2005년도 대한전기학회 하계학술대회 논문집 2005. 7. 18 - 20

신경회로망을 이용한 배전용 변압기의 단기부하예측

김병수*, 신호성*, 송경빈*, 박정도**
숭실대학교*, 위덕대학교**

Short-Term Load Forecasting of Pole-Transformer Using Artificial Neural Networks

Byoung-Su Kim*, Ho-Sung Shin*, Kyung-Bin Song*, Jung-Do Park**
Soong-Sil University*, Uiduk University**

Abstract - In this paper, the short-term load forecasting of pole-transformer is performed by artificial neural networks. Input parameters of the proposed algorithm are peak loads of pole-transformer of previous days and their temperatures. The proposed algorithm is tested for one of the pole-transformers in seoul, korea. Test results show that the proposed algorithm improves the accuracy of the load forecasting of pole-transformer compared with the conventional algorithm.

1. INTRODUCTION

The electric power quality is one of the hot issues in power systems. The status of pole-transformer directly affects electric power consumers in their electric power quality. The electric power supply interruption caused by pole-transformers brings about great inconvenience and economical losses to the customers. Furthermore, blackout is one of the worst events to deteriorate the quality of electric power and reliability of the power utility. An unexpected over-loads of electric power demands causes a sport of oil or damage to pole-transformers. Therefore, in order to prevent an area from being damaged by over-loads, peak load of pole-transformer should be forecasted with a reasonable manner and then distribution systems should be planned according to the forecasted load demands.[1]

If the planning or the operation of a distribution system can not cope with or consider the over-loads of the distribution system, the following problems may arise: precipitation of reliability, economical loss by a blackout and a physical damage by a explosion of equipment . Thus, the accurate load forecasting of pole-transformer is essential to reduce accidents and increase reliability in distribution systems.

A number of studies for the load forecasting have been reported in many literatures. In recent 10 years, many researchers have focused on the artificial neural networks for the load forecasting. The neural network technique makes a difference of the forecasting accuracy according to the learning pattern. Hence, this paper investigates various cases and finds the pattern with the most superior correctness.

2. LOAD FORECASTING OF POLE-TRANSFORMER

The purpose of this paper is to forecast loads of pole-transformers for stable electric power supply to the customers. That purpose is to take steps against to prevent unexpected over-loads of pole-transformers. For load management of secondary distribution systems, it is focused on forecasting the peak load of transformer and deciding the over-loads of the transformer.

So, this paper contributes to improvement of supply reliability for customers and the distribution automation by forecasting the peak load of pole-transfers. The flowchart of the proposed algorithm is shown in figure 1.

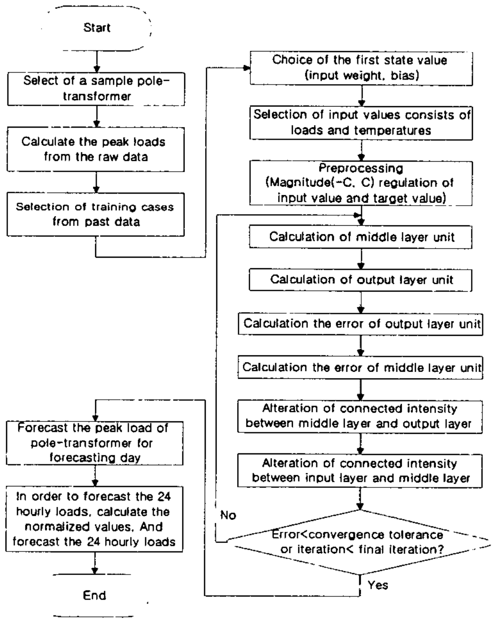

Figure 1. The flowchart of the proposed algorithm

The proposed algorithm

Step1) Calculate the peak loads from raw data of a sample pole-transformer.

Step2) Training set is selected from past load data. And input data is selected from loads and temperatures.

Step3) The variable range of the data has from 0 to 10 million-kW but, in the case of temperature data, it has from -10 to 40 Celsius. Therefore, the range of load and temperature are regulated.

Step4) Training by error back-propagation algorithm[2]

Step5) Forecast the peak load by trained neural network.

In the step 2), the peak loads obtained by the following raw data: voltage, current and power factor. From this data, real and reactive powers can be calculated. Then, peak real power can be calculated by using selected 7 days data from training case. Besides, the proposed method includes max/min temperatures of the selected area.

2.1 Application of algorithm using artificial neural networks

The proposed neural networks which consist of three layers of the input, the hidden layer and the output layer forecast for five input variables. The input layer has five elements or nodes for a forecast for time t;[3][4].

- x_1 - Load at $t-7$ $L_{max}(d-7)$
- x_2 - Max. Temperature at $t-7$ $T_{max}(d-7)$
- x_3 - Min. Temperature at $t-7$ $T_{min}(d-7)$
- x_4 - Max. Temperature at $t-7$ $T_{max}(d)$
- x_5 - Min. Temperature at $t-7$ $T_{min}(d)$

Where, d is a forecasted day. The neural networks can be described as input vector $x^{(k)}$ to kth target output vector $o^{(k)}$. And load and temperature data of the input variables has very different ranges, the use of the original data to the neural network may cause a convergence problem. To avoid this problem, it is necessary to scale the variable range of the data.
The input and target output parameters are scaled in the following way.

$$x_{i,\in}^{(k)} = \frac{x_i^{(k)} - \tilde{x}_i}{s_i}, \ (i=1,\ldots 5; k=1,\ldots,7) \qquad (1)$$

$$o_{\in}^{(k)} = \frac{o^{(k)} - \tilde{o}}{s_o} \qquad (2)$$

Where, $s_i (i=1,\ldots,5)$ is an estimate of the standard deviation of $x_i^{(k)}$, $s_0$is an estimate of the standard deviation of $o^{(k)} (k=1,\ldots,7)$ and $\tilde{x}$is the average value for ith component of the input vector.
Based equation of neural network is as follows;

$$y_i = \sum_{j=1}^{5} w_{ij} x_j + \theta_i \qquad (3)$$

where W_{ij} (i=1, ⋯, n, j=1, ⋯, 5) are the weight factors between the input nodes and the hidden nodes; $\theta$is the threshold value at the hidden nodes defined by the algorithm; and x_j (j=1, ⋯, 5) are the input elements as defined above.
y_i is input value of log-sigmoid transfer function that is trained using back propagation algorithm.

$$F(y_i) = \frac{1}{1+\exp(-y_i)} \qquad (4)$$

In the network, the output is a sum of weighted nonlinear terms given by

$$\hat{o} = \sum_{i=1}^{n} W_i F(y_i) \qquad (5)$$

where, $\hat{o}$is the single component of the output vector and represents the forecasted load.
The error function $E$of neural network is defined as;

$$E = \frac{1}{2}\sum_{k=1}^{K}(o^{(k)} - \hat{o}^{(k)})^2 \qquad (6)$$

where, K is the number of cases used to train the network, $o^{(k)}$ is the actual output of the training case k, and $\hat{o}^{(k)}$ is the forecast output of the network. Here, the convergence tolerance is set at 10^{-4} and iterations are set at 10^5.
This algorithm is a kind of steepest-descent algorithm used to minimize the error function.

$$T_{p+1} = T_p - \alpha \frac{\partial E}{\partial T_p} \qquad (7)$$

$$T = \{w_{ij}, W_i, \theta_i\} \ (i=1, \cdots, n; j=1, \cdots, 5) \qquad (8)$$

where, p is the iteration index, α is the learning step length (e.g. 0.01), and $\frac{\partial E}{\partial T_p}$ is the gradient vector of E with respect to T at the pth iteration.

2.2 Selection of training cases

The selection of 7 days training case is shown in Table1, 2, 3 and 4.

Table 1. Selection of 7 days for the previous days

Sun	Mon	Tue	Wed	Thu	Fri	Sat
17	18	19	20	21	22	23
24	25	26	27	28	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19 (d-1)	20 (d-2)	21 (d-3)	22 (d-4)	23 (d-5)
24 (d-6)	25 (d-7)	26 d day	27	28	29	30

Table 2. Selection of 7 days for the previous weekday

Sun	Mon	Tue	Wed	Thu	Fri	Sat
17	18	19	20	21	22	23
24	25	26	27	28	1	2
3	4	5	6	7	8	9
10	11	12	13 (d-1)	14 (d-2)	15 (d-3)	16
17	18	19 (d-4)	20 (d-5)	21 (d-6)	22 (d-7)	23
24	25	26 d day	27	28	29	30

Table 3. Selection of days for the previous same type days

Sun	Mon	Tue	Wed	Thu	Fri	Sat
3	4	5(d-1)	6	7	8	9
10	11	12(d-2)	13	14	15	16
17	18	19(d-3)	20	21	22	23
24	25	26(d-4)	27	28	1	2
3	4	5(d-5)	6	7	8	9
10	11	12(d-6)	13	14	15	16
17	18	19(d-7)	20	21	22	23
24	25	26d day	27	28	29	30

Table 4. Selection of 14 days for the previous weekday

Sun	Mon	Tue	Wed	Thu	Fri	Sat
17	18	19	20	21	22	23
24	25	26	27	28 (*d*-1)	1 (*d*-2)	2
3	4	5 (*d*-3)	6 (*d*-4)	7 (*d*-5)	8 (*d*-6)	9
10	11	12 (*d*-7)	13 (*d*-8)	14 (*d*-9)	15 (*d*-10)	16
17	18	19 (*d*-11)	20 (*d*-12)	21 (*d*-13)	22 (*d*-14)	23
24	25	26 *d* day	27	28	29	30

In Table 1, input data is applied by 7 days for the previous days. Table 2 shows a case of which input data is applied by 7 days for the previous weekday by excluding Monday, weekday. Input data of Table 3 applied by 7 days for the previous same type days. And Table 4 shows a case of which input data is applied by 14 days for the previous days.

3. TEST RESULTS

In this test, learning rate (α) is 0.01 and the number of neurons in a hidden layer is 8. The training is performed for all training cases. If error converges within 10^{-4}, the training is stopped. Otherwise, the training is calculated for 10^5 times. An error of the load forecasting is defined as:

$$Error(\%)=\frac{\left|P_t^{Forecast}-P_t^{Actual}\right|}{P_t^{Actual}}\times 100 \qquad (9)$$

Where, $P_t^{Forecast}$ is the forecasted load of the predicted day; P_t^{Actual} is the actual load of the predicted day.

The accuracy of the load forecasting depends on the number of hidden layer and the training case. The table 5. indicates that the algorithm is the highest accuracy at 8 hidden layers.

Table 5. Results for various numbers of hidden layer

hidden layer	11.27 (Tue)	11.28 (Wed)	11.29 (Thu)	11.30 (Fri)	Error(%)
6	1.6582	9.2508	11.3227	7.9985	7.5576
8	1.5961	8.1115	10.7136	7.7457	7.0417
10	1.6094	9.3682	11.1476	8.0871	7.5530
12	1.7606	8.493	11.5724	8.2634	7.5224
14	1.8293	9.2114	11.0226	8.4472	7.6276
16	1.7022	8.9685	10.8853	7.9837	7.3849

Table 6. Results of various training cases

Training case	Selection of 7 days for the previous days	Selection of 7 days for the previous weekday
11.27(Tue)	1.6228	20.3834
11.28(Wed)	17.4533	16.3788
11.29(Thu)	11.7524	13.9296
11.30(Fri)	2.4462	13.9677
Error(%)	8.3134	16.1642
Training case	Selection of 7 days for the previous same type days	Selection of 14 days for the previous weekday
11.27(Tue)	1.6356	17.9026
11.28(Wed)	9.1620	14.948
11.29(Thu)	10.6533	15.9475
11.30(Fri)	7.8966	14.3647
Error(%)	7.3369	15.7907

The table 6. indicates that Case 3 yield the higher accuracy than the other cases. Consequently, Case 3 and hidden layer(8) are selected for the best training set. The following Table 7 shows the test results during November 20~23, 2001 and March 26~29, 2002.

Table 7. The results of the forecasting accuracy for the proposed algorithm

Dates	Error(%) of selected case	Dates	Error(%) of selected case
3.26(Tue)	0.4265	11.20(Tue)	0.6292
3.27(Wed)	0.9305	11.21(Wed)	5.3854
3.28(Thu)	1.3845	11.22(Thu)	1.0629
3.29(Fri)	4.5217	11.23(Fri)	1.7276
Average	1.8158	Average	2.2013

Compared with other cases, test results show that the proposed algorithm improves the accuracy of the load forecasting for the pole-transformer.

4. CONCLUSION

The status of pole-transformer directly affects electric power consumers in their electric power quality. Furthermore, the electric power supply interruption caused by pole-transformer brings about great inconvenience and economical losses to the customers. Therefore, in order to prevent an area from being damaged by over load, peak loads of pole-transformer should be forecasted with reasonable manners.

The proposed method is tested with sample systems, which indicates that the proposed algorithm improves the accuracy of the load forecasting of the pole-transformer.

ACKNOWLEDGMENT

This work has been supported by EESRI (02-Jeon-01), which is funded by MOCIE(Ministry Of Commerce, Industry and Energy).

[REFERENCES]

[1] *A study on the optimal load management of distribution transformer*, KEPRI, Technical Report, 2003. 2.

[2]Kim sang Yun, *Pattern Recognition & Learning Using MATLAB*, HongReung Science Publisher. 2003.

[3]Ha Sung Gwan, "Short-Term Load Forecasting using Neural Nerworks and The Sensitivity Of Temperatures in the Summer Season", Soong-Sil University, Master Thesis of Soong-Sil University, Feb. 2005.

[4]I. Drezga and S. Rahman, " Input variable selection for ANN-based short-term load forecasting", IEEE Trans. Power Systems, vol.13, no. 4, pp. 1238-144, 1998.