Interacting Multiple Model Approach for Very Short-Term Load Forecasting and Confidence Interval Estimation

Che Guan, *Student Member*, Peter B. Luh, *Fellow, IEEE*,
Laurent D. Michel and Y. Bar-Shalom, *Fellow, IEEE*

Department of Electrical & Computer Engineering
University of Connecticut
Storrs, CT 06269, USA

{che.guan, luh, ldm & ybs}@ engr.uconn.edu

Peter B. Friedland

ISO New England
Holyoke, MA 01040, USA

pfriedland@iso-ne.com

***Abstract* - Very short-term load forecasting predicts load over one hour into the future in five minute steps and performs the moving forecast every five minutes. To quantify prediction accuracy, the confidence interval is estimated in real time. This is essential for area generation control and resource dispatch, and helps operators make good decisions. However, an effective prediction is difficult in terms of complicated dynamic load features. This paper develops an interacting multiple model approach using Kalman trained neural networks. Because the hourly input-output functions of the neural network can be nearly linear and nonlinear, it is hard to evaluate which one plays an important role at each step, and thus hard to accurately capture the dynamic load features. The key idea is to use two models to process the same load in parallel with dynamic mixing weights for the models. A neural network trained by an extended Kalman filter is used to capture load features with nearly linear hourly input-output function and a neural network trained by an unscented Kalman filter is used for the nonlinear one. The overall estimate is a weighted sum of the model-conditioned estimates. Numerical testing demonstrates the significant value of the presented method for load forecasting with good confidence interval estimation.**

Index Terms – Confidence interval estimation, Extended Kalman filter, Unscented Kalman filter, Interacting Multiple Models, Very short-term load forecasting.

I. INTRODUCTION

Very short-term load forecasting (VSTLF) predicts the load over one hour into the future in five minute steps and performs a moving forecast every five minutes. In order to quantify the reliability of the prediction, a confidence interval (CI) is estimated in real time. Accurate VSTLF will improve automatic generation control performance and ensure revenue adequacy within the Independent System Operator (ISO [*]) multi-settlement market by reducing the ex-ante dispatch. With the help of a small CI, the operator can make low-risk decisions. However, the hourly input-output model function can be nearly linear or nonlinear. It is hard to determine which one should be relied upon with a high priority at each step, and thus hard to perform effective forecasting.

Among the variety of prediction methods, neural networks have been widely used because of their strong learning capacity. Multilayer preceptron network with back propagation as its learning algorithm is one of the popular networks. But the general network cannot estimate a good CI due to the restriction of the model itself. Also, the back propagation operates entirely on the basis of first-order information (Haykin and Kailath, 2002). To produce a good CI and accelerate the convergence, the neural network training was formulated around the extended Kalman filter (EKFNN) by treating the weight as the state (Singhal, and Wu, 1989; Puskorius and Feldkamp, 1991; Zhang and Luh, 2005). This works well for nearly linear input-output functions. On the other hand, if the daily input-output function of the network is highly nonlinear, the neural network trained by unscented Kalman filter (UKFNN) will give a superior performance. Then, the CI is obtained from the standard deviation corresponding to the estimated variance. However, the input-output function is dynamically changing between nearly linear and nonlinear relations each day, one model cannot evaluate which function plays the most important role at each step, and thus cannot execute an efficient way to process it.

A synergistic combination of multilevel wavelet decomposition and networks with data pre-filling had been presented for VSTLF (Guan et al., 2009a). Since this method cannot estimate a good CI for the predicted load due to the restriction of the model itself, the framework is extended to multilevel wavelet neural networks trained by hybrid Kalman algorithms (Guan et al,. 2009b) to address the issue.

In this paper, an interacting multiple model (IMM) using Kalman algorithms trained neural networks is developed for VSTLF and CI estimation. Because the hourly input-output function of the neural network dynamically changes between nearly linear and nonlinear relations, it is hard to evaluate which is the important one at each step. To capture the dynamic load features, the method of interacting multiple neural networks trained by Kalman algorithms is developed to predict the load in parallel with dynamic mixing weights for the running models. The IMM-EKFNN is used to capture the load feature with nearly linear hourly input-output function and IMM-UKFNN for the nonlinear one. They process the same load series but with separate configurations for load features. An updated mixing probability is calculated to mix the models each time. The overall estimate is a weighted sum of model-conditioned estimates on predictions. Numerical testing results for a simple example and load forecast of New England in Section III demonstrate the value of the method for VSTLF and CI estimation.

[*] This work was supported by a grant from ISO New England.

II. Interacting Multiple Neural Networks Trained by Kalman Algorithm with Missing Load Filtering

The method of IMM is presented here for VSTLF over one hour into the future in five minute steps and moving forecast every five minutes. This paper considers input preparation and IMM using EKFNN and UKFNN. The whole process is depicted in Figure 1. Subsection III-A briefly describes the inputs to the network. Subsection III-B presents the load prediction and interval estimation based on IMM - EKFNN and IMM-UKFNN in parallel. The final prediction is a weighted sum of the model-conditioned estimates.

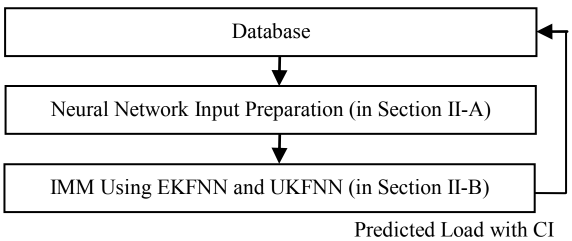


Figure 1. Overall structure of the method

A. *Input Preparation to Neural Network*

To prepare a good input, the neural network uses the relative increment (Charytoniuk and Chen, 2000) in previous hour's load l_t as denoted by (1), which is then normalized by (2) in order to produce the element u_t for the input vector u(t) = $\{u_t, u_{t+1}, \ldots\}^T$

$$l_t^{RI} = \left(l_t - l_{t-1}\right)/l_{t-1} \tag{1}$$

$$u_t = \left(l_t^{RI} - l_{\min}^{RI}\right)/\left(l_{\max}^{RI} - l_{\min}^{RI}\right). \tag{2}$$

Based on data analysis, it is hard to reduce the prediction error of peak and valley load, where the load changes very fast. If we assume that the load at peak and valley appear at the same hour for the target day and the day of last week with same weekday index, the idea is to find the hour index of latter and use it as the additional input to mark the former. This input index will help network to emphasize the load feature at peak and valley hour. According to the observation of actual daily load, there are two peaks and two valleys in the daily load pattern. Thus, four indices are used to represent them. If the predicted load is around peak or valley hour, the relevant index is assigned by one and the other three by zero; if the predicted load does not happen at peak or valley hour, four inputs are assigned by zeros. Besides, other inputs to the network include the hour and weekday indices. All the inputs as a whole are depicted in Figure 2.

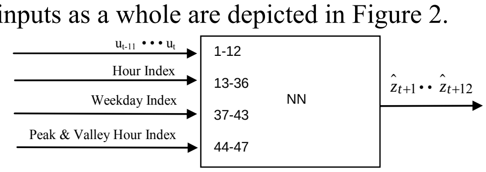


Figure 2. Inputs and outputs of the neural network

B. *IMM Using EKFNN and UKFNN*

In order to produce an accurate confidence interval for VSTLF, the Kalman algorithm replaces back propagation learning and is used to train neural networks by treating the neural network weight and desired output as the state and observation. Data analysis depicted in Figure 3 shows that most hourly input-output network functions are nearly linear related. But some are nonlinear related, especially around peak and valley hours. Through linearization, EKF has been widely adopted for state estimation of nearly linear system. On the other hand, UKFNN shows a superior performance if the function is nonlinear related. Because the peak and valley hours often change, it is hard to evaluate which type of input-output function dominates VSTLF in a day. Therefore, it is not able to evaluate which load feature should be given more focus at each step. In order to fully capture dynamic load features, the key idea is to apply IMM running EKFNN and UKFNN in parallel to form an optimal weighted sum of the prediction of two networks via undergoing a soft switching according to the latest updated mode probabilities. The IMM forms an optimal weighted sum of the output of two filters.

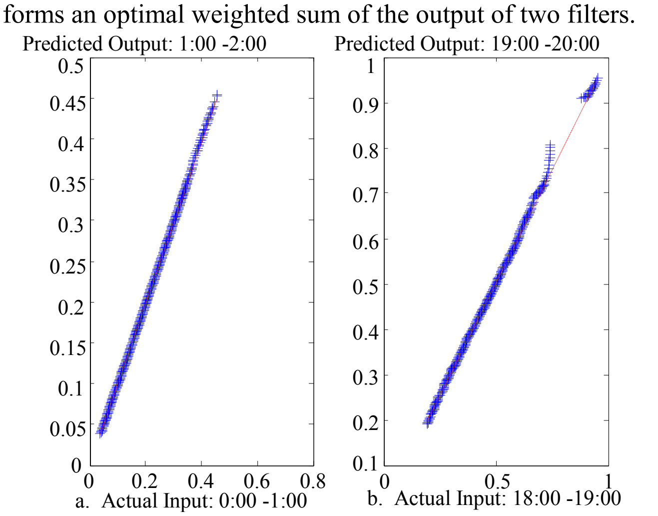
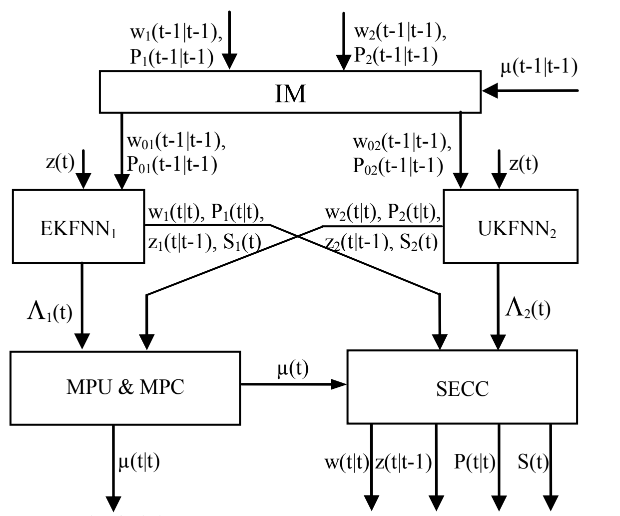


Figure 3. The dynamic features of the hourly input-output functions: (a) the input-output is nearly linear related; (b) the input-output is nonlinear related



IM: Interaction/Mixing

EKFNN: Extended Kalman Filter Trained Neural Network

UKFNN: Unscented Kalman Filter Trained Neural Network

SECC: State Estimate and Covariance Combination

MPU&MPC: Mode Probability Update and Mixing Probability Calculation

Figure 4. Overall structure of IMM using EKFNN and UKFNN

In our method, the IMM runs two Kalman filters in parallel, each using a different configuration for the same load. If we denote mixing probability by μ, likelihood function by Λ, and the filter state, measurement and their covariance matrices by w, z, P and S with superscript 1 and 2 indentifying each model, the structure is depicted in Figure 4.

Data analysis shows that most of hourly input-output neural network functions are nearly linear. Through linearization, EKF replaces the back propagation learning of the network and trains it by treating its weight as state and output as the measurement in system with dynamics. The formulation of training network via EKF (Zhang and Luh, 2005) is described via state and measurement functions by (3) and (4)

$$w(t+1)=w(t)+\varepsilon(\mathrm{t}) \tag{3}$$

$$z(t)=\mathrm{h}\left(u(t-1), w(t)\right)+v(\mathrm{t}) . \tag{4}$$

where EKF updates network weights denoted by w(t) using a set of input-output observations {u(t), z(t) t=1,…,T}, where T is the number of samples, u(t) and z(t) are input and output neural network. h(•) is the input-output network function, and ε(t) and ν(t) are process and observation noises. Key steps for standard EKF derivation (Bar-Shalom, Li and Kirubarajan, 2001) are summarized from (5)-(11)

$$\widehat{w}(t \mid t-1)=\widehat{w}_{01}(t-1 \mid t-1) \tag{5}$$

$$P(t \mid t-1)=P_{01}(t-1 \mid t-1)+Q(t-1) \tag{6}$$

$$S_1(t)=H_1(t) \cdot P(t \mid t-1) \cdot H_1(t)^T+R(t) \tag{7}$$

$$K(t)=P(t \mid t-1) H_1(t)^T S_1(t)^{-1} \tag{8}$$

$$\widehat{w}_1(t \mid t)=\widehat{w}(t \mid t-1)+K(t) \cdot\left(z(t)-\hat{z}_1(t \mid t-1)\right) \tag{9}$$

$$\hat{z}_1(t \mid t-1)=\mathrm{h}\left(u(t), \widehat{w}(t \mid t-1)\right)+v(\mathrm{t}) \tag{10}$$

$$P_1(t \mid t)=P(t \mid t-1)-K(t) S_1(t) K(t)^T$$

$$\text{where} \quad H_1(t)=\left.\left(\frac{\partial h(u, w)}{\partial w}\right)\right|_{\substack{u=u(t-1) \\ w=\widehat{w}(t \mid t-1)}} . \tag{11}$$

In the above, H is an observation function, Q and R are covariance matrices of process and measurement noises, S is the measurement covariance matrix, K is the gain, and P is the weight covariance and updated based on Bayesian formula.

When the Jacobian function is highly nonlinear, the EKF can give poor performance because the mean and covariance are propagated through linearization of the underlying non-linear model. UKF uses a sampling technique known as the unscented transform to pick a minimal set of sample points around the mean. These sigma points are then propagated through the non-linear function, from which mean and covariance of estimate are recovered. The result is a filter which more accurately captures true mean and covariance. It also avoids the need to calculate Jacobian function. With same formulation for system with dynamics, key steps of UKF (Julier, Uhlmann and Durrant-Whyte, 1995) are summarized:

$$\widehat{w}(t \mid t-1)=\sum_{i=0}^{2N} W_s^i \cdot \chi^i(t \mid t-1) \tag{12}$$

$$P(t \mid t-1)=$$

$$\sum_{i=0}^{2N} W_c^i\left[\chi^i(t \mid t-1)-\widehat{w}(t \mid t-1)\right]\left[\chi^i(t \mid t-1)-\widehat{w}(t \mid t-1)\right]^T \tag{13}$$

$$\hat{z}_2(t \mid t-1)=\sum_{i=0}^{2N} W_s^i \cdot \gamma^i(t) \tag{14}$$

$$S_2(t)=\sum_{i=0}^{2N} W_c^i\left[\gamma^i(t)-\hat{z}(t \mid t-1)\right] \cdot\left[\gamma^i(t)-\hat{z}(t \mid t-1)\right]^T \tag{15}$$

$$K(t)=P_{w(t) z(t)} \cdot S_2(t)^{-1} \tag{16}$$

$$\widehat{w}_2(t \mid t)=\widehat{w}(t \mid t-1)+K(t) \cdot\left(z(t)-\hat{z}_2(t \mid t-1)\right) \tag{17}$$

$$P_2(t \mid t)=P(t \mid t-1)-K(t) S_2(t) K(t)^T . \tag{18}$$

where $W^0_c = W^0_s = \lambda/(N+\lambda)+(1-\alpha^2+\beta)$, $W^i_c = W^i_s = .5/(N-\lambda)$, λ is a scaling parameter, χ and γ are sigma points, and γ are projected through the input-output network function, α determines the spread of the sigma points around $\bar{w}$, β is used to incorporate prior knowledge of the distribution of w (Ilyas et al., 2008).

Given EKFNN and UKFNN, the dynamic weights are generated to mix two models. First, the mixing probabilities $\mu_{i|j}$(t-1|t-1) produced by (19) is used to mix with the estimate w_j(t-1|t-1) and P_j(t-1|t-1) from model M_j (M_1 is EKFNN and M_2 is UKFNN) so that w_{0j}(t-1|t-1) and p_{0j}(t-1|t-1) are calculated by (20)-(21)

$$\mu_{i|j}(t-1 \mid t-1)=P\left\{M_i(t-1) \mid M_j(t), Z^{t-1}\right\}$$

$$=\frac{1}{\bar{c}_j} \cdot P\left\{M_j(t) \mid M_i(t-1), Z^{t-1}\right\} \cdot P\left\{M_i(t-1) \mid Z^{t-1}\right\}$$

$$\text{where} \quad \bar{c}_j=\sum_{i=1}^{2} p_{ij} \cdot \mu_i(t-1) \quad j=1,2 \tag{19}$$

$$\widehat{w}_{0j}(t-1 \mid t-1)=\sum_{i=1}^{2} \widehat{w}_i(t-1 \mid t-1) \mu_{i|j}(t-1 \mid t-1), \ j=1,2 \tag{20}$$

$$P_{0j}(t-1 \mid t-1)=$$

$$\sum_{i=1}^{2} \mu_{i|j}(t-1 \mid t-1) \cdot\left\{\begin{array}{l} P_i(t-1 \mid t-1)+ \\ \left[\widehat{w}_i(t-1 \mid t-1)-\widehat{w}_{0j}(t-1 \mid t-1)\right] \\ \cdot\left[\widehat{w}_i(t-1 \mid t-1)-\widehat{w}_{0j}(t-1 \mid t-1)\right]^T \end{array}\right\} . \tag{21}$$

Because most hourly input-output network functions are nearly linear, IMM-EKFNN is given with a high transition probability, which results in a small probability assigned for IMM-UKFNN. Then, the generated outputs w_{0j}(t-1|t-1) and p_{0j}(t-1|t-1) are used as inputs to the filter matched to M_j(t) so that the likelihood function Λ_j is produced by (22),

$$\Lambda_j(t)= N\left\{\begin{array}{l} z(t) ; \hat{z}^j[t \mid t-1 ; \\ \widehat{w}_{0j}(t-1 \mid t-1), S^j\left(t ; P^{0j}(t-1 \mid t-1)\right)] \end{array}\right\} . \tag{22}$$

Finally, mode probability updates $\mu_j(t)$ denoted by (23) together with posterior state $w_j(t|t)$, covariance $P_j(t|t)$, prediction $z_j(t|t-1)$ and innovation covariance $S_j(t)$ will be used for the combination of model-conditioned estimates and covariances $w(t|t)$, $z(t|t-1)$, $P(t|t)$ and $S(t)$ by (24)-(27)

$$\mu_j = 1/c \cdot \Lambda_j(t) \cdot \bar{c}_j$$
$$where \quad c = \sum_{i=1}^{2} \Lambda_j(t) \cdot \bar{c}_j \quad j = 1,2 \tag{23}$$

$$\hat{w}(t|t) = \sum_{j=1}^{2} \hat{w}^j(t|t) \cdot \mu_j(t) \tag{24}$$

$$P(t|t) = \sum_{j=1}^{2} \mu_j(t) \left\{ \begin{array}{l} P^j(t|t) + \\ \left[\hat{w}^j(t|t) - \hat{w}(t|t)\right] \cdot \left[\hat{w}^j(t|t) - \hat{w}(t|t)\right]^T \end{array} \right\} \tag{25}$$

$$\hat{z}(t|t-1) = \sum_{j=1}^{2} \mu_j(t) \cdot \hat{z}^j(t|t-1) \tag{26}$$

$$S(t) = \sum_{j=1}^{2} \mu_j(t) \cdot S^j(t)\,. \tag{27}$$

Among four outputs, $z(t|t-1)$ and $S(t)$ will be further used to produce the forecasting load and estimated CI. This derivation is straightforward by the standard IMM (Bar-shalom, Li and Kirubarajan, 2001).

Because the input is the normalized RI in load, the output represents predicted relative increment and has to be taken inverse transform by (28)-(29)

$$\hat{z}'_t = \hat{z}_t \cdot \left(l_{\max}^{RI} - l_{\min}^{RI}\right) + l_{\min}^{RI} \tag{28}$$

$$\hat{l}_t = \left(\hat{z}'_t + 1\right) \cdot l_{t-1}, \quad \hat{l}_{t+1} = \left(\hat{z}'_{t+1} + 1\right) \cdot \hat{l}_t \cdots\,. \tag{29}$$

The relative increment on input load is a nonlinear transformation, the further derived equation for the estimated standard deviation (Guan et al., 2009b) is used to estimate CI as written by (30)

$$\hat{\sigma}^2_{t+k} = \sum_{j=0}^{k} \left\{ \left(1 + \sum_{i=0}^{k} \hat{z}'^2_{t+i} - \hat{z}'^2_{t+j}\right) \cdot \sigma'^2_{t+j} \right\} \cdot l^2_{t-1},\ k = 0, \cdots, n_z - 1$$
$$= \sum_{j=0}^{k} \left\{ \left(1 + \sum_{i=0}^{k} \left(a \cdot \hat{z}_{t+i} + b\right)^2 - \left(a \cdot \hat{z}_{t+j} + b\right)^2\right) \cdot a^2 \cdot \sigma^2_{t+j} \right\} \cdot l^2_{t-1}$$
$$where \quad a = \left(l_{\max}^{RI} - l_{\min}^{RI}\right), b = l_{\min}^{RI}\,. \tag{30}$$

III. Numerical Testing Results

The method has been implemented in MATLAB on a personal computer with Pentium Dual Core 2.20GHz CPU and 2-G memory. Two examples are presented below. Example 1 uses a classroom-type problem to examine the neural networks trained by interacting multiple models. Example 2 predicts New England 2008 load and demonstrates benefits of the IMM using EKFNN and UKFNN for capturing the dynamic load, and shows the accuracy of load forecasting with good estimated CIs.

Example 1. Consider the following signal: $y(t) = 100 \cdot sin(20 \cdot t) + 15 \cdot sin(280 \cdot t)$. The process and measure noises are normally distributed with mean of $1\text{x}10^{-3}$ and variance 10^{-6}. This constructed signal has a strong resemblance to the actual load in terms of the relative amplitude and trend. The objective is to test through $y(t)$ for $t \in [1457, 1458, \ldots, 1478]$, and forecast 12 points into the future by using latest 12 points as inputs. Figure 5 shows that the network input-output function can be nearly linear or nonlinear related each time. Thus the presented method will be suitable to address the problem. Furthermore, different ways of using Kalman trained networks with single UKFNN, single EKFNN, two IMM-EKFNNs, and IMM-UKFNN and IMM-EKFNN are compared and show that our method gives the best forecasting with good estimated CIs as shown in last columns of Table III to Table V. Results in Figure 6 shows that the prediction follows the actual closely with good estimated CIs.

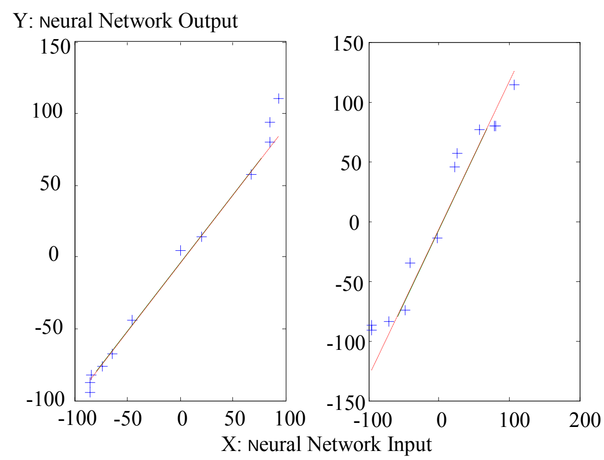


Figure 5. The input-output functions in classroom problem: the input-output is nearly linear related (left) and the input-output is nonlinear related (right)

TABLE III
Mean Absolute Error (MAE) for Classroom Problem

No.	UKFNN	EKFNN	$\text{IMM-EKFNN}_{1,2}$	**IMM-EKFNN_1 IMM UKFNN_2**
1	4.6904	2.1504	2.2043	**1.9634**
2	2.9942	2.9099	4.7814	**2.4364**
3	3.0909	5.0760	2.9050	**1.4125**
4	4.7700	2.3344	3.0229	**1.8074**
5	2.4628	8.3402	5.3613	**1.6564**
6	3.3421	2.8483	2.0101	**1.5230**
7	3.7846	2.8756	3.3878	**2.5002**
8	3.0989	5.3681	4.7074	**2.2066**
9	4.5074	5.3087	2.1120	**1.9198**
10	3.1416	2.8824	4.3244	**2.4711**
11	3.3874	3.4634	3.6324	**1.4301**
12	4.6498	2.8609	2.7177	**1.7992**

TABLE IV
Average Interval for Classroom Problem

No	UKFNN	EKFNN	$\text{IMM-EKFNN}_{1,2}$	**IMM-EKFNN_1 IMM UKFNN_2**	**His. Ave. Interval**
1	2.9627	3.2209	2.5370	**3.2424**	**2.3837**

2	2.7126	2.4298	3.2955	**3.4639**	**2.8340**
3	2.9047	3.8238	2.7742	**3.3949**	**1.9705**
4	2.9551	2.7305	2.6766	**3.4197**	**2.1446**
5	2.7510	5.2619	3.5068	**3.6220**	**2.2160**
6	2.8565	3.2193	2.4715	**3.4014**	**1.8948**
7	2.7383	2.4477	2.8636	**3.3978**	**2.9912**
8	2.8880	5.4240	3.3940	**3.4157**	**2.6018**
9	2.9889	5.0304	2.4852	**3.2319**	**2.3083**
10	2.6840	2.4395	3.0360	**3.4505**	**2.8802**
11	2.8959	2.9054	2.9979	**3.4372**	**1.9057**
12	2.9199	3.2569	2.4847	**3.3979**	**2.0853**

TABLE V

One Sigma Coverage (%) for Classroom Problem

No.	UKFNN	EKFNN	IMM-EKFNN$_{1,2}$	**IMM-EKFNN$_1$ IMM UKFNN$_2$**
1	47.8261	73.9130	60.8696	**82.6087**
2	39.1304	56.5217	34.7826	**73.9130**
3	56.5217	34.7826	47.8261	**95.6522**
4	43.4783	69.5652	47.8261	**91.3043**
5	60.8696	34.7826	39.1304	**82.6087**
6	47.8261	65.2174	69.5652	**95.6522**
7	30.4348	39.1304	34.7826	**73.9130**
8	56.5217	69.5652	30.4348	**82.6087**
9	47.8261	52.1739	60.8696	**86.9565**
10	39.1304	56.5217	39.1304	**78.2609**
11	56.5217	47.8261	47.8261	**95.6522**
12	43.4783	60.8696	47.8261	**100.0000**

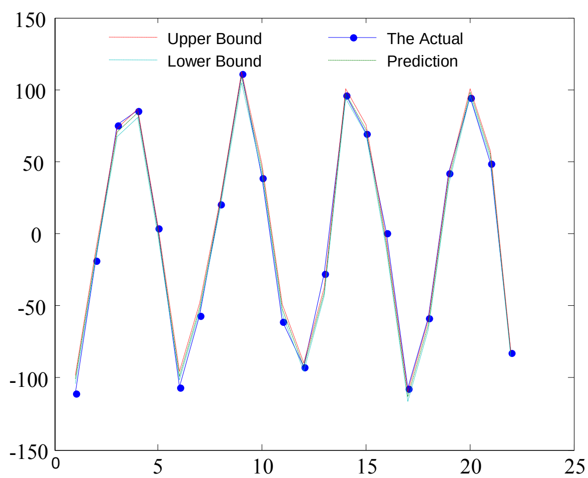


Figure 6. Actual and predictions with estimated CIs

Example 2. This example compares different ways of using Kalman trained networks with single UKFNN, single EKFNN, two IMM-EKFNNs, and IMM-UKFNN and IMM-EKFNN. Training period is from October, 2006 to December, 2007, and prediction from January, 2008 to June, 2008. Table III demonstrates that our presented method in the last column of the table superiors to the former three in terms of the MAE; Table IV lists the estimated average interval and indicates the estimated CI is very close to historical averaged interval. Table V shows that our method has the largest one sigma coverage. Fianally, the actual and predictions together with lower and upper bounds are plotted in Figure 5, which shows predictions follow the actual quite well with good estimated CIs.

TABLE VI

Hourly MAE (MW) for ISO-NE 2008 Load (January – June)

Min.	UKFNN	EKFNN	IMM-EKFNN$_{1,2}$	**IMM-EKFNN$_1$ IMM UKFNN$_2$**
5	31.0876	17.6997	18.1214	**18.6394**
10	52.4165	27.0772	28.1326	**27.7465**
15	69.6501	33.7970	36.1952	**35.8595**
20	84.7680	40.3436	42.5046	**42.0913**
25	96.8930	47.1390	48.8989	**48.5038**
30	113.8299	53.5859	55.0000	**54.6265**
35	128.9448	60.4286	61.5044	**60.8765**
40	154.5469	66.8100	68.0365	**67.1928**
45	169.4482	72.8069	74.3490	**73.0295**
50	171.8297	79.1913	80.3075	**78.8457**
55	184.0054	85.7669	87.0814	**84.6589**
60	198.2765	94.4831	94.2359	**91.0988**

TABLE VII

Hourly Average Interval (MW) for ISO-NE 2008 Load (January – June)

Min.	UKFNN	EKFNN	IMM-EKFNN$_{1,2}$	**IMM-EKFNN$_1$ IMM UKFNN$_2$**
5	37.7291	35.6316	35.7334	**36.5198**
10	54.9670	49.9406	49.7115	**51.1372**
15	54.9670	60.9013	60.2495	**62.0119**
20	85.1550	70.1374	69.0069	**71.1696**
25	93.7346	78.2793	76.6842	**79.2229**
30	104.8561	85.7997	83.7719	**86.5615**
35	117.3021	92.7464	90.7545	**93.4984**
40	130.2830	99.1993	97.4770	**100.0277**
45	136.8371	105.3371	103.3674	**106.0310**
50	146.0397	111.2197	109.0308	**111.7823**
55	151.8509	116.9204	114.5637	**117.3741**
60	157.1169	122.8449	120.1376	**122.9854**

TABLE VIII

One Sigma coverage (%) for ISO-NE 2008 Load (January – June)

Min.	UKFNN	EKFNN	IMM-EKFNN$_{1,2}$	**IMM-EKFNN$_1$ IMM UKFNN$_2$**
5	67.6969	79.2811	88.7592	**88.0723**
10	61.3095	76.2592	84.6383	**86.6071**
15	59.8901	75.3892	82.9670	**84.2949**
20	59.0659	74.2674	82.3031	**83.7454**
25	58.1273	73.3516	81.2042	**83.0357**
30	57.4863	71.9780	80.3342	**82.3489**
35	56.5476	71.1310	79.9908	**81.2729**
40	54.9451	69.3223	79.0064	**80.2427**
45	53.9606	68.5668	78.1364	**79.2811**
50	52.9762	67.3993	77.2665	**78.4799**
55	52.1291	66.4377	76.5797	**77.5412**
60	50.4808	64.7436	74.9542	**76.3736**

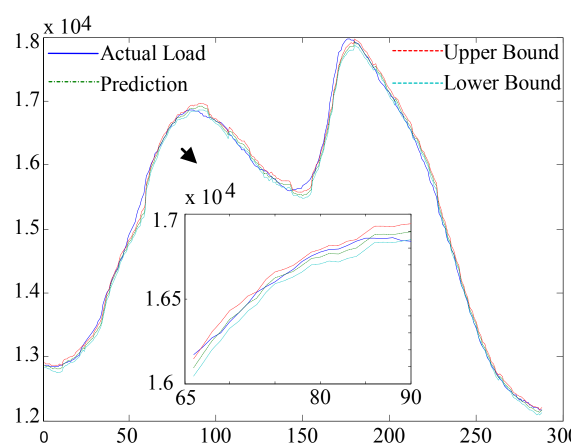


Figure 7. Actual & predictions of Load (Jan. 26th, 2008) with estimated CIs

IV. CONCLUSION

A method of interacting multiple models using EKFNN and UKFNN is important for VSTLF with CI estimation. The peak and valley hour indices together with other data help prepare good inputs to neural network. The EKFNN and UKFNN can accurately capture the features of the nearly linear and nonlinear hourly input-output functions. Using the dynamical mixing weights, IMM will further determine which function should be relied upon with a high priority at each step, and thus perform the effective forecasting. The combination of IMM-EKFNN and IMM-UKFNN performed significantly better than a single UKFNN or EKFNN, and the combination of two IMM-EKFNNs in load forecasting, average interval and one sigma coverage. Testing on New England's load showed that our method produce an accurate load forecasting with very good estimated CIs.

REFERENCES

[1] Bar-Shalom, Y., Li, X. R. and Kirubarajan, T. (2001). Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction, J. Wiley and Sons, 200-209, and 385-386.

[2] Charytoniuk, W and Chen, M. S. (2000). Very Short-Term Load Forecasting Using Artificial Neural Networks, *IEEE Transaction on Power Systems, 15(1),* 263-268.

[3] Guan, C., Luh, P. B., Coolbeth, M. A., Zhao, Y., Michel, L. D., Chen, Y., Manville, C. J., Friedland., P. B. and Rourke, S. J. (2009a). Very Short-Term Load Forecasting: Multilevel Wavelet Neural Networks with Data Pre-filtering, *Proceedings of IEEE Power & Energy Society 2009*.

[4] Guan, C., Luh, P. B., Michel, L. D., Coolbeth, M. A. and Friedland, P. B. (2009b) Hybrid Kalman Algorithms for Very Short-term Load Forecasting and Confidence Interval Estimation, submitted to *IEEE Power and Energy Society 2010 General Meeting*.

[5] Haykin, S. and Kailath, T. (2002). Adaptive Filter Theory, Fourth Edition, *Pearson Education, Inc,*.759-760.

[6] Ilyas, M., Lim, J., Lee, J. G. and Park, C. G. (2008). Federated Unscented Kalman Filter Design for Multiple Satellites Formation Flying in LEO, *International Conference on control, Automation & Systems,* 453-458.

[7] Julier, S.J., Uhlmann, J. K. and Durrant-Whyte, H.F. (1995). A New Approach for Filtering Nonlinear Systems, *Proceedings of the American Control Conference,* 1628-1632.

[8] Puskorius., G. V. and Feldkamp, L. A. (1991). Decoupled extended Kalman filter training of feedforward layered networks. *International Joint Conference on Neural Networks, Seattle 1991,* 771-777.

[9] Singhal, S. and Wu, L. (1989). Training Multilayer Perceptrons with the Extended Kalman Algorithm. *Proceedings of International Conference on Acoustics, Speech and Signal Processing* 2, 1187–1190.

[10] Zhang, L. and Luh P. B. (2005). Neural Networks – Based Market Clearing Price Prediction and Confidence Interval Estimation With an Improved Extended Kalman Filter Method. *IEEE Transaction on Power Systems, 20(1),* 59-66.