short-term load forecasting plays an important role in the real-time control and the security functions of an energy management system. Throughout the last decades, considerable research efforts have been devoted to short-term load forecasting and the enhancement of its overall accuracy. So far, a variety of models that have surfaced in the field of short-term load forecasting include the exponential smoothing model, state estimate model, multiple linear regression model and stochastic model [1-4]. Generally, these techniques are based on statistical methods and extrapolated past load behavior while taking into account the effect of other influencing factors such as weather and day of week. However, the techniques used for these models employ a great number of complex and non-linear relationships between the load and the factors. A large amount of computational time is required and may lead to numerical instabilities. Some deficiencies in the presence of an abrupt change in weather or environmental variables are also believed to affect the load forecasting.

As a result of the development of artificial intelligence (AI), many artificial neural networks (ANNs) have been proposed and applied to solve short-term load forecasting problems [5-7]. It is known that ANNs do not require any implicitly defined relationship between input and output variables. The corresponding mapping between input and output is obtained employing a training algorithm. The ANNs modeling needs only the selection of input and corresponding output variables, thus avoid the difficulties with conventional modeling processes. Peng *et al.* [8] proposed a minimum distance based strategy to identify the appropriate historical patterns of load and temperature for training the network. A partially connected network consisting of main and supporting blocks is also proposed, which makes use of model reference and functional relationships between input and output variables. With the development of genetic algorithm (GA), Dipti Srinivasan [9] presents an ANN evolved by a genetic algorithm for short-term load forecasting. Also, Ruey-Hsun Liang [10] proposes an ANN combined with a fuzzy system, the objective of which is to study a self-organizing fuzzy-neural network. However, the structures of these two types of ANN are very complex.

All these prior works focus on the application of back propagation training algorithms to train and update the model parameters, and those models have a set of problems which include the dependence on initial parameters and long training time. On the other hand, for a radial basis function (RBF) neural network, because of the initial center of its training center is the subset of samples space, the training time is very short than that of back propagation training algorithm[11].

In this paper, dynamic radial basis function (RBF) neural network is proposed based on RBF neural network to address the first problem, and for the validation of this model, an experiment is carried out in a thermoelectric plant.

2. DYNAMIC RBF NEURAL NETWORK MODEL

2.1 Definitions of SD, PSD and $SCIL$

A dynamic RBF neural network is a RBF neural network whose nodes of hidden layers can be increased dynamically and automatically when there is a large deviation between a new sample and the training samples. This model is supported by judgment and programming mechanism, and definitions of sample deviation (SD), partial sample deviation (PSD) and sensitivity coefficients of input layer ($SCIL$). To definite SD, PSD and $SCIL$, a new sample X_{new}, training samples X, and a training sample X_k with its output Y_k must be given firstly:

$$X_{new} = [x_{new1}, x_{new2}, \cdots, x_{newm}, \cdots, x_{newM}] \quad (m = 1, 2, \cdots, M) \tag{1}$$

Where M is the dimension of a sample, and x_{newm} is the m^{th} node of input layer.

$$X_k = [x_{k1}, x_{k2}, \cdots, x_{km}, \cdots, x_{kM}] \quad (k = 1, 2, \cdots, N;\ m = 1, 2, \cdots, M) \tag{2}$$

Where N is the dimension of training space.

$$X = [X_1, X_2, \cdots, X_k, \cdots, X_N] \quad (k = 1, 2, \cdots, N) \tag{3}$$

$$Y_k = [y_{k1}, y_{k2}, \cdots, y_{kj}, \cdots, y_{kJ}] \quad (k = 1, 2, \cdots, N;\ j = 1, 2, \cdots, J) \tag{4}$$

Where J is the dimension of output layer.

With a new sample X_{new} and a training sample X_k, the parameters of SD and PSD are defined as follows:

$$SD = [SD_1, SD_2, \cdots, SD_k, \cdots, SD_N] \quad (k = 1, 2, \cdots, N) \tag{5}$$

$$SD_k = \frac{1}{M}\sum_{m=1}^{M}\left|\frac{x_{km} - x_{newm}}{x_{newm}}\right| \tag{6}$$

Where x_{km} and x_{newm} are described in equations (2) and (1).

$$PSD = [PSD_1, PSD_2, \cdots, PSD_m, \cdots, PSD_M] \quad (m = 1, 2, \cdots, M) \tag{7}$$

$$PSD_m = \frac{x_{km} - x_{newm}}{x_{newm}} \tag{8}$$

The sensitivity coefficients of input layer are the partial derivatives of node of output layer with respect to that of input layer, which can be described as:

$$SCIL_{mj} = \partial y_{kj} \big/ \partial x_{km} \quad (m = 1, 2, \cdots M;\ j = 1, 2, \cdots, J) \tag{9}$$

$$SCIL = \begin{bmatrix} SCIL_{11}, \cdots, SCIL_{1j}, \cdots, SCIL_{1J} \\ SCIL_{21}, \cdots, SCIL_{2j}, \cdots, SCIL_{2J} \\ \cdots \\ SCIL_{m1}, \cdots, SCIL_{mj}, \cdots, SCIL_{mJ} \\ \cdots \\ SCIL_{M1}, \cdots, SCIL_{Mj}, \cdots, SCIL_{MJ} \end{bmatrix} \tag{10}$$

The sensitivity coefficients of input layer can be calculated by perturbing each node of input layer, one at a time, by a small amount (δx_{km}), keeping the other nodes of input layer constant and evaluating the change in the calculated value of the nodes of output layer, δy_{kj}. The sensitivity coefficient of input layer is then

$$SCIL_{mj} = \partial y_{kj} / \partial x_{km} = \delta y_{kj} / \delta x_{km} \quad (m = 1, 2, \cdots M;\ j = 1, 2, \cdots, J) \tag{11}$$

δx_{km} is a small value such as $x_{km}/10$ or $x_{km}/100$.

2.2 Judgment mechanism for a new sample X_{new}

The following two inequations are criterions for the judgment of a new sample X_{new}:

$$\min(SD_1, SD_2, \cdots, SD_k, \cdots, SD_N) > 0.2 \tag{12}$$

$$|PSD_m| > 0.5,\ (m = 1, 2, \cdots, M) \tag{13}$$

If a new sample X_{new} consists with any one of the inequations above-mentioned, that means there is a large deviation between the new sample and training space, the new sample should be added to training space in order to update the network.

2.3 Programming mechanism

To practise a dynamic RBF neural network, not only lies on the judgment of a new sample, but also on the network structure predefined. During programming, the structure of a dynamic RBF neural network should be defined initially. In the short-term forecasting neural network of this paper, a three-layer network is used with 18 input nodes, 50 hidden nodes and one output node. The input nodes are crucial running parameters which include temperature of feedwater, pressure of feedwater, flow of feedwater, temperature of main steam, pressure of main steam, flow of main steam, water flow of desuperheater, inlet steam pressure of reheater, inlet steam temperature of reheater, outlet steam pressure of reheater, outlet steam temperature of reheater, water flow of reheater attemperator, oxygen percentage of flue gas, temperature of exhaust flue gas, inlet air temperature of air heaters, inlet air flow of forced fans, speed of coal feeder and water flow of continuous blowdown. And output node is load of thermoelectric utility. All these parameters must be normalized before use. Among hidden nodes, 7 nodes are activated at first while others will always be in dormancy only if a new sample consists with any one of inequations (12) and (13).

When forecasting a short-term load of a new sample, judgment should be made firstly, if it does not consist with any one of inequations (12) and (13), the forecasting procedure will be performed based on RBF neural network, otherwise, the sensitivity coefficients of input layer and the partial sample deviation will be employed to forecast the load:

$$y_{newkj} = (1 + \sum_{m=1}^{M} SCIL_{mj} PSD_m x_{newm}) y_{kj} \quad (k = 1, 2, \cdots, N;\ j = 1, 2, \cdots, J) \tag{14}$$

where y_{kj} is the most similar sample in training space to new sample X_{new}.

After the forecasting, the new sample will be added to training space together with one hidden node be activated, and train the network again for next forecasting.

2.4 Evaluation of results

To test the performance of the network, the relative percentage error (RPE) is used and defined as follows:

$$RPE = \left| \frac{actual_i - forecast_i}{actual_i} \right| \times 100 \tag{15}$$

in the formulation, $actual_i$ is the actual load of sample i and $forecast_i$ is the forecasted load of that sample.

3. RESULTS

To demonstrate the effectiveness of the proposed approach, short-term load forecasting was performed on a thermoelectric plant. A great mount of data are collected every hour, and 15 groups of data are selected as samples for the test. Samples 1 to 9 are employed for training the network, and others are used for verifying the result given by the network.

Tab. 1 SD of samples 10 to 15

samples	*10*	*11*	*12*	*13*	*14*	*15*
1	0.053	0.046	0.061	0.036	0.242	0.238
2	0.037	0.056	0.055	0.023	0.234	0.232
3	0.049	0.062	0.065	0.034	0.23	0.228
4	0.031	0.076	0.071	0.033	0.233	0.231
5	0.036	0.065	0.074	0.028	0.256	0.252
6	0.031	0.083	0.058	0.024	0.197	0.182
7	0.038	0.083	0.073	0.033	0.246	0.243
8	0.035	0.047	0.039	0.036	0.24	0.234
9	0.019	0.054	0.058	0.024	0.249	0.245

Tab. 2 Number of PSD greater than 0.5

samples	*10*	*11*	*12*	*13*	*14*	*15*
1	0	0	0	0	2	2
2	0	0	0	0	3	3
3	0	0	0	0	2	2
4	0	0	0	0	3	3
5	0	0	0	0	3	3
6	0	0	0	0	1	1
7	0	0	0	0	3	3
8	0	0	0	0	3	3
9	0	0	0	0	2	2

Tab. 3 RPE (%) of different forecasting methods

samples	*RBF neural network*	*RBF neural network with* $SCIL$ and PSD	*Dynamic RBF neural network*
10	1.801	1.801	1.803
11	1.796	1.796	1.781
12	2.273	2.273	2.276
13	1.406	1.406	1.409
14	6.426	3.459	
15	6.851	3.588	2.189

According to equations (6) and (8), SD and PSD of samples 10 to 15 are calculated, which are shown in table 1 (SD) and table 2 (the number of PSD greater than 0.5).

As is shown in Table 1, the sample deviations of samples 10 to 13 are all less than 0.2, however, among the deviations of samples 14 and 15, only (14, 6) and (15, 6) are less than 0.2. And in Table 2, the partial sample deviations of samples 10 to 13 are all less than 0.5, while the partial sample deviations of samples 14 and 15 are all great than 0.5. So samples 14 and 15 are not similar samples to that of training space, and one of them should be added to training space to update the original network.

Figure 1 is the load curve gained by using RBF neural network model. The forecasting resultsobtained by RBF neural network model, RBF neural network model with $SCIL$ and PSD and dynamic RBF neural network model are shown in Fig.2, Fig.3 and Fig.4 respectively. Table 3 compares the *RPE* of results obtained by different models. As is indicated, the *RPE* of samples 10 to 13 are very small while that of samples 14 and 15 are quite different by using various models. It can be seen that the two samples' *RPE* are largest based on RBF neural network model due to large *SD* and *PSD*, and by using *SCIL* and *PSD*, their *RPE* becomes smaller and acceptable, the most should be emphasized is that the *RPE* of sample 15 gets very small after the addition of sample 14 to original training space.

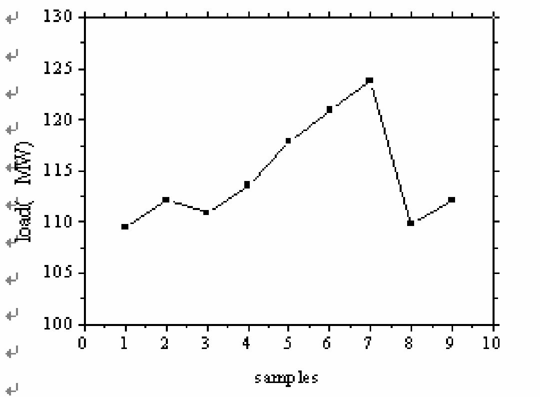

Fig. 1 Loads curve with RBF neural network

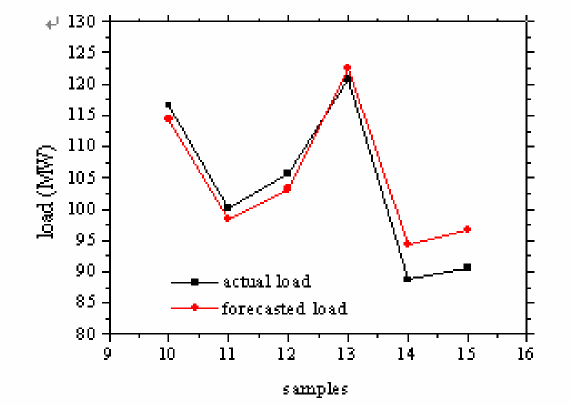

Fig. 2 Forecasting results with

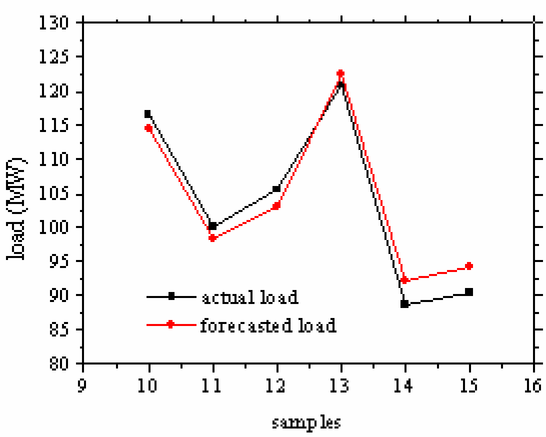

Fig. 3 Forecasting results with ***PSD*** **and** ***SCIL***

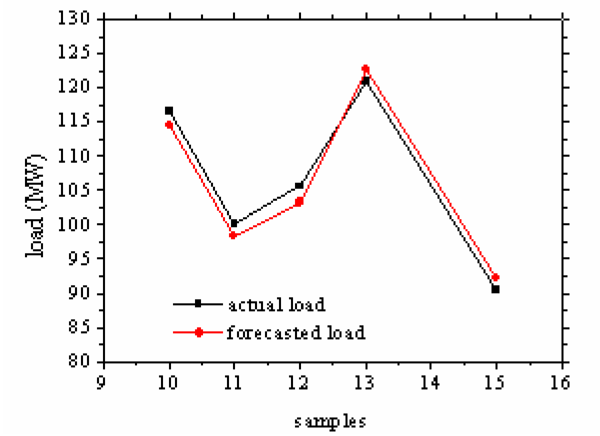

Fig. 4 Forecasting results with dynamic RBF neural network

4. CONCLUSIONS

In this paper, dynamic neural network is proposed together with three significant parameters which are defined as sample deviation, partial sample deviation and sensitivity coefficient of input layer. To evaluate this model, an experiment was performed on a thermoelectric utility. The result indicates that sample deviation and partial sample deviation are two crucial parameters for the judgment of a new sample. Based on the judgment, partial sample deviation and sensitivity coefficient of input layer, the load of a new sample with large sample deviation or partial sample deviation can be forecasted correctly. The most significant is that the model can learn new knowledge by the addition of a new sample which has a large deviation or partial sample deviation to original training space. This model can also be employed for the forecasting of utility efficiency, unburned carbon in ash and other short-term forecastation.

REFENRENCES

[1] Hagan M T, Behr S M. The time series approach to short term load forecasting[J]. IEEE Trans on Power Systems, 1987, (2):785-791.

[2] Papalexopoulos A D, Hesterberg T C. A regression-based approach to short term system load forecasting[J]. IEEE Trans on Power Systems, 1990, 5(4):1535-1544.

[3] Moghram I, Rahman S. Analysis and evaluation of five short term load forecasting techniques[J]. IEEE Trans on Power Systems. 1989, 4(4):1484-1491.

[4] Haida T, Muto S. Regression based peak load forecasting using a transformation technique[J]. IEEE Trans on Power Systems, 1994, 9:1788-1794.

[5] Chen S T, Yu D C. Weather sensitive short term load forecasting using a nonfully connected artificial neural network[J]. IEEE Trans on Power System. 1992, 7(3):1098-1105.

[6] Park D, Marks R. Electric load forecasting using a artificial neural network[J]. IEEE Trans on Power Systems, 1991, 6(2):442-449.

[7] Peng T, Karady G. An adaptive neural network approach to one week ahead load forecasting[J]. IEEE Trans on Power Systems, 1993, 8(3):1195-1202.

[8] Peng T M, Hubble N F. Advancement in the application of neural networks for short-term load forecasting[J]. IEEE Trans on Power Systems, 1992, 7(1):250-257.

[9] Dipti Srinivasan. Evolving artificial neural networks for short term load forecasting[J]. Neurocomputing, 1998, 23:265-276.

[10] Ruey-Hsun Liang, Ching-Chi Cheng. Short-term load forecasting by a neural-fuzzy based approach[J]. Electrical Power and Energy Systems. 2002, 24:103-111.

[11] Fredric M. Ham, Ivica Kostanic. Principles of Nerocomputing for Science and Engineering [M]. 2003, 140-146