Comparative Analysis of Short-Term Load Forecasting Using Kalman Filter and NARX model

[1]Mandar Dixit, [2]Pranita Chavan
Electrical Engineering Department, ViMEET, Khalapur
Pillai HOC COET, Rasayani
[1]mmdixit@vishwaniketan.edu.in

Abstract— Over the years, different methods have been developed and improved upon to forecast load demands. These methods of load forecasting are classified into two categories: classical approaches and artificial intelligence (AI) based techniques. Classical approaches are based on various statistical modelling methods. These approaches forecast future values of the load by using a mathematical combination of previous values of the load and other variable such as weather data. This includes the use of regression exponential smoothing, Box-Jenkins, autoregressive integrated moving average (ARIMA) models and Kalman filters. Short-term load forecasting (STLF) aims towards prediction of electricity loads for a period of minutes, hours, days or weeks. Accurate load forecasting will lead to appropriate scheduling and planning with optimize energy cost. The time dependent factors, random factors, and weather factors have different effects on load forecasting patterns. The papers give comparative analysis based on Kalman Filter and NARX neural network model. The models are verified for summer and winter data.

Keywords— ***Short-term load forecasting, Kalman filter, NARX Model***

I. Introduction

In recent years, with the opening of electricity markets, electrical power system load forecasting plays an important role for electrical power operation. Accurate load forecast will lead to appropriate operation and planning for the power system, thus achieving a lower operating cost and higher reliability of electricity supply. Short-term load forecasting (STLF) of electric power is important in operation scheduling, economic dispatch, unit commitment, energy transactions and fuel purchasing [1, 2]. Short-term load forecasting aims towards prediction of electricity loads for a period of minutes, hours, days or weeks. The quality of short-term load forecasts with lead time ranging from one hour to several days ahead has significant impact on the efficiency of any power utility [3]. In the developing countries like India the power sector is often unable to meet peak demands. It seems essential that the scheduling of generation is to be planned carefully since one has to work within stringent limits. Hence, suitable strategies are necessary for generation control and load management.

II. Forecasting Models

Owing to the importance of STLF, research in this area in the last years has resulted in the development of numerous forecasting methods. These methods are mainly classified into two categories: classical approaches and artificial intelligence (AI) based techniques. Classical approaches are based on various statistical modeling methods. These approaches forecast future values of the load by using a mathematical combination of previous values of the load and other variable such as weather data. Classical STLF approaches use regression exponential smoothing, Box-Jenkins, autoregressive integrated moving average (ARIMA) models and Kalman filters. Recently several research groups have studied the use of artificial neural networks (ANNs) models and Fuzzy neural networks (FNNs) models for load forecasting [8]. With the development of AI in recent years, people become able to forecast using FNN and ANN with the back propagation method. Although the back propagation method has solved a number of practical problems, its poor convergence and Methoology speed can somewhat deter engineers. Meanwhile, a conventional ANN model sometimes can suffer from a sub-optimization problem [9, 10].

In this paper, a STLF procedure based on a Kalman filtering, ARIMA and ANN based models are illustrated in detail. The model is applied to different load conditions of hourly load shape. Corrections in the prediction, especially for the peak hours can be made by applying a simple method of error feedback. The forecasting results obtained are quite promising, thus demonstrating the good potentials on such a kind of electric load. Weather is one of the principal causes of load variations as it affects domestic load, public lighting, commercial loads etc. therefore, it is essential to choose relevant weather variables and model, their influence on power consumption. Principal weather variables found to affect the power consumption include temperature, cloud cover, visibility and precipitation. Hence the comparative analysis is performed based on summer and winter data.

III. KALMAN FILTER MODEL

Kalman filtering, also known as linear quadratic estimation (LQE), is an d that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone, by estimating a joint probability distribution over the variables for each timeframe.

The algorithm works in a two-step process. In the prediction step, the Kalman filter produces estimates of the current state variables, along with their uncertainties. Once the outcome of the next measurement (necessarily corrupted with some amount of error, including random noise) is observed, these estimates are updated using a weighted average, with more weight being given to estimates with higher certainty. The algorithm is recursive. It can run in real time, using only the present input measurements and the previously calculated state and its uncertainty matrix; no additional past information is required.

In order to use the Kalman filter to estimate the internal state of a process given only a sequence of noisy observations, one must model the process in accordance with the framework of the Kalman filter. This means specifying the following matrices:

- F_k, the state-transition model;
- H_k, the observation model;
- Q_k, the covariance of the process noise;
- R_k, the covariance of the observation noise;
- and sometimes B_k, the control-input model, for each time-step, k, as described below.

The Kalman filter model assumes the true state at time k is evolved from the state at $(k - 1)$ according to

$$x_k = F_k x_{k-1} + B_k u_k + w_k \quad (1)$$

where

- F_k is the state transition model which is applied to the previous state x_{k-1};
- B_k is the control-input model which is applied to the control vector u_k;
- w_k is the process noise which is assumed to be drawn from a zero mean multivariate normal distribution, with covariance, Q_k: $w_k \sim (0, Q_k)$

At time k an observation (or measurement) z_k of the true state x_k is made according to

$$z_k = H_k x_k + v_k \quad (2)$$

where

- H_k is the observation model which maps the true state space into the observed space and
- v_k is the observation noise which is assumed to be zero mean Gaussian white noise with covariance R_k: : $v_k \sim (0, R_k)$

The initial state, and the noise vectors at each step $\{x_0, w_1, \ldots, w_k, v_1 \ldots v_k\}$ are all assumed to be mutually independent.

ARIMA (p,q,d); Auto Regressive Integrated Moving Average produced by Elrazaz and Mazi (1989) is an extension of ARMA. It utilizes a difference operator to transform the non-stationary time series into a stationary one. While p is the number of auto-regressive terms and q is the number of lagged forecast error, d is the number of non-seasonal differences. If differencing is eliminated (i.e. d=0), then ARIMA model transform into an ARMA model. Consider a time series x_t, and then the first order differencing is defined as:

$$x_t' = x_t - x_{t-1} \quad (3)$$

We can use L to express differencing:

$$(1 - \varphi_1 L - \varphi_2 L^2 - \cdots - \varphi_p L^p)(1 - L)^d x_t \quad (4)$$

Thus ARIMA (p,q,d) is defined as:

$$= c + (1 - \psi_1 L - \psi_2 L^2 - \cdots - \psi_q L^q)\varepsilon_t \quad (5)$$

$$\emptyset(L)(1 - L)^d x_t = c + \psi(L)\varepsilon_t \quad (6)$$

IV. ANN NARX MODEL

The NARX is a recurrent time delay network (TDNN) with a delay line as shown in Figure 3 on the inputs and a feedback connection from the output to the inputs. A delay line is a lag of time in the inputs. The inputs like in the statistical VARMA model are a mixture of past values of the same time series, and past values of another independent time series. A NARX as shown in figure 3 network can be mathematically represented as

$$y_t = f\left(y_{t-1}, y_{t-2}, \dots, y_{t-p}, X_t, X_{t-1}, X_{t-2}, \dots, X_{t-q}; w\right) + \varepsilon_t \qquad (7)$$

Where x_t *and* y_t denote, respectively, the input and the output of the model at discrete time t. The parameters p and q are memory delays, with $q < p$. The function f is a non-linear function of the input and output of the model. The predicted output y_t is regressed on the input values (exogenous) x_{t-k} and the output value y_{t-k}. Figure 1 shows the architecture of a NARX network with two hidden layer. This can be generalized to multiple inputs (N) and outputs (M).

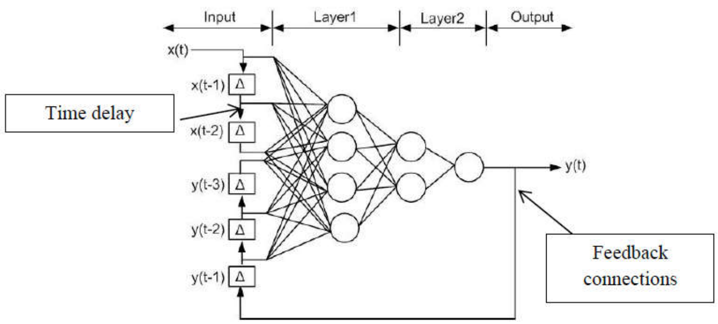

Fig 1: Typical architecture of NARX network

For learning purposes, a dynamic back-propagation algorithm is required to compute the gradients, which is more computationally intensive than static back-propagation and takes more time. In addition, the error surfaces for dynamic networks can be more complex than those for static networks. Training is more likely to be trapped in local minima. In general, in function approximation problems, for networks that contain up to a few hundred weights, will have the fastest convergence. This advantage is especially noticeable if very accurate training is required. However, as the number of weights in the network increases, the advantage of this algorithm decreases.

The selected training method for our work uses the advantage of availability at the training time of the true real output set. It is possible to use the true output instead of the estimated output to train the network which has the feedback connections decoupled. The decoupled network has a common feed forward architecture which can be trained with classical static back-propagation algorithm. In addition, during training, the inputs to the feed forward network are just the real/true ones – not estimated ones, and the training process will be more accurate.

V. METHOD OF IMPLEMENTATION

Following steps are implemented for Kalman Filtering Prediction Model:

- Run Kalman filter over the residual sequence with model in order to produce the filtering estimate of AR weight vectors. Predict the weights over the missing parts.
- Run Kalman smoother over the Kalman filter estimation result above, which results in smoothed MAP estimate of the weight time series including the missing parts.
- Run Kalman filter over the residual sequence with model in order to produce filtering estimate of the short term periodicity. The periodicity is also predicted over the missing parts.
- Run Kalman smoother over the Kalman filter estimation result above, which results in smoothed MAP estimate of the periodicity time series including the missing parts.

Following steps are used for implementation of model:

- The old data with its time stamping is firstly pre-processed and prepared in order to get processed in the NARX neural model.

- All the pre-processed data is now fed to the NARX neural network with 10 hidden layers and Levenberg-Marquardt as the training function with MSE as the performance evaluating parameter.
- After the training the regression of around 0.97 is achieved of the training set and 0.95 of the overall system
- For the prediction of the data after the training the closed loop system is prepared in the next stage and then a value ahead of the previous value is predicted every time the simulation executed model shown in Fig 2

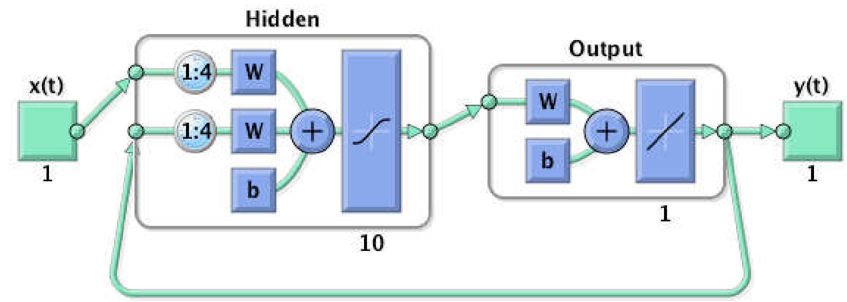

Fig. 2 NARX Predictor model

VI. ANALYSIS OF RESULTS

For analysis of both methods following data of a 33kV feeder in Khalapur Section of Maharashtra was considered. The data has been collected for a period of six month of April 2015 and December 2015

TABLE I PERFORMANCE ERROR

Algorithm	Performance error	
	Winter	Summer
Kalman	1.726	2.57
Narx	0.059	0.066

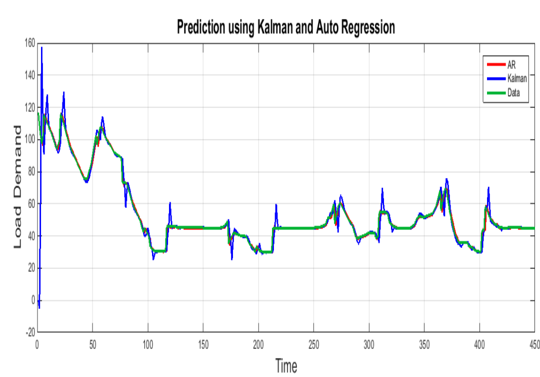

Fig. 3 Prediction Graph with Kalman Filter for summer and winter load

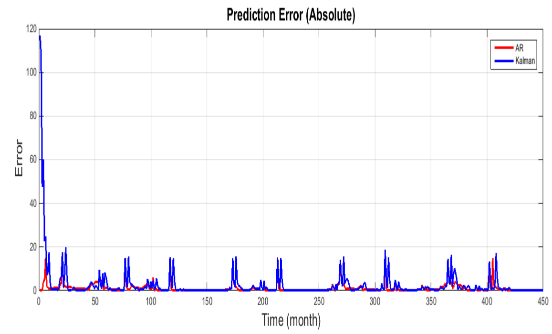

Fig. 4 Prediction error with Kalman Filter for summer and winter load

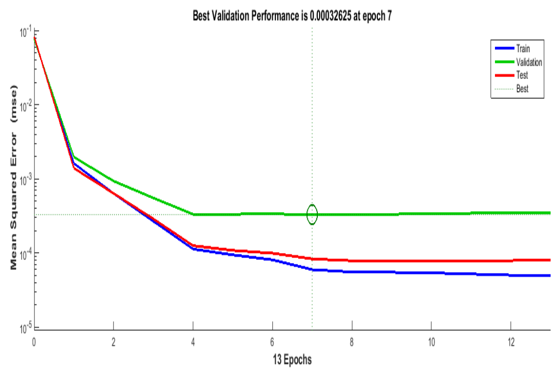

Fig.5 NARX neural network performance graph summer load

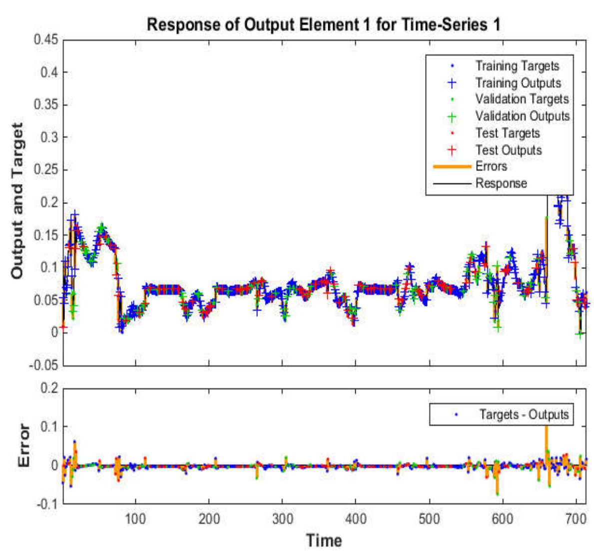

Fig.6 NARX neural network time response graph for summer load

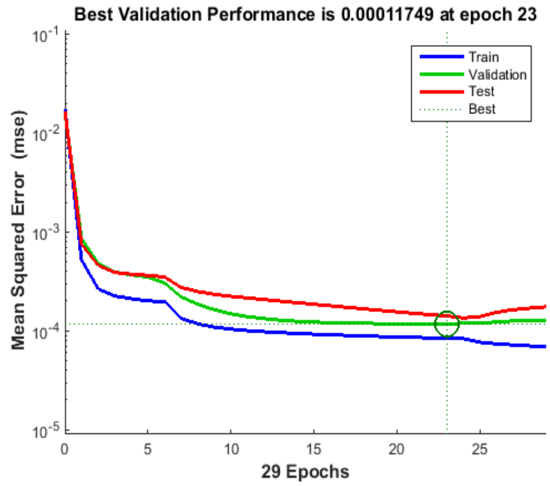

Fig.7 NARX neural network performance graph winter load

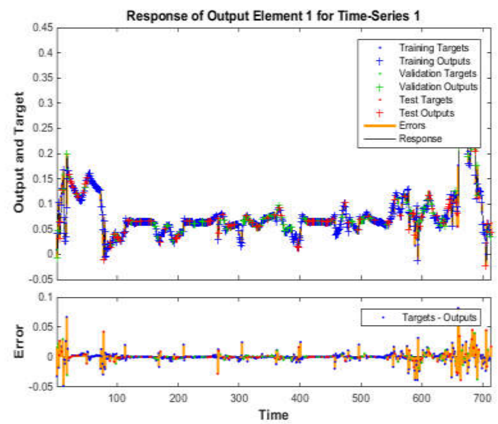

Fig. 8 NARX neural network time response graph for winter load

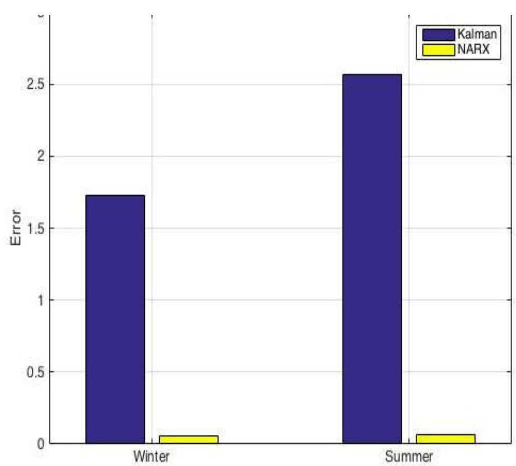

Fig. 9 Comparison graph of summer and winter data

VII. CONCLUSION

The paper has presented an application of a Kalman predictor and NRX model to the Short-Term Forecasting of the load shape of 33kV Feeder data. The basic model, main procedure and designing features are illustrated in detail. The obtained results show that the proposed architecture can be promising for the achieved forecasting accuracy. Furthermore, the Kalman filtering based approach allows very interesting possibilities in terms of integration and modification of both parameters and input data. Further improvements are expected from an extensive application to electric load data of further years jointly with the integration of data base with meteorological data.

ACKNOWLEDGMENT

We would like to thank MSEDCL for their valuable support.

REFERENCES

[1] M. Farahat and M. Talaat "A New Approach for Short-Term Load Forecasting Using Curve Fitting Prediction Optimized by Genetic Algorithms" IEEE, 14th International Middle East Power Systems Conference MEPCON'10, pp. 106-110, Egypt, 2010.

[2] Khotanzad, E. Zhou, and H. Elragal, "A Neuro-Fuzzy Approach to Short-Term Load Forecasting in a Price-Sensitive Environment", IEEE Transactions on Power Systems, Vol. 17, No. 4, (2002), pp. 1273-1282.

[3] R. Sadowruk and E. Barbosa, "Short-Term Forecasting of Industrial Electricity Consumption in Brazil", Int. J. Forecast, Vol. 18, (1999), pp. 215–224.

[4] M. A. Farahat, Long – Term Industrial Load Forecasting and Planning Using Neural Networks Technique and Fuzzy Inference Method", 39th International Universities Power Engineering Conference UPEC'2004, University of West of England, Bristol, Vol. 1, 6 – 8 September (2004), pp. 368 - 372.

[5] G. Gross and F. D. Galiana, "Short term load forecasting," Proc. IEEE, vol. 75, no. 12, pp. 1558-1573, Dec. 1987.

[6] I. Moghram, S. Rahman, "Analysis and evaluation of five short-term load forecasting techniques," IEEE Trans. on Power Systems, Vol. 4, No. 4, October 1989.

[7] G. Liao and T. Tsao, "Application of a Fuzzy Neural Network Combined With a Chaos Genetic Algorithm and Simulated Annealing to Short-Term Load Forecasting", IEEE Transactions on Evolutionary Computation, Vol. 10, No. 3, (2006), pp. 330-340.

[8] R. C. Bansal, "Bibliography on the Fuzzy Set Theory Applications in Power Systems (1994–2001)", IEEE Transactions on Power Systems, Vol. 18, No. 4, (2003), pp. 1291-1299.

[9] H. Mao, X. Zeng, G. Leng, Y. Zhai, and J. A. Keane, "Short-Term and Midterm Load Forecasting Using a Bilevel Optimization Model", IEEE Transactions on Power Systems, Vol. 24, No. 2, (2009), pp. 1080-1090.

[10] S. H. Ling, F. Leung, H. K. Lam, and P. Tam, "Short-Term Electric Load Forecasting Based on a Neural Fuzzy Network", IEEE Transactions on Industrial Electronics, Vol. 50, No. 6, (2003), pp. 1305-1316.