Asian Journal of Information Technology 6 (2): 154-159, 2007

PSO Based Combinations of ANNs for Short Term-Daily Peak Load Forecasting

[1]P. Subbaraj and [2]V. Rajasekaran
[1]Department of EEE, Kalasalingam University, PSNA College of Engineering and Techonlogy, India
[2]Krishnankoil, Tamilnadu, Dindigul, Tamilnadu, India

Abstract: This study presents a new approach for short term - daily peak load forecasting using Particle Swarm Optimization based Combinations of Artificial Neural Network (PSOCANN) modules. In this study, a set of neural networks has been trained with different architecture and with different training parameters. The Artificial Neural Networks (ANNs) are trained and tested for the actual load data of Chennai city (India). A method of optimal linear combination is used to combine selected networks to produce better results, rather than using a single best trained ANN. The obtained test results indicate that the proposed method of approach improves the accuracy of the load forecasting.

Key words: Combination of artificial neural network, short term-daily peak load forecasting, particle swarm optimization

INTRODUCTION

Load forecasting is an important component of power system to establish economical and reliable operations for power stations and their generating units. An accurate load forecasting approach, used to predict load demand, is the essential part of any energy management system (Papadaki *et al*., 1998; Tomonobu *et al*., 2002; Desouky and Kateb, 2000; Bakirtzis *et al*., 1995; Rehman and Hazim, 1993; Peng *et al*., 1992). Particularly peak load forecasting plays a key role in generation scheduling. The accuracy and consistency in load forecasting increase the efficiency of daily system operations and decrease the production costs. In order to obtain a high forecast accuracy and flexibility apart from conventional models such as time-series, regression models, the artificial neural networks have been developed for load forecasting. The attention and attraction of ANN is the ability to learn complex and non-linear relationships through a training process with the use of historical data and weather information (Lu *et al*., 1993; Alex *et al*., 1994; Lee and Park, 1992; Al Fuhaid *et al*., 1997; Dregza and Rahman, 1998; Dash *et al*., 1993).

In this study, a number of neural networks are trained with different architecture and with different training parameters for the given input and output relationships. Of these trained networks, fifteen neural networks with best performance are selected for various combinations to develop a combination module for load forecasting application rather than using only the single best trained ANN. The Optimal Linear Combination (Sherif and Bruce, 1995) of these trained networks is achieved by the computational technique, Particle Swarm Optimization. Using the selected fifteen neural networks, fourteen different combinations can be created for the combination module with the above reference technique and all the combinations are tested and the results of these combinations are compared themselves and with the conventional ANN (single best trained network) with best performance.

The combinations that are developed for the combination module is proposed to achieve Short Term – Daily Peak Load Forecasting, where the objective is to predict daily peak load for the 1st week of November 2005 (winter) for the power system of Chennai city (Tamilnadu State - India).

ARCHITECTURE OF ANN

The ANN is modeled with one input layer of 24 neurons, one hidden layer and one output layer with

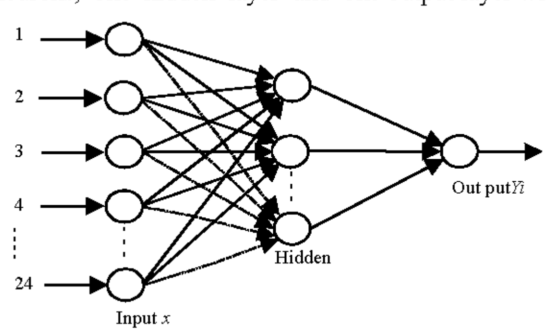

Fig. 1: General architecture representation of ANN

Corresponding Author: P. Subbaraj, Department of EEE, Kalasalingam University, PSNA College of Engineering and Techonlogy, India

Table I: Selected input variables

Input variables	Index	
Peak Load of previous day	L(d–1)	(1)
Temperature of previous day (Mean, Max, Min)	T(d–1)	(2-4)
Relative Humidity of previous day (Mean,Max,Min)	RH(d–1)	(5-7)
Wind Speed of previous day (Max)	WS(d–1)	(8)
PeakLoad of previous week	L(d–7)	(9)
Temperature of previous week (Mean,Max,Min)	T(d–7)	(10-12)
Relative Humidity of previous week (Mean,Max,Min)	RH(d–7)	(13-15)
Wind Speed of previous day (Max)	WS(d–7)	(16)
Temperature of forecast day (Mean,Max,Min)	T(d)	(17-19)
Relative Humidity of previous day (Mean,Max,Min)	RH(d)	(20-22)
Wind Speed of previous day (Max)	WS(d)	(23)
Day index	d	(24)

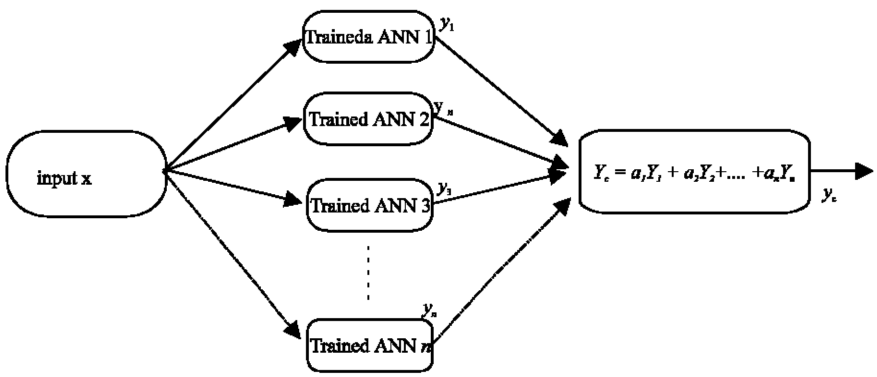

Fig. 2: Block diagram representation of combining *n*-trained neural networks

one neuron. Using past experience and heuristics, the structure and the input variables (Drezga and Rahman, 1998) are selected. The process of optimizing the network parameters is performed by the back propagation algorithm. Figure 1 shows the representation of general architecture of ANN and Table 1 gives the details of different input variables selected to the ANN.

Table 1 shows the list of selected input variables. With these input variables selection, a number of ANNs are trained with different architectures and different training parameters. Of these trained networks, based on the error measures (performance) the best networks are selected and using these networks, various combinations are developed for the combination module to improve the accuracy of prediction. The daily peak load forecasting has been applied for Chennai city with the help of these selected network structures and by using the developed combinations of ANN models. The Mean Absolute Percent Error (MAPE) and Root Mean Squared Error (RMSE) are the error measures used to analyze the results (Tomonobu *et al.*, 2002; Desouky and Kateb, 2000; Alex *et al.*, 1994; Drezga and Rahman, 1998; Tetsuro *et al.*, 2001) They are defined by (1) and (2).

$$MAPE = \left(\frac{1}{N} \sum_{i=1}^{N} \frac{|y_i - d_i|}{d_i} \right) \times 100 \qquad (1)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(y_i - d_i\right)^2} \qquad (2)$$

where is the predicted load and is the actual (desired) load for a day *i* and *N* is the total number of test data.

COMBINATIONS OF ANNs

In this segment, the Optimal Linear Combination (OLC) (Sherif and Bruce, 1995) problem is formulated for a set of *n*-trained neural networks. There are *n*-trained artificial neural networks for the given input-output relations.

x	is the input to all the neural networks,
Y_j	is the predicted output for the input x, (j = 1, 2, … n),
d	is the desired output for the given input x and
$e_j = d - y_j$	is the error of the jth neural network for the given input x,
$Y_c = a_1 Y_1 a_2$ $Y_2 + ... + a_n y_n$	is the linear combination of the outputs of n-trained neural networks for a given input x and the corresponding error for the input x is given by $e_c = d - Y_c$.

a_j is the combination weight associated with Ann's outputs (j = 1, 2 … n).

Figure 2 shows the general block diagram representation of combining the n-trained neural networks. The input x is applied to the all n-trained ANNs. The outputs are predicted and obtained from the n-trained ANNs and then given to the combination module in which the various combinations are created. This combination module follows the computational algorithm, given below and produces the combined output y_c.

The problem is to find good values for the combination weights a_j, (j= 1, 2 … n) using the optimal linear combination of the outputs of n-trained ANNs. The OLC is defined by the optimal combination weights vector that minimizes the expected loss

$$\int_s l\left(d_c\left(X^c : a\right)\right) dF_X^c, \tag{3}$$

where s is the support of $F^c_X$and l is a loss function. The input x is as an observation of a random X_c variable from a multivariate distribution function F^c_X. Although various loss functions could be followed, here the loss function is restricted to squared-error loss, $l(e_c) = (e_c)^2$. The objective is then to minimize the Mean Squared Error (MSE) of y_c given by,

$$\text{MSE}\left(y_c\left(x:a\right)\right) = E\left(e_c\left(x:a\right)^2\right) \tag{4}$$

In this study two different computational algorithms are pursued to achieve the optimal linear combination of weights for n-trained neural networks by minimizing the MSE and so that to obtain the required performance measures (1, 2) for the selected load forecasting problem.

IMPLEMENTATION OF PSO

Particle Swarm Optimization (PSO) is a stochastic global optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995 based on simulation of social behaviors of (Kennedy and Eberhart, 1995) bird flocking and fish schooling. It uses a number of particles that constitute a swarm. During flight or swim each particle adjusts its position according to its own experience and the experiences of neighbors, which includes the current velocity and position and the best previous position encountered by it and its neighbors.

In a two dimensional search space, let x and v be the particle coordinates for position and velocity respectively. The best previous position of a particle is obtained and denoted as. The index of the best particle among all the particles in the group is denoted as. The modified velocity and the position can be calculated using the following formulas:

$$\begin{aligned} v_i^{k+1} &= wv_i^k + c1rand(\ )\times\left(p_{best_i} - s_i^k\right) \\ &+c2rand(\ )\times\left(g_{best} - s_i^k\right) \end{aligned} \tag{7}$$

where

V^k_j: velocity of particle i at iteration k
w: inertia weight factor
C_1 C_2: acceleration constant
$rand(\ )$: uniform random value between 0 and 1
S^k_j: current position of particle i at iteration k
$P_{best\,j}$: P_{best} of particle i
g_{best}: P_{best} of the group

`Proper selection of weight w provides a balance between global and local explorations, thus requiring less iteration on average to find a sufficiently optimal solution. The following is the equation to obtain weight w.

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times iter \tag{8}$$

where

$w_{\max}$ is the initial weight
$w_{\min}$ is the final weight
$iter_{\max}$ is the maximum number of iterations
$iter$ is the current number of iteration,

using the above procedures, a certain velocity that gradually gets close to p_{best} and g_{best} can be determined. Equation 9 can modify the current position of ith particle.

$$s_i^{k+1} = s_i^k + v_i^{k+1} \tag{9}$$

The acceleration constants (c_1, c_2) influence the speed of each particle. Low values of c_i minimize the speed of optimization process and require large number of iterations. Large values of c_i numerically unstablize the optimization process. Hence, the acceleration constants c_1 and c_2 are often set at 10-20% according to the past experiences.

The sequences of PSO for the selected problem are:

- Initialize the parameters such as the swarm size, inertia weight factor, acceleration constant etc., generate the initial trial vectors for a_j, ($j = 1,2,...n$) at random.
- Calculate the fitness value for the each particle according to in the population pool.

- Compare the fitness value of each particle with its. The particle with the best fitness value among is denoted as g_{best}.
- Modify the velocity v of each particle according to (7).
- Change the member position s of each particle according to (9).
- If the fitness value of created off spring is better than the current p_{best} then the p_{best} value is replaced by the current value and if the p_{best} is better than the g_{best} then the best value is set to .
- The process of creating new trials and the modification of each particle will be repeated until the best objective value (minimum value of) is not obviously improved or the given number of total iterations is reached.

The PSO control parameters are chosen as:
Population size: 30
Maximum no. of iteration: 2000
w_{max} and w_{min}: 0.9 and 0.1
Acceleration constant(C_1, C_2): 2

RESULTS AND DISCUSSION

The entire research of this selected problem is carried out in AMD Sempron 1.4 GHz processor. The programs for the algorithms are coded in MATLAB 6.5 software. Initially, the neural networks with the different architecture and with different training parameters have been selected for this problem. For the case of architecture, the number of hidden neurons has been varied from the range of one neuron to 50 neurons, so that 50 different neural networks in terms of architecture are modeled and created for training. These neural networks are trained with different learning rates and it has been varied from 0.1 to 1.5 with step 0.1. Totally, 750 networks are obtained and they are trained with different architecture and with different learning rates. The number of iterations has also been varied from 500 to 20,000 and finally it is set to 5,000 for all the networks.

All these neural networks are trained for the months from August to October 2005 (three months and 92 input data sets). The data set of 1st week of November 2005 (test data) is selected to test the trained networks. It is

Table 2: Performance of different combinations of ANNs using EP and PSO

Combination modules	Combinations of networks	PSO Based combination module MAPE (%)	RMSE (MW)
Module 1	ANN 1,2	2.6479	s37.4549
Module 2	ANN 1,2,3	2.3184	32.7168
Module 3	ANN 1,2,3,4	2.1539	30.1585
Module 4	ANN 1,2,…5	2.0507	30.1229
Module 5	ANN 1,2,…6	1.9688	29.3325
Module 6	ANN 1,2,…7	1.8875	28.3728
Module 7	ANN 1,2,…8	1.8711	28.3431
Module 8	ANN 1,2,…9	1.8457	27.4553
Module 9	ANN 1,2,…10	1.7036	25.8491
Module 10	ANN 1,2,…11	1.6421	25.4068
Module 11	ANN 1,2,…12	1.5593	24.1769
Module 12	ANN 1,2,…13	1.5372	23.4918
Module 13	ANN 1,2,…14	1.5221	22.5762
Module 14	ANN 1,2,…15	1.5132	22.5410

Table 3: Best-selected networks

Networks	Topologyi/p-hid-o/p	Learning rate	1st week of November 2005 MAPE (%)	RMSE MW)
ANN 1	24-36-1	0.9	2.6133	38.2462
ANN 2	24-30-1	0.9	2.7315	37.9887
ANN 3	24-42-1	0.7	2.8352	44.8815
ANN 4	24-22-1	0.9	2.8495	47.1023
ANN 5	24-28-1	0.9	2.9547	47.6269
ANN 6	24-24-1	0.6	3.1021	49.4306
ANN 7	24-09-1	0.8	3.1146	47.0745
ANN 8	24-15-1	0.9	3.1774	48.9686
ANN 9	24-26-1	0.7	3.2215	51.8642
ANN 10	24-25-1	0.9	3.2328	54.7465
ANN 11	24-12-1	0.9	3.3127	55.3298
ANN 12	24-08-1	0.8	3.3852	57.7522
ANN 13	24-20-1	0.6	3.9812	60.1847
ANN 14	24-17-1	0.9	4.1624	61.9814
ANN 15	24-10-1	0.9	4.2926	62.7522

Iterations =5,000

Table 4: Comparision of performances

	1st week of November 2005	
Method	MAPE %	RMSE (MW)
Best Conventional ANN	2.6133	38.2462
PSOCANN (module 14)	1.5132	22.5410

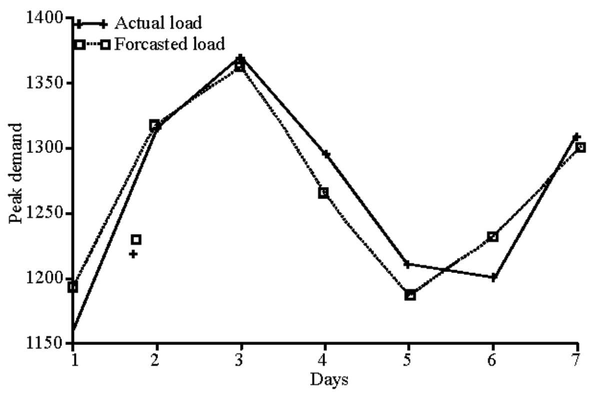

Fig. 3: Variations between actual and forecasted loads for 1st week of November 2005 using PSO based combination module 14

understood that the better neural networks are obtained for the hidden neurons that are varying from eight to 42 and for the learning rate 0.6 to 0.9. Based on the performance measures given in (1) and (2), the first fifteen ranked (top 15) neural networks are selected to develop different combinations for the combination module to obtain combined output and the performances of different combinations are studied and they are tabulated in Table 2.

It is found that when more number of networks included in the combination, the performance of the combination is also improved, that is the accuracy of load forecasting is increased. The results produced by the combination module of different combinations, using the above said two different algorithms are compared with each other and with the best conventional neural network with respect to the performance measures MAPE and RMSE.

Table 3 shows the details of selected neural networks with the tested results. From Table 3 it is understood that the conventional ANN, that is the single best-trained network with the structure of 24-36-1 and with the learning rate of 0.9 produced the best performance values of MAPE and RMSE as 2.6133 and 38.2462 MW respectively. The ANN with the structure 24-10-1 and the learning rate of 0.9 produced the least performance values among the selected 15 networks as 4.29 and 59.75 MW for the given set of test data of the 1st week of November 2005.

Table 2 and 3 gives the details of results produced by the combination module obtained with different

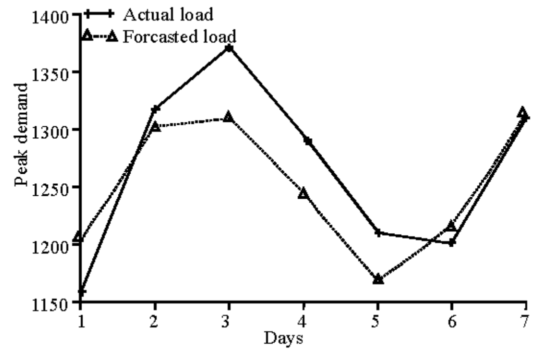

Fig. 4: Variations between actual and forecasted loads for 1st week of November 2005 using best conventional ANN with the structure 24-36-01 and with learning rate of 0.9

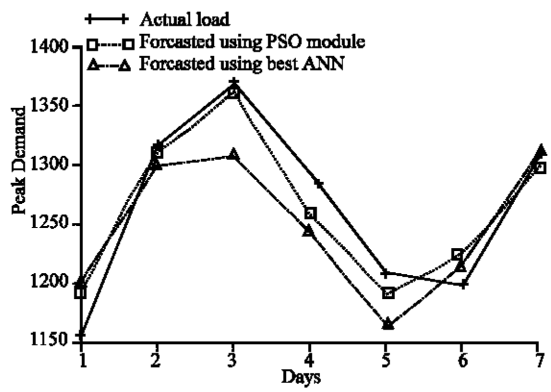

Fig. 5: Comparison between PSO based combination module 14 and best conventional ANN with the structure 24-36-1 for the forecasted loads of 1st week of November 2005

combinations, using the referenced technique and the first 15 ranked conventional ANNs. In comparison with the results in terms of performance measures, all the combinations of networks produce satisfactory performance than the best conventional ANN

Table 4 gives the details of results produced by the CANN module using the referenced technique and the best conventional ANN. In comparison with the results in terms of performance measures, the PSOCANN modules produce better performance than the best conventional ANN. And among the different cases of combinations, the module 14 displays a best performance (i.e., the error measures MAPE and RMSE are hugely reduced from the values of 2.6133% and 38.2462 MW to 1.5131% and 22.541 MW respectively) than the other combination modules and the single best trained conventional ANN for the selected problem.

Figure 3 shows the variations between the actual and forecasted loads using the PSO based combination module14, in which all the selected 15 networks are combined together to produce combined output (forecasted output). Figure 4 shows the variations between the actual and forecasted loads using the single best trained ANN. Figure 5 shows the comparison between the forecasted loads using PSO based combination module 14 and the best conventional ANN with the structure 24-36-1 and with a learning rate of 0.9.

CONCLUSION

This study proposes a week ahead daily peak load forecasting combination module using Particle Swarm Optimization. A set of neural networks are trained and some of the networks with best performance are selected for combining the outputs. PSO based approach is discussed and applied to develop the different neural network combinations for the optimal linear combination module that combines the outputs of the selected networks. The obtained results indicate that the proposed PSO based combination modules can provide power system engineer with the reason of forecasting results. The proposed combination modules using the referenced algorithm can forecast a week ahead daily peak load demand (STLF) more accurately than the single best trained network and other conventional methods.

REFERENCES

Alex, D., Papalexopoulos, Shangyou Hao and Tie-Mao Peng, 1994. An Implementation of a neural network based load forecasting model for the EMS. IEEE. Trans. Power Sys., 9: 1956-1962.

AlFuhaid, A.S., M.A. El-Sayed, and M.S. Mahmoud, 1997. Cascaded Artificial Neural Networks for Short-term load forecasting, IEEE. Trans. Power Sys., 12: 1524-1529.

Bakirtzis, A.G., J.B. Theocharis, S.J. Kiartzis and K.J. Satsios, 1995. Short term load forecasting using fuzzy neural networks, IEEE. Trans. Power Sys., 10: 1518-1524.

Dash, P.K., S. Dash, and S. Rahman, 1993. A Fuzzy Adaptive Correction Scheme for Short term load forecasting using fuzzy layered neural network, ANNPS. IEEE. Proceedings of the Second International Forum on Application of Neural Networks to Power Sys., pp: 432-437.

Drezga, I. and S. Rahman, 1998. Input variable selection for ANN based short-term load forecasting, IEEE. Trans. Power Sys., 13: 1238-1244.

Drezga, I. and S. Rahman, 1998. Short-Term load forecasting with local ANN predictors, IEEE. Trans. Power Sys., 14: 844-858.

EI Desouky, A.A. and M.M. EIKateb, 2000. Hybrid adaptive techniques for electric-load forecast using ANN and ARIMA, IEE. Proceedings, Gener. Trans. Distrib., 147: 213-217.

Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization', Proceedings of IEEE International Conference on Neural Networks, (ICNNN), Perth, Australia, pp: 1942-1948.

Lee, K.Y. and J.H. Park, 1992. Short-term load forecasting using an artificial neural network, IEEE. Trans. Power Sys., 7: 124-132.

Lu, C.N., H.T. Wu and S. Vemuri, 1993. Neural network based short term load forecasting. IEEE. Trans. Power Sys., 8: 336-342.

Papadakis, S.E., J.B. Theocharis, S.J. Kiartzis, and A.G. Bakirtzis, 1998. A novel approach to short-term load forecasting using fuzzy neural networks, IEEE. Trans. Power Sys., 13: 480-492.

Peng, T.M., N.F. Hubele and G.G. Karady, 1992. Advancement in the application of Neural Networks for Short-Term Load Forecasting. IEEE. Trans. Power Sys., 7: 250-257.

Rahman, S. and O. Hazim, 1993. A generalized knowledge-based Short-Term load forecasting technique, IEEE. Trans. Power Sys., 8: 508-514.

Sherif Hashem and Bruce Schmeise, 1995. Improving model accuracy using optimal linear combinations of trained neural networks, IEEE. Trans. Neural Networks, 6: 792-794.

Tetsuro Matsui, Tatsuya Iizaka and Yoshikazu Fukuyama, 2001. Peak load forecasting using analyzable structured neural network, Proceedings of IEEE, PES 2001, Columbus, Ohio USA, Winter Meeting, pp: 1-6.

Tomonobu Senjyu, Hitoshi Takara, Katsumi Uezato and Toshihisa Funabashi, 2002. One-Hour-Ahead load forecasting using fuzzy neural network, IEEE. Trans. Power Sys., 17: 113-118.