Very Short-term Load Forecasting: Multilevel Wavelet Neural Networks with Data Pre-filtering

Che Guan, Peter B. Luh, *Fellow, IEEE*, Matthew A. Coolbeth, Yige Zhao, Laurent D. Michel,

Ying Chen, *Student Member*, *IEEE*, Claude J. Manville, Peter B. Friedland and Stephen J. Rourke

***Abstract*--Very short term load forecasting predicts the load over one hour into the future in five minute steps, and is important in resource dispatch and area generation control. Effective forecasting, however, is difficult in view of noisy real-time data gathering and complicated features of load. This paper presents a method based on multilevel wavelet neural networks with novel pre-filtering. The key idea is to use a data pre-filtering method to detect and eliminate spikes within load, apply the wavelet technique to decompose the load into several frequency components, perform appropriate transformation on each component, and feed it together with other appropriate input to a separate neural network. Numerical testing demonstrates the significant value of data pre-filtering and multilevel wavelet neural networks, and shows that our method provides accurate forecasting.**

Index Terms*--Multilevel wavelet decomposition, Neural networks, Pre-filtering, Very short-term load forecasting.*

I. INTRODUCTION

VERY short-term load forecasting (VSTLF) has been essential for the area generation control and resource dispatch. 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. Very short-term load forecasting, however, is difficult in terms of the effective data gathering and complicated features of the load.

Methods for very short-term load forecasting are limited. Existing methods include extrapolation, autoregressive, fuzzy logic, Kalman filtering, and neural network (NN). The first two often use recent load (e.g., load of the last hour) and statistical historical load (e.g., the mean of similar day load with same time index and weekday index to the target) to extrapolate the future. The last three often use recent load to update system parameters and then generate the forecast. Usually, weather condition is ignored because of the large time constant of load as a function of weather. The representative methods will be briefly reviewed in Section II. These models and techniques are practical and applicable, but cannot fully extract key features of very short term load. Another important aspect of VSTLF is to effectively handle noisy load data. Because of the malfunctioning of data gathering devices, spikes of different widths and amplitudes are often present in the data, affecting training and forecasting. However, few of existing methods analyze the effects of spikes on the forecast load or present effective ways to detect and correct them.

On a different front, progress has been made on short-term load forecasting. Our recent work [20] combines the similar day method, single-level wavelet decomposition, and neural networks to effectively forecast New England load. This method opens the door for analyzing complex load features at different frequencies. However, since features of very short term load are different from those of short term load, new methods need to be developed for VSTLF.

In this paper, the multilevel wavelet neural network method with novel data pre-filtering is presented to forecast next hour's load in five-minute steps. To reduce the effects of spikes on prediction, a novel data pre-filtering technique is developed to detect and correct spikes from raw load data. This method produces a new smoother signal without spikes without altering normal data. The smoothened load contains multiple frequencies. To capture the features at different frequencies, the wavelet technique is used to decompose the load into multiple levels. Each level is appropriately transformed and fed into a separate neural network so that the load features can be well captured. Forecasts from individual networks are then transformed back and combined to form the overall forecast. Numerical testing results for a simple example and for forecasting New England's load in Section IV demonstrate the values of data pre-filtering, multilevel wavelet decomposition, load transformation and neural networks for VSTLF.

II. LITERATURE REVIEW

Among VSTLF models, extrapolation may be the most straightforward method. It constructs new load based on a discrete set of known data, e.g., by direct extrapolation with least square [1] or curve fitting [2]. Sometimes, load increment is predicted, e.g., by using weighted average of load increments of previous load as the forecast increment [3]. A similar increment prediction method was reported in [4] with load pre-grouped into sets based on similarity of the load.

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

Che Guan, Peter B. Luh, Yige Zhao and Ying Chen are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269-2157 USA.

Laurent D. Michel and Matthew A. Coolbeth and Claude J. Manville are with the Department of Computer Science and Engineering, University of Connecticut, Storrs, CT 06269-2155.

Peter B. Friedland and Stephen J. Rourke are with ISO New England, Holyoke, MA 01040 USA.

978-1-4244-4241-6/09/$25.00 ©2009 IEEE

Similarly, auto-regression has been used to generate forecast based on the known past. A simple linear combination of previous load was used to perform forecast, the coefficients of which were updated online via least mean square algorithm [5]. Auto-regression was extended to an autoregressive integrated moving average method which is applicable for stochastic problems and recursive forgetting factor least square was used to update system parameters by [6]. An advantage of this model-based approach is that it can extrapolate a short-length data sequence to create a longer data sequence for improved power spectrum estimation [21].

Fuzzy logic can identify and approximate any unknown nonlinear dynamic systems on the compact set to arbitrary accuracy [15]-[16]. A fuzzy logic system [5] was implemented by drawing similarities in load trend (e.g., between weekdays and weekdays) from a huge of data. A pattern database generated via effective training was then used to predict the load change. Fuzzy logic was also merged into neural network named 'fuzzy neuron system', and parameters of which were optimized via chaos theory [7]-[8], [17].

The Kalman filtering is used recursively to estimate the optimal load forecast parameters and overcome the unknown disturbance in the linear part of the systems during load prediction [18]. To generate a good forecast, load was separated into deterministic and stochastic components in [9] [10]. Both components can be predicted via the Kalman estimator. To make it simple, the former was forecasted via least square [9].

Neural networks assume a functional relationship between load and affecting factors, and estimate the functional coefficients by using historical data. Their functional forms are highly non-linear [20]. Most neural network methods for VSTLF use inputs e.g., time index, load of previous hour, load of the yesterday and previous week with same hour and weekday index to the target hour [11]-[13]. Load may be preprocessed before used (e.g., spikes are simply replaced with artificially created reasonable value.) Specifically, the load in second (e.g., load generated per three second) has to be integrated into minute load (e.g., load in five minute step) [13] because it is convenient for the network to perform the forecast with the same time step for both input and output load. Then load in minute step (e.g., five minute) are fed to the network in different ways. Relative increment in load of previous hour was used as inputs in [11] since the daily patterns of relative load increments are more repeatable than the daily load curves. Logarithmic difference in load of last hour and a half was used as inputs to improve data stationary and to emphasize the look-ahead feature of training data [13]. Both load and increment in load of last hour and week were adopted in [12]. Variables representing weather conditions are often neglected due to the relatively large time constant of the load and weather relationship [11]. Furthermore, different networks are used for particular time leads, certain periods of day (e.g., summer and autumn) with unique pattern in load dynamics, and different trends and specifications in load data (e.g., load of previous hour, yesterday and last week). However, the effect of generated spikes (caused by the malfunction of devices) on training and the accuracy of forecasts are seldom analyzed, and a few of papers present effective ways to detect and correct them. Besides, real-time load contain many features (e.g., hourly and weekly trends with fluctuation in the minute step), it's difficult to find proper ways to capture their properties with a single model.

III. Multilevel Wavelet Neural Networks with Novel Data Pre-filtering

VSTLF deals with different time leads (e.g., four second, five minute and etc.) Our work here is to forecast the load hourly in five minute step. It is convenient for the network to process the input load with same step. Therefore, the four second load generated from data gathering devices is integrated into the five minute before the forecast is taken.

The method of multilevel wavelet neural networks with data pre-filtering is presented here for very short-term load forecasting over one hour into the future in five minute steps. Our method consists of spike filtering, multiple level wavelet decomposition, and neural networks. The process as a whole is depicted as in Figure 1. Subsection III-A introduces data pre-filtering for spike detection and elimination. Subsection III-B describes the multilevel wavelet decomposition technique. Subsection III-C presents the appropriate load transformation for different decomposed components of the load to neural networks. Forecasts from individual networks are then combined to form the overall forecast.

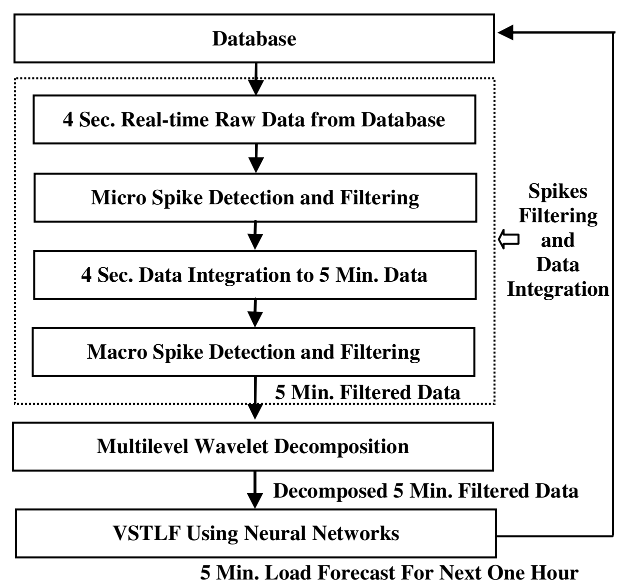


Fig. 1. Overall Structure of the Method

A. *Spike Filtering*

Spikes caused by the malfunctioning of data collection devices can have different amplitudes (positive or negative) and durations (last in seconds or minutes), and are randomly distributed over the load signal. They do not reflect true load, affect neural network training, and degrade prediction quality. Thus, it is desirable to correct them before the signal is used.

Empirically, segments of load with slightly abnormal amplitudes usually have short duration, whereas those with large abnormal amplitudes often last long. Here, they are respectively denoted as "micro spikes" and "macro spikes," and will be treated differently. In our work, the micro and macro spike filtering are presented for the two types of spikes. The former detects and repairs the micro spike with the smoothed signal but does not change normal load; the latter corrects the macro spike with the linear interpolation. The two filtering are successively used to process five minute load but only the micro spike filtering is used for the four second load before the load is integrated into the five minute. Macro spike filtering is not used to process the four second load because it can largely change the raw load and brings the large difference to the five minute load after the integration.

The key idea for micro-spikes filtering is to use a "zero phase filter" over the raw signal to smooth it out. If the difference between this smoothed signal and the original signal exceeds a threshold, the spike is found and located. Then the spike is replaced with a corrected signal. Different from other filters which smooth both spikes and normal load, this method does not alter normal load and produces a new smoother signal without spikes.

The approach to smooth the load is to perform the zero phase filtering by processing the input data X = {x(1),…,x(n)} in both forward and reverse directions with equal weights over an interval. The smoothed sequence Y = {y(1),…,y(n)} produced by (1) has zero phase distortion meaning no time shift but exhibits a slight magnitude modification.

$$y(n) = \frac{\sum_{i=n-m+1}^{n} x(i)}{m} \qquad (1)$$

where n is the filter order, m is filter order of the feedback and feed forward, which informs the length of the filter window, and y(n) is the average of x(n-m+1) to x(n).

The spike indicator Z = {z(1),…,z(n)} is produced by (2)

$$z(n) = y(n) - x(n) \qquad (2)$$

where a large value (positive or negative) for z(n) is indicative of a spike. The micro spike is identified if the absolute value of z(n) exceeds a threshold (THD1). To replace a spike with a corrected signal, the value of x(n) is replaced by y(n). This is a localized correction which does not alter normal segments of original signal. Figure 2 shows details of the process.

The idea of macro spike filtering for five minute load is similar to the filtering at the micro level. Different from the micro spike(s) repair technique, simply performing a local replacement of the original signal by its filtered counter-part will not be effective given that the smoothed data y(n) will be very much affected by the average of the window [x(n-m+1)…x(n)]. Therefore, an advanced approach is to detect the edges of the spike within the windows flagged by the zero-phase filter defined by equation (1). The load segment between two edges is then replaced by a linear interpolation. Because the joints between the interpolated portion and the normal signal are not as smoothed as the true load, connected edges are smoothed by the zero-phase filter. This repair method is to side-step the excessive averaging that happens right at the edge of the macro spike(s). Figure 4 depicts the five minute load before and after the macro filtering method being applied.

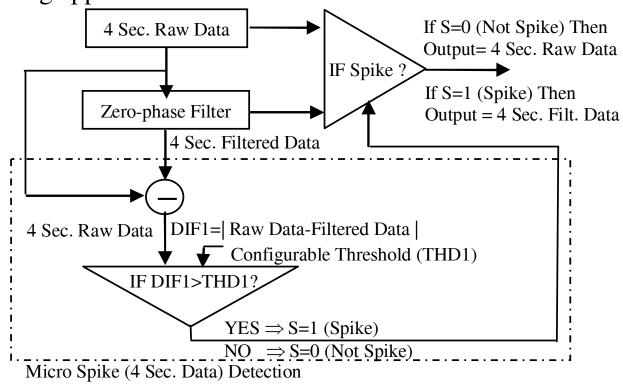


Fig. 2. Micro spikes filtering

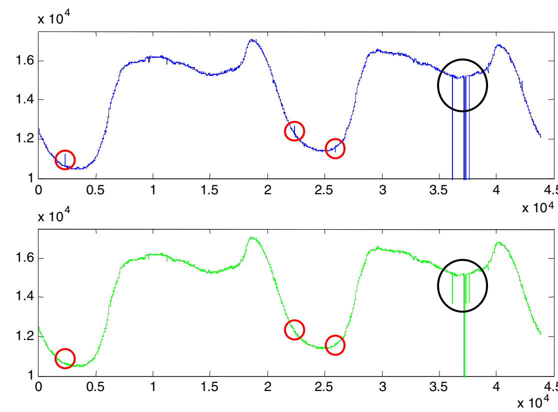


Fig. 3. Micro spikes of 4 sec. load filtering bef. (up) & after (down)

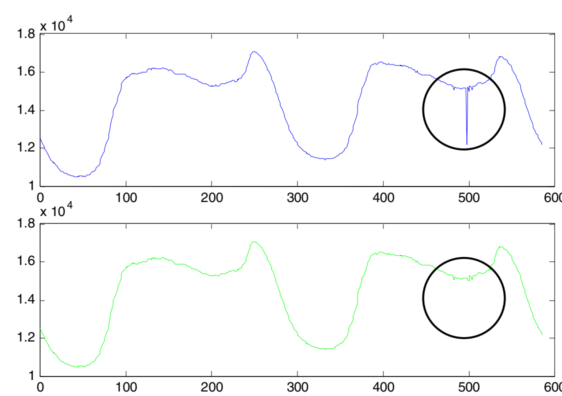


Fig. 4. Macro spikes of 5 min. load filtering bef.(up) & aft.(down)

B. Multi-level Wavelet Decomposition

Very short term load has a variety of features (e.g., periodical daily and hourly pattern, and repeatable minutely relative increment in load.) In view of the frequency, the load spectrum presents different trends of load pattern (e.g., the slowly and fasting changing components). An intuitive way is to decompose the load signal into different frequency components (e.g., low and high frequency components.) The low frequency as called "approximation" represents a slowly changing of the signal, whereas the high frequency component as called "detail" is the difference between two successive approximations [22] [23]. On a different front, our previous work for STLF provides accuracy forecast with one-level wavelet decomposition [20]. The process consists of the analysis and the synthesis stages. At former stage, the low and high pass filters are respectively used to compute the average and difference of indexed load data and obtain decomposed coefficients; in the latter stage, each coefficient is passed to the "inverse low filter" or "inverse high filter" to reconstruct the signal. In this way, orthogonal components (low and high frequency components) are obtained with very small error to the original data when components are summed. The multiple level decompositions repeat the process. But it is only necessary to further decompose the low frequency because it is hard for the network to capture the decomposed components of the high frequency.

In our method, Daubechies 4 wavelet is used to decompose the load (previous hour and similar day) into four levels, which generate the best forecast according to testing results (Example 3 of Section IV). As depicted in Figure 5, the input load is decomposed into low (L) and high (H) frequencies at level one, then the low frequency is further decomposed in to low-low (LL) and low-high (LH) components. Repeating this process, low-low-low (LLL), low-low-high (LLH), low-low-low-low (LLLL) and low-low-low-high (LLLH) components are sequentially obtained. Components included in the dashed line are fed to separate network to capture the property.

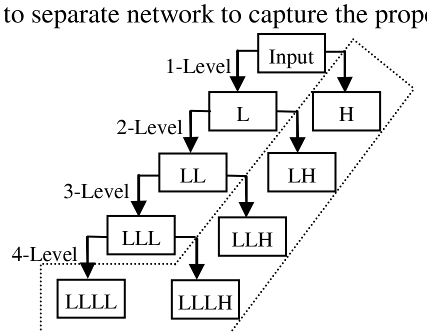


Fig. 5. Wavelet Decomposition into Four Levels

C. Neural Networks

In order to improve the forecasting, the load for a historical hour similar to the target hour for a forecast appears among the input to the network. Then the selected load of similar day and the load of previous hour are decomposed into low and high frequency components. To process different components of the load signal, a simple way is to treat them equally and pass each component to a network. It is reasonable for the network to directly process high frequencies which represent changes in the load. However, it is difficult for the network to capture features of the low frequency component since it delivers less useful information than the relative increment (RI) in load as defined by (3)

$$RI\,L_d(t) = \frac{L_d(t) - L_d(t-1)}{L_d(t-1)} \qquad (3)$$

where L represents the load, t represents the sampled time at 5-minute intervals, and subscript d is the day index.

The key idea to capture the property of each decomposed component is to feed relative increment in load (previous hour and similar day) of the low frequency, the load (previous hour) of high frequencies, and together with hour and weekday index to separate networks as shown in Figure 6.

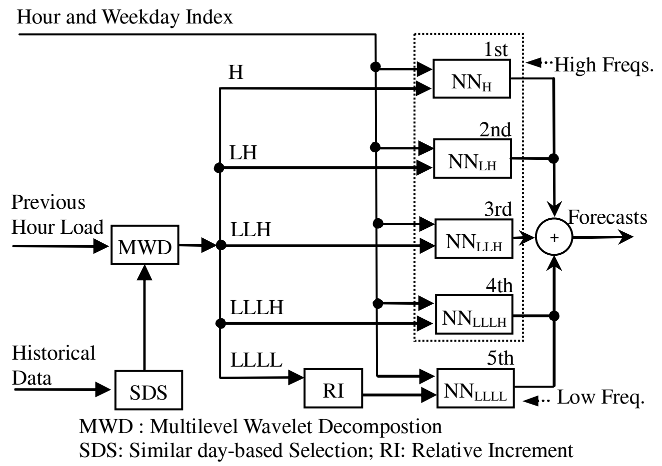


Fig. 6. Structure of Multilevel Wavelet Neural Networks

Low (LLLL) frequency neural network predicts the load for the next hour by using the low frequency of previous hour's load, a chosen similar day's load, and the hour and weekday index as depicted in Figure 7. The low frequency components of previous hour and similar day present hourly daily and weekly patterns as well as repeatable increment changes. Their properties can be well captured when their relative increments are computed and fed to the network. This combination has been demonstrated the best prediction accuracy as described in example 3 of Section IV.

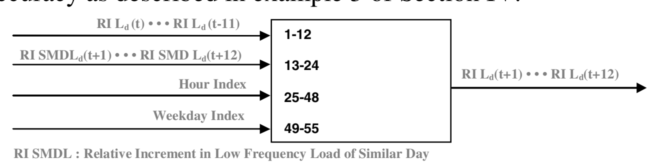


Fig. 7. Low (LLLL) Frequency Neural Network

The rationale supporting the use of relative increment in load as inputs is that even if current weather conditions are different from those used for training, the increment may still have similar trend or curvature [11]. Intuitively, the computation of the relative increment in load plays the role of a high pass filter which properly captures the periodical changes of the load. The prediction error will propagate as forecast step increases.

To generate a precise forecasting, the method is to select the best similar day, the day of history that is most similar to current day [20], then select corresponding hour of that day. The best similar day, for a given day, d, is the day among all previous days with the same weekday index as d, which has the lowest similarity index with d. The similarity between days S and T index can be computed by (4)

$$Similarity = \frac{\sum_{i=1}^{24}(|W_t - W_s| - |H_t - H_s|)}{24} \quad (4)$$

where W and H denote the humidity and wind-chill temperature for each hour of a day, t denoting the target day and s the day whose index is to be computed.

The high frequency (H, LH, LLH and LLLH) neural networks as depicted in Figure 8 directly predict the high frequency load components because the relative increment in high frequency components change too fast to be captured by the network. Testing from Example 3 of Section IV shows that the network can capture the features of the four level high frequency components if they are direct used. Here, neither high frequency components of similar day load nor their relative increments are used due to their weak relations to high components of the load of the target hour. The testing from Example 3 of Section IV also demonstrates the effect of different components of similar day to the forecast accuracy. Other inputs (hour index and weekday index) are identical to the low frequency network.

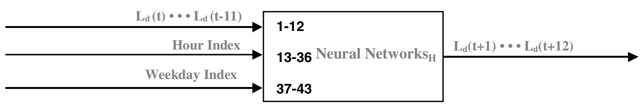


Fig. 8. High Frequency (H, LH, LLH, LLLH) Neural Networks

Five multilayer perception networks are used for five components of the load signal. These networks are trained by using historical data and update hourly with the latest information. The training terminates when the training error is less than or equal to a specified threshold. The final forecasted load is the sum of predictions generated by five networks. The accuracy of the forecast can be evaluated by using the standard Mean Average Percentage Error (MAPE) by (5)

$$MAPE = \frac{1}{12}\sum_{i=1}^{12}\frac{|L_P(t+i) - L_A(t+i)|}{L_A(t+i)} \times 100\% , \quad (5)$$

where t represents the sampled point every 5 minute, and $L_A(t)$ and $L_P(t)$ respectively denote actual and predicted values for the load of next hour.

IV. Numerical Testing Results

The above method has been implemented in MATLAB on a Pentium Dual Core 2.20GHz personal computer. Three examples are presented below. Example 1 uses a classroom-type problem to examine the combination of multilevel wavelet decomposition and neural networks. Example 2 predicts New England 2007 load and demonstrates the new filtering method. Example 3 predicts New England 2008 load, and examines the effects of using multilevel wavelet decomposition, transformations for each component to the network, and similar day's load on prediction accuracy.

Example 1. Consider the following signal:

$$\begin{aligned} y(t) &= 200\sin(2\pi\cdot 50\cdot t) + 20\sin(2\pi\cdot 110\cdot t) \\ &\quad + 11\sin(2\pi\cdot 230\cdot t) + 6\sin(2\pi\cdot 400\cdot t) \\ &\quad + 14\sin(2\pi\cdot 600\cdot t), \end{aligned}$$

which is composed of a low frequency component $200sin(2\pi{\cdot}50{\cdot}t)$ and a serious of high frequency components $20sin(2\pi{\cdot}110{\cdot}t)$, $11sin(2\pi{\cdot}230{\cdot}t)$, $6sin(2\pi{\cdot}400{\cdot}t)$ and $14sin(2\pi{\cdot}600{\cdot}t)$ Two thousand noisy data sets $(t, \tilde{y}(t))$ were randomly generated for training with

$$\tilde{y}(t) = y(t) + \varepsilon(t),$$

where $t \in [1, 2\ldots2000]$ and $\varepsilon(t) \in$ U(0, 0.5). The objective is to predict $y(t)$ for $t \in$ [2001, 2002…2100]. This constructed signal has a strong resemblance to the actual load in terms of the relative amplitude and frequency.

Our method and a single neural network which directly processes the relative increment in load without wavelet decomposition are tested. Mean absolute error (MAE) as defined by

$$MAE = \frac{1}{100}\sum_{t=2001}^{2100}|\tilde{y}(t) - y(t)|,$$

is calculated for both predictions. For our method, MAE is 1.2763, and for the standard NN method, it is 8.3787. This indicates that the prediction obtained by using multilevel wavelet networks (four level in our method) is closer to the true signal y(t) than the prediction obtained by using a single NN method. The Predictions obtained by using our method are plotted in Figure 9. The y(t) with its decomposed components are also drawn as the comparison in the figure .

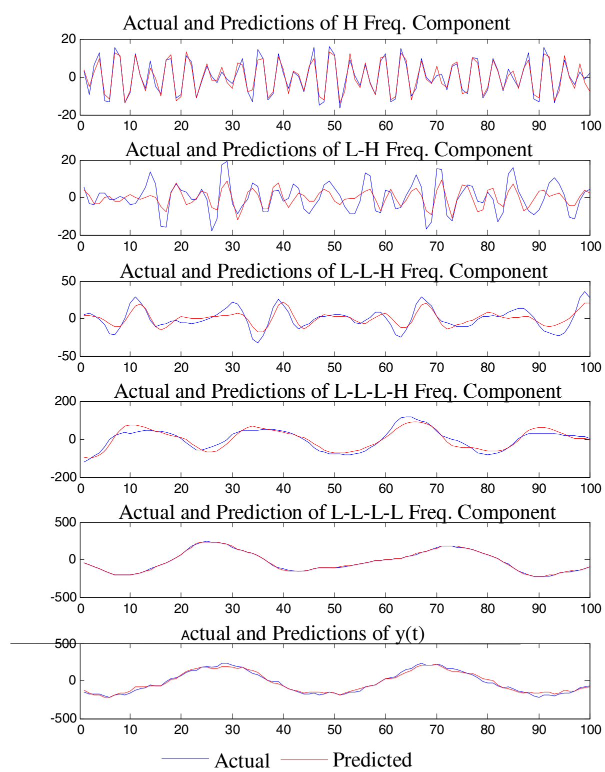


Fig. 9. Results for Example 1

Example 2. The new filtering method is used to predict New England 2007 load from January to June. Training period is from October, 2006 to December, 2006. MAPEs presented in Table I show that our new filtering method improves the forecast against feeding the load without filtered to NN. In all cases, the best number of hidden neurons for each network is list in the order: H-LH-LLH-LLLH-LLLL, etc.

TABLE I
MONTHLY MAPES (%) FOR NEW ENGLAND 2007 WINTER LOAD

	With new filtered signal	With load not filtered
January	.3418	.6447
February	.3029	.5664
March	.4174	.6191
April	.4070	.5566
May	.3889	.5474
June	.3265	.4310
No. of Hidden Neurons	5-5-10-3-27	8-5-5-12-11

Example 3. The values of multilevel wavelet decomposition, the transformation for each decomposed component, and the load of the similar day are examined to perform New England 2008 load from January to June. Training period is from October, 2006 to December, 2007. Four cases are included. Case 1 compares our method with different levels of decomposition with the use a single neural network without wavelet decomposition. Case 2 compares the combination of load transformations to neural networks and shows the best adopted one. Case 3 shows the benefits of using the low frequency of the similar day's load. Case 4 shows the precise predictions. The nomination for all cases is to feed the low frequency component of the relative increment in load of previous hour and similar day, and high frequency components of the load to each NN.

Case 1. In order to effectively capture load features, the multilevel wavelet decomposition from one level to five levels to each dedicated network and a single network without multilevel wavelet decomposition are compared. MAPEs presented in Table II show that multilevel wavelet decomposition improves prediction accuracy (for the five minute, ten minute,…, one hour out) as compared to a single NN method. It also shows four level wavelet neural networks result in the best prediction accuracy. MAPEs in Table III show the same conclusion but further convey an overview of the monthly forecast.

TABLE II
HOURLY MAPES (%) FOR WAVELET NEURAL NEWTWORKS WITH / WITHOUT MULTILEVEL DECOMPOSITION

	Single NN	One Level	Two Level	Three Level	Four Level	Five Level
5 min	.1474	.1115	.0966	.0854	.1073	.2633
10 min	.2544	.2386	.1856	.1392	.1379	.2662
15 min	.3410	.3362	.2523	.1833	.1735	.2846
20 min	.4215	.4033	.3058	.2258	.1999	.3041
25 min	.4999	.4767	.3643	.2759	.2351	.3508
30 min	.5798	.5470	.4245	.3330	.2801	.4164
35 min	.6652	.6232	.4886	.3836	.3216	.4692
40 min	.7589	.7076	.5618	.4388	.3638	.5138
45 min	.8467	.7793	.6289	.4934	.4143	.5825
50 min	.9382	.8498	.6932	.5483	.4667	.6448
55 min	1.0297	.9271	.7616	.6010	.5294	.7066
60 min	1.1241	1.013	.8406	.6607	.5931	.8157

TABLE III
MAPES (%) FOR WAVELET NEURAL NEWTWORKS WITH / WITHOUT MULTILEVEL DECOMPOSITION

	Single NN	One Level	Two Level	Three Level	Four Level	Five Level
January	.5584	.5191	.4397	.3673	.3550	.6402
February	.6294	.5934	.4878	.3634	.3593	.4288
March	.7720	.6738	.5333	.4051	.3827	.4792
April	.7490	.7153	.5316	.4465	.3249	.4711
May	.6345	.5811	.4788	.3497	.2655	.4745
June	.5556	.5209	.4089	.3060	.2760	.3970
No. of Hidden Neurons	11	6-12	13-6-5	13-6-14-8	6-13-12-13-18	7-8-7-12-13-26

Case 2. The transformations for each component to NN are compared. MAPEs show that the one using relative increment in load of the low frequency to NN_L and high components to NN_H leads to the best prediction accuracy at .3% level, that feeding the low frequency component to NN_L and relative increment in high frequency components to NN_H leads to 2%, and that feeding all frequency components or their relative increments to both NN_L and NN_H leads to very large errors.

Case 3. This example compares different ways of using the load of similar day: not using load of similar day; using load of similar day to NN_L and NN_H; using high component of similar day load to NN_H; using low component of similar day load to NN_L. MAPEs presented in Table IV show that the first three ways degrade the prediction accuracy as comparing to the last way (our presented method).

TABLE IV
MAPES (%) VALUE OF USING THE SIMILAR DAY'S LOAD
(NEW ENGLAND 2008 LOAD)

	With load of last hour	With load of last hour and similar day	With load of last hour and high freq. of similar day	With load of last hour and low freq. of similar day
Jan	.3433	.3657	.4864	.3550
Feb	.3446	.3580	.5690	.3593
March	.4008	.4315	.6176	.3827
April	.3452	.3748	.6335	.3249
May	.2777	.3188	.5296	.2655
June	.2766	.3315	.4768	.2760
No. of Hidden Neurons	7-7-6-7-20	6-6-11-12-13	5-13-12-13-13	6-13-12-13-18

Case 4. The five components of the load in January 1, 2007 are predicted. The MAEs for each predicted decomposed component are list in Table V. The predictions together with

the actual are plotted in Figure 10. The result shows predictions follow the actual quite well.

TABLE V
ACCURACY MAES (MW) OF LOAD FREQUENCIES

	H	LH	LLH	LLLH	LLLL
Jan	13.8517	14.5573	8.3237	47.7551	53.2032
Feb	14.1566	14.7165	8.2966	44.1162	52.0918
March	15.8425	16.1979	8.4769	40.2735	59.6286
April	15.6786	16.4230	8.6618	45.2211	44.1583
May	14.0379	14.4596	7.3972	33.8316	34.0491
June	16.4451	16.9491	8.0365	39.3184	41.2554

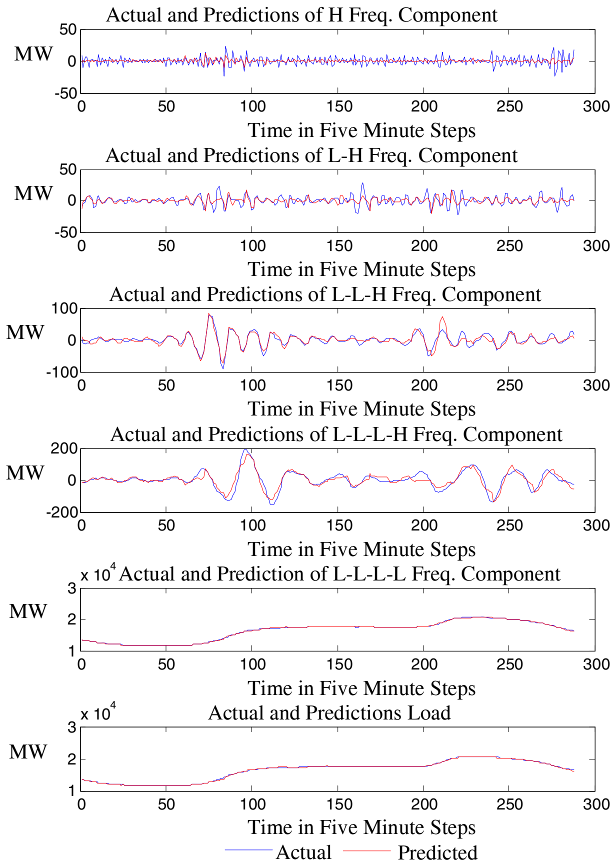


Fig. 10. Actual and predictions of Frequency Load (Jan. 1st, 2008)

V. CONCLUSION

This paper presents a neural network-based approach incorporating multilevel wavelet decomposition and a novel data pre-filtering method as to forecast very short-term load over one hour in future in five minutes steps. The key idea is to detect spikes and replace them with corrected signals, but the normal data are not altered. A synergistic combination of multi-level wavelet decomposition and neural networks is used to capture key features of load at different frequencies. To better improve the forecast, similar day's load is used to prepare good input data. Testing results show that this method provides accurate predictions.

VI. REFERENCES

[1] F. Wang, K. Xie, E. K. Yu, G. Q. Liu and M. Y. Wang, "A Simple and Effective Ultra Short Term Load Forecasting Method," Power System Technology, 1996, 20(3):41-43,48.

[2] D. S. Luo and H. Y. He, "A Shape Similarity Criterion Based Curve Fitting Algorithm and Its Application in Ultra-Short-Term Load Forecasting," Power System Technology, 2007, 31(21):81-84.

[3] J. Y. Zhou, B. M. Zhang, J. C. Shang, J. Yao and M, Cheng, "Very Short Term Load Forecast Based on Multi-sample Extrapolation and Error analysis," Electric Power Automation Equipment, 2005, 25(2):17-21.

[4] Z. L. Yang, G. Q. Tang, Y. M. Song and R. Z. Cao, "Improved Cluster Analysis Based Ultra-short Term Load Forecasting Method," Automation of Electric Power System, 2005, 29(24):83-86.

[5] K. Liu, S. Subbarayan, R. R. Shoults, M. T. Manry, C. Kwan, F. L. Lewis and J. Naccarino, "Comparison of Very Short-Term Load Forecasting Techniques," IEEE Transaction on Power Systems, 1996, 11(2): 877-882.

[6] J. C. Lu, X. P. Zhang and W. Sun, "A Real-time Adaptive Forecasting Algorithm for Electric Power Load," 2005 IEEE/PES Transmission and Distribution Conference & Exhibition, 2005, China.

[7] H. Y. Yang, H. Ye, G. Wang, J. Khan and T. Hu, "Fuzzy Neural Very Short term Load Forecasting Based on Chaotic Dynamics Reconstruction," Chaos, Solitons and Fractals, 2006 29(2): 462-469.

[8] S. Kawauchi, H. Sugiharra and H. Sasaki, "Development of Very-Short-Term Load Forecasting Based on Chaos Theory," Electrical Engineering in Japan, 2004, 148(2):646:653.

[9] K. Xie, F. Wang, and E. K. Yu, "Very Short-Term Load Forecasting by Kalman Filter Algorithm," Proceedings of the CSEE, 1996, 16(4):245-249.

[10] D. J. Trudnowski and W. L. Mcreynolds, "Real-Time Very Short-Term Load Prediction for Power-System Automatic Generation Control," IEEE Transaction on Power Systems, 2001, 9(2), 2001:254-260.

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

[12] H. Daneshi and A.Daneshi, "Real Time Load Forecast in Power System," 3rd International Conference on Deregulation and Restructuring and Power Technologies, DRPT 2008, 4523494: 689-695.

[13] P. Shamsollahi, K. W. Cheung, Q. Chen and E. H. Germain, "A Neural Network Based Very Short Term Load Forecaster for the Interim ISO New England Electricity Market System," IEEE Power Industry Computer Applications Conference, 2001, 217-222.

[14] L. Dichen, Q. Xiaoman, Z. Lin and H. Sheng, "Research of Adaptive Computation of Very Short-term Load Forecast in VQC," Transmission and Distribution Conference & Exhibition: Asia and Pacific, 2005 IEEE/PES, Aug 2005, China.

[15] J. H. Chow, F. F. Wu and J. A. Momoh, Applied Mathematics for Restructured Electric Power Systems, Springer, 2005, 270-285.

[16] H. K. Alfares and M. Nazeeruddin, "Electric Load Forecasting: Literature Survey and Classification of Methods," International Journal of System Science, 2002, 33 (1): 23-34.

[17] X. F. Li, E. Zou and T. S. Zhang, "Optimization design of Feed-forward Neural Network Structure based on Chaos Wariables," Control and Decision, 2003, 18 (6): 703-707.

[18] F. Zhao; H. S. Su, "Short-Term Load Forecasting Using Kalman Filter and Elman Neural Network," 2007 Second IEEE Conference on Industrial Electronics and Applications, 2007,1043-1047.

[19] ISO New England, "System Operation Procedures." April 4, 2007, www.iso-ne.com/rules_proceds/operating/sysop/out_sched/sop_outsch_0040_0010.pdf.

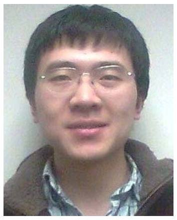
[20] Y. Chen, P. B. Luh, C. Guan, Y. G. Zhao, L. D. Michel, M. A. Coolbeth, P. B. Friendland and S. J. Rourke, "Short-term Load Forecasting: Similar Day-based Wavelet Neural Networks," IEEE Transaction on Power Systems, 2008, submitted.

[21] S. K. Mitra, "Digital Signal Processing: A Computer-Based Approach, Third edition" McGraw-Hill Companies, Inc., New York, 2007.

[22] S. G. Mallat, "A theory for multiresolution signal decomposition-the wavlet representation," IEEE Trans. Pattern Anal. Mach. Intell., 1989, 11(7) : 674-693.

[23] A. J. da Rocha Reis and A. P. Alves da Silva, " Feature Extraction via Multiresolution Analysis for Short-Term Load Forecasting," IEEE Transaction on Power Systems, 2005, 20 (1): 189-198.

VII. BIOGRAPHIES



Che Guan received his Bachelor degree from Electrics and Information Engineering at Changchun University of Science and Technology in 2004, and Master degree from Mechanical and Electrical Engineering at Chinese Academy of Sciences in 2007. He is currently a graduate student in the Electrical and Computer Engineering Department at the University of Connecticut. He is interested in load forecasting for power systems.

Peter B. Luh received his Ph.D. degree in Applied Mathematics from Harvard University in 1980, and has been with the University of Connecticut since then. Currently he is the SNET Professor of Communications & Information Technologies, and the Head of the Electrical & Computer Engineering Department. He is interested in planning, scheduling, and coordination of design, manufacturing, and supply chain activities; configuration and operation of elevators and HVAC systems for normal and emergency conditions; schedule, auction, portfolio optimization, and load/price forecasting for power systems. He is a Fellow of IEEE, Vice President of Publication Activities for the IEEE Robotics and Automation Society, the founding Editor-in-Chief of the new IEEE Transactions on Automation Science and Engineering (2003-2008), and was the Editor-in-Chief of IEEE Transactions on Robotics and Automation (1999-2003).

Matthew A. Coolbeth graduated with a Bachelor degree in computer science from the University of Connecticut in 2007. He is currently a graduate student at the same university in the department of computer science and engineering.

Yige Zhao (B'07) was born in Liaoning, P. R. China, on September, 1984. He received his bachelor degree in Communication and Electrical Engineering from Beijing Institute of Technology, Beijing, P. R. China in 2007. Currently, he is pursuing his M.S. degree in Electrical Engineering in University of Connecticut, Storrs, CT.

Laurent D. Michel the technical biography is not available.

Ying Chen received the Ph.D degree in electrical and computer engineering from the University of Connecticut, Storrs, in 2008. She is currently with Edison Mission Marketing and Trading, Boston as a risk analyst.

Claude J. Manville is currently an undergraduate student at the University of Connecticut in the department of Computer Science Engineering.

Peter B. Friedland the technical biography is not available.

Stephen J. Rourke the technical biography is not available.