Renewable Energy 37 (2012) 241–249

Contents lists available at ScienceDirect

Renewable Energy

journal homepage: www.elsevier.com/locate/renene

Multi-step forecasting for wind speed using a modified EMD-based artificial neural network model

Zhenhai Guo [a], Weigang Zhao [b,*], Haiyan Lu [c], Jianzhou Wang [b]

[a] State Key Laboratory of Numerical Modeling for Atmospheric Sciences and Geophysical Fluid Dynamics, Institute of Atmospheric Physics, Chinese Academy of Sciences, Beijing 100029, China
[b] School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, China
[c] Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia

ARTICLE INFO

Article history:
Received 28 August 2010
Accepted 14 June 2011
Available online 13 July 2011

Keywords:
Wind speed multi-step forecasting
Empirical mode decomposition
Feed-forward neural network
High frequency
Partial autocorrelation function

ABSTRACT

In this paper, a modified EMD-FNN model (empirical mode decomposition (EMD) based feed-forward neural network (FNN) ensemble learning paradigm) is proposed for wind speed forecasting. The nonlinear and non-stationary original wind speed series is first decomposed into a finite and often small number of intrinsic mode functions (IMFs) and one residual series using EMD technique for a deep insight into the data structure. Then these sub-series except the high frequency are forecasted respectively by FNN whose input variables are selected by using partial autocorrelation function (PACF). Finally, the prediction results of the modeled IMFs and residual series are summed to formulate an ensemble forecast for the original wind speed series. Further more, the developed model shows the best accuracy comparing with basic FNN and unmodified EMD-based FNN through multi-step forecasting the mean monthly and daily wind speed in Zhangye of China.

1. Introduction

Nowadays the demand for electricity grows rapidly as a result of social, economical and industrial development, while reserves of fossil fuel for power generation are being fast depleted and environmental pollution is increasing. This contradiction forces humankind to search for renewable, clean and pollution free energy sources [1]. Wind energy is the one which meets the above requirements. Consequently, in spite of the high cost of wind power over that for fossil fuel, wind power may become an important source of energy in the time to come. Especially for the supply of small electrical loads at remote locations where the wind energy resource is very rich, it is preferable to others for extracting kinetic energy from the wind.

In order to work efficiently on the wind electricity market, it is evident that predicting the wind energy production in the following time is crucial for farm owners and these predictions will help producers take decisions for the sale of energy and thus to increase production and profits. If an accurate prediction of the wind speed for the following time can be evaluated, the total amount of active power that can be produced by each generator on a wind farm can be determined and therefore, the amount of energy that could be sold during the next hour would be known too [2], in conclusion, wind speed forecasting plays a decisive role in wind energy forecasting since the prediction of the wind energy is usually obtained through estimating wind speed.

In general, wind speed variations in each of the time scales have corresponding implications on the operation of the utility system. The short-term variations (time scales of minutes/hours and day(s)), have the largest bearing on utility scale operation while the long-term variations (time scales of month(s) and year(s)), have planning and developmental implications. Under the short-term scales, accurate wind energy forecasts are critical to minimize scheduling errors which impact grid reliability and market based ancillary service costs [3]. And under the long-term scales, those are important for site selection, performance prediction, planning of windmills and the selection of an optimal size of the wind machine for a particular site [4]. Consequently, developing a simple and efficient forecasting method in this study for mean monthly and daily wind speed (represent long-term and short-term scales respectively) is very important for future wind power planning and also crucial for the control, scheduling, maintenance, and resource planning of wind energy conversion systems [5].

Unfortunately, wind speed is considered as one of the most difficult weather parameters to model and forecast due to the complex structures of parameters which affect wind strongly such as topographical properties of the earth, the rotation of the world,

* Corresponding author. Tel.: +86 931 8914050; fax: +86 931 8912481.
E-mail address: zwgstd@gmail.com (W. Zhao).

0960-1481/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.renene.2011.06.023

temperature and pressure difference [5,6]. Moghram and Rahman review five forecasting methods [7]. These are: Multiple linear regression, Time series, General exponential smoothing, State space and Kallman filter and Knowledge-based approach. In their experiment on short-term load forecasting, each method is described briefly and implemented to predict the hourly load of a southeastern utility. Summer loads and winter loads were modeled separately and the results compared in terms of percent error. No one method was determined to be superior. The transfer function approach was the best predictor over the summer months, but was the second worst predictor over the winter months. The authors conclude that because of its strong dependency on historical data, the transfer function approach did not respond well to abrupt changes as did the knowledge based approaches. The conclusion reached is that there is no one best approach for load forecasting [8]. Turn to the more non-stationary wind speed data, the forecasting must also suffer that trouble for the same reason.

In recent years, different wind speed forecasting models have been developed including: linear time series analysis approaches ARIMA models [3,9–12], Kalman filters [13,14] and more recent machine learning techniques, including support vector machines [4] and various forms of neural networks such as multi-layer feed-forward neural networks [1,2,5,6,15] and recurrent neural networks [16–18]. These methods have some improvements, however, most of them suffer from a limitation that they only directly try to let the fitness much closer the original data, through resorting to trial and error approaches which can be tedious and time consuming. But as we all know, traditional statistical and econometric models are built on linear assumptions and they cannot capture the nonlinear patterns hidden in the wind speed series, while the nonlinear models also have their shortcomings and disadvantages: ANN often suffers from local minima and over-fitting; SVM and GP including ANN, are sensitive to parameter selection [19]. Considering the wind speed data are very disorder and unsystematic, we should give a deep insight into the original data besides improving the models themselves. Additionally for multi-layer feed-forward neural networks, there is another limitation that many studies ignore the relationship between the input(s) and output(s): empirically select some previous values of the original series as inputs and make up the sample pair of vector with the current value. This method sometimes leads a poor convergence rate and forecasting precision, moreover, the operability of this method is poor because we cannot determine how many previous values should be selected as the inputs.

Motivated by the thought of [19], the study in this paper develops a modified model, which is modeled through improving the EMD-based artificial neural network in order to remedy the above shortcomings, for wind speed forecasting. And this paper is organized as follows: Section 2 describes the process of EMD technique, the FNN model is described in Section 3. In Section 4, the Modified EMD-based FNN model which is the kernel of this study is shown. Finally in Section 5, used as comparison, the two models, which are the basic FNN and the EMD-FNN without getting rid of IMF1, are proven determining their statistical errors for wind speed multi-step forecasting in order to illustrate the improvement of accuracy and efficiency of the modified model.

2. Empirical mode decomposition (EMD)

As a form of adaptive time series decomposition technique, the empirical mode decomposition (EMD) is relatively easy to understand and implement, whose main idea is to sift the nonlinear and non-stationary time series data using the Hilbert-Huang transform (HHT) until the final data series are stationary. The key innovation embodied in the EMD is the introduction of the intrinsic mode functions (IMFs), which is based on local natural properties of the signal and gives a subsequent meaning to the concept of instantaneous frequency. An IMF can be best defined as a hidden oscillation mode that is embedded in the data series, since it is allowed to be non-stationary and either be amplitude and/or frequency modulation [20]. According to [21], an IMF is defined as a function that satisfies the following two conditions:

1) In the whole data series, the number of extrema (sum of maxima and minima) and thus the number of zero-crossings must be equal or differ at the most by one.
2) At any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

With the above definition for the IMF, we can then decompose any function as follows [22]: identify all the local extrema, then connect all the local maxima with a cubic spline line as the upper envelope and repeat the procedure for the local minima to produce the lower envelope (The upper and lower envelopes should cover all the data between them); Their mean is designated as $m_1(t)$, and the difference between the signal $x(t)$ and $m_1(t)$ is the first component $h_1(t)$; i.e.,

$$h_1(t) = x(t) - m_1(t) \tag{1}$$

Ideally, $h_1(t)$should satisfy the definition of an IMF. However, that is usually not the case since changing a local zero from a rectangular to a curvilinear coordinate system may introduce a new extrema, and further adjustments are required. Therefore, a repeat of the above procedure, the sifting, is necessary [23]. In the subsequent sifting processes, $h_1(t)$can be treated only as a proto-IMF which is treated as the data in the next iteration:

$$h_{11}(t) = h_1(t) - m_{11}(t) \tag{2}$$

The sifting process will be repeated k times, until h_{1k} becomes a true IMF; that is

$$h_{1k}(t) = h_{1(k-1)}(t) - m_{1k}(t) \tag{3}$$

then, it is designated as

$$c_1(t) = h_{1k}(t) \tag{4}$$

Huang et al. also suggested a criterion for stopping the sifting process. This stoppage criterion is determined by using a Cauchy type of convergence test. Specifically, the test requires the normalized squared difference between two successive sifting operations defined as

$$SD_k = \sum_{t=0}^{T} \frac{|h_{1(k-1)}(t) - h_{1k}(t)|^2}{h_{1(k-1)}^2(t)} \tag{5}$$

to be small. According to [21], if SD_k is less than 0.2 or 0.3, then set $c_1(t) = h_{1k}(t)$ and $c_1(t)$ is the first IMF. After test, it is found that SD_k limited to less than 0.2 or 0.3 is very strict and can guarantee the amplitude and frequency of the obtained IMF to have enough physical meaning.

Generally, $c_1(t)$ should contain a component that has the finest scale or the shortest period of the signal. It follows that $c_1(t)$ can be separated from the rest of the data by

$$r_1(t) = x(t) - c_1(t) \tag{6}$$

Then we will obtain the residue $r_1(t)$ which contains a component with a longer period than the previous component. Treating it as a new signal and repeating the same sifting process as described above, we can then obtain the second IMF and residue

($c_2(t)$ and $r_2(t)$). Further more, this procedure can be repeated with all the subsequent $r_j(t)$, and the result is

$$r_i(t) = r_{i-1}(t) - c_i(t), i = 2, 3, \cdots, n \tag{7}$$

The sifting process can be stopped finally by anyone of the following two predetermined criteria [22]: either when the component $c_n(t)$ or residue $r_n(t)$ becomes less than the predetermined value of substantial consequence, or when the residue $r_n(t)$ becomes a monotonic function from which no more IMF can be extracted.

For the final residue $r_n(t)$, even for data with zero mean it still can be different from zero and if the data have a trend it should be that trend. Summing up all the IMFs and the final residue, we can reconstruct the original signal $x(t)$ by

$$x(t) = \sum_{i=1}^{n} c_i(t) + r_n(t) \tag{8}$$

3. Artificial neural networks (ANNs)

Artificial neural networks (ANNs) are a class of typical intelligent learning paradigm, widely used in some practical application domains including: pattern classification, function approximation, optimization, prediction and automatic control and many others [1,19]. In this study, standard multi-layer feed-forward neural network (FNN) is used for modeling the decomposed IMFs and the residual component.

3.1. The structure of FNN

A multi-layer feed-forward neural network consists of an input layer, an output layer, and usually one or more hidden layers. Each of those layers contains nodes, and these nodes are connected to nodes at adjacent layer(s). Associated with each connection is a weight [24]. Fig. 1 demonstrates the structure of FNN.

Each node in FNN is an interconnected identical simple processing unit called neuron. The connection to a neuron has an adjustable weight factor associated with it. Every neuron in the network sums its weighted inputs to produce an internal activity level u_i [1],

$$u_i = \sum_{j=1}^{n} w_{ij}x_{ij} - w_{i0} \tag{9}$$

where w_{ij} is the weight of the connection from input j to neuron i, x_{ij} is input signal number j to neuron i, and w_{i0} is the threshold associated with unit i, called bias of neuron i. The threshold is treated as a normal weight with the input clamped at -1.

Then the internal activity is passed through a nonlinear function f called activation function or transfer function, which is used to transform the output to fall into an acceptable range (between 0 and 1, except linear function) before the output reaches the next layer, to produce the output of the neuron y_i, see Eq. (10). The typical activation function of FNN are logistic sigmoid function (logsig), hyperbolic tangent sigmoid function (tansig) and linear function (purelin), shown in Eq. (11). Fig. 2 shows the single neuron i with typical transfer functions.

$$y_i = f(u_i) \tag{10}$$

$$f(x) = \begin{cases} \dfrac{1}{1+exp(-x)} & \text{logsig,} \\ \dfrac{2}{1+exp(-2x)} - 1 & \text{tansig,} \\ x & \text{purelin.} \end{cases} \tag{11}$$

3.2. The training method of FNN

The connection weights and bias values of FNN are initially chosen as random numbers and then fixed by the results of a training process. Many alternative training processes are available, out of which the present study adopts one popular scheme, namely back-propagation (BP) [25] which is presented in [26].

The following description of BP algorithm profits from the work of [1]. Assume there are P input–output pairs of vectors for FNN. The goal of any training algorithm is to minimize the global (mean sum squared) error E between the real network output $o^{(p)}$ and the desired output $d^{(p)}$; defined below

$$E = \frac{1}{2P}\sum_{p}\sum_{k}\left(d_k^{(p)} - o_k^{(p)}\right)^2 \tag{12}$$

where p is the index of the P training pair of vectors, k is the index of elements in the output vector, $d_k^{(p)}$ and $o_k^{(p)}$ are the kth element of the pth desired output vector and real output vector respectively.

In order to minimize the cost function represented in Eq. (12), it is necessary to adjust the weights of the connections between neurons. And the weight adjustment of the connection between neuron i in layer m and neuron j in layer $m+1$ can be expressed as

$$\Delta w_{ji} = \eta\delta_j^{(p)}o_i^{(p)} \tag{13}$$

where i is the index of units in layer m, η is the learning rate, $o_i^{(p)}$ is the pth output of unit i in the mth layer, and $\delta_j^{(p)}$ is the pth delta error term back-propagated from the jth unit in layer $m+1$ defined by

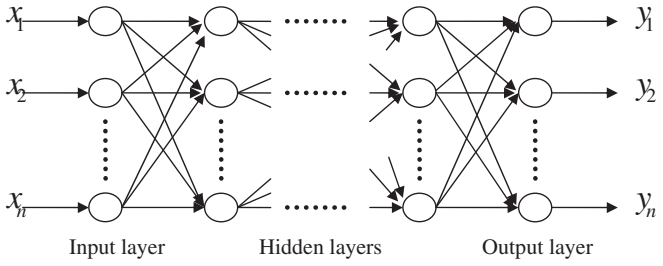

Fig. 1. The multi-layer feed-forward neural network.

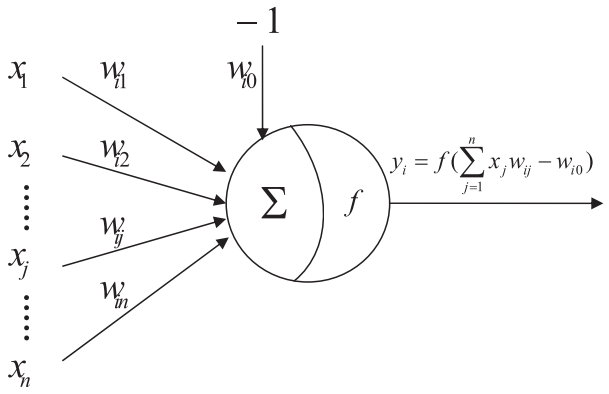

Fig. 2. A single neuron.

$$\delta_j^{(p)} = \left[d_j^{(p)} - o_j^{(p)}\right] o_j^{(p)} \left[1 - o_j^{(p)}\right], \quad \text{if neuron } j \text{ is in the output layer}$$
$$\delta_j^{(p)} = y_j^{(p)} \left[1 - y_j^{(p)}\right] \sum_k \delta_k^{(p)} w_{kj}, \quad \text{if neuron } j \text{ is in a hidden layer}$$

where k is index of neurons in the layer $(m+2)$, ahead of the layer of neuron j. Choosing a small learning rate η leads to slow rate of convergence, and too large η leads to oscillation. A simple method for increasing the rate of learning without oscillation is to include a momentum term as

$$\Delta w_{ji}(n+1) = \eta \delta_j^{(p)} o_i^{(p)} + \alpha \Delta w_{ji}(n) \tag{14}$$

where n is the iteration number, and α is a positive constant which determines the effect of past weight changes on the current direction of movement in weight space. Fig. 3 demonstrates the detailed process of the back-propagation algorithm for training multi-layer feed-forward neural networks.

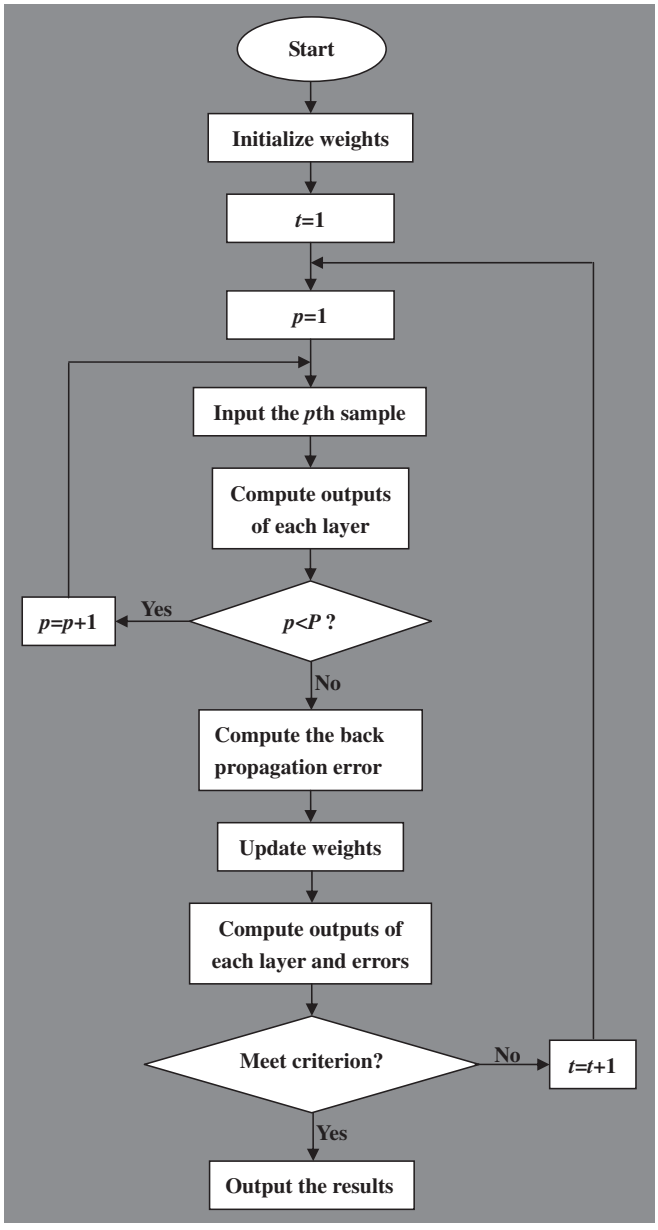

Fig. 3. The flowchart of FNN trained by BP algorithm.

3.3. Some notes

It needs to explain that determining the number of FNN's layers does not exist general rules, nevertheless, it is accepted that a network with three layers connected towards ahead, with a transference function identified in the output unit and a logic function in the units of the intermediate layer, can approximate any continuous function in a reasonable way [15]. So in this paper, we select a three-layer feed-forward neural network, where neurons in the input layer have no transfer function, logsig and purelin were used in the hidden layer and output layer as activation function respectively, for this work. On the other hand, according to Kolmogorovs theorem, Plumb et al. state that a hidden layer of $2n+1$ nodes is sufficient to map any function for n inputs [27], and we adopt this viewpoint in this paper.

What's more, because the amplitude of the original data varies and sometimes even there is a great disparity, if we input these data to FNN for training directly, the large fluctuations of the values will forestall the neural network learning process so that it cannot reflect the small changes in measured values. Therefore, before the network training, the time series is necessary to been normalized as follow

$$x'_k = \frac{x_k - x_{min}}{x_{max} - x_{min}} \tag{15}$$

After simulation, we can re-scale the data through the contrary process of Eq. (15).

4. The modified EMD-based FNN model

For a deep insight into the data structure, the original wind speed series, with characteristics of nonlinearity and non-stationarity, is first decomposed by EMD into a finite and often small number of intrinsic mode functions (IMFs) which are the various frequency sub-series or components extracted from the original series and one residual series (we uniformly call the IMFs along with the residual series as sub-series in the following section). Then these simple sub-series are modeled by FNN models respectively, such that the tendencies of these sub-series can be predicted. Finally, aggregate the prediction results of all sub-series to produce an ensemble fitness and forecasting result for the original wind speed series (this model can be denoted by EMD-FNN).

Our studies showed that the high frequency (named IMF1) is always so small that has little contribution to model fitting, while it sometimes has a great disturbance for the forecasting precision of wind speed. The reason is that IMF1 is the most disorder and unsystematic part of the wind speed series and has little regularity. Although the model fits it well, its forecasting is unpredictable and even so bad. It can be seen that the more disorder and non-stationary the original series is the more irregular its IMF1 will be, which more probably results in worse forecasting precision. In the modified model, considering the smallness of IMF1, we get rid of IMF1 to improve the forecasting precision of the disorder wind speed series.

Further more, in order to overcome the limitation of ignoring the relationship between input(s) and output(s) of FNN, inspired from the identification of parameter p in ARMA (p,q) model (see Eq. (16)), we use the statistics tools, that is the partial autocorrelation function (PACF) and the resulting partial autocorrelation graph which is simply the plots of PACF against the lag length, to determine the input variables of the neural network for orienting on the matter [15]. Concretely, assuming the x_i is the output variable, if the partial autocorrelation at lag k is out of the 95% confidence interval which is $[-1.96/\sqrt{N}, 1.96/\sqrt{N}]$ approximately, x_{i-k} is one of the input variable. It is necessary to note that sometimes all of the PACF

coefficients are in the 95% confidence interval. At this moment we take the previous one value, that is x_{i-1}, as the input variable. In this paper, not only basic FNN but also the FNNs in EMD-FNN and its modified model select inputs through above-mentioned improved method.

$$\varphi(B)x_t = \theta(B)a_t \tag{16}$$

where

$$\begin{cases} \varphi(B) = 1 - \varphi_1 B - \varphi_2 B^2 - \cdots - \varphi_p B^p \\ \theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q \end{cases}$$

Here, the description of PACF is as follows [28]. For a time series $\{w_1, w_2, \ldots, w_n\}$, the Covariance at lag k (if $k = 0$, it is the Variance), denoted by γ_k, is estimated in Eq. (17).

$$\hat{\gamma}_k = \frac{1}{n}\sum_{i=1}^{n-k}(w_i - \overline{w})(w_{i+k} - \overline{w}), k = 0, 1, \ldots, M \tag{17}$$

where $\overline{w}$ is the mean of the series, $M = n/4$ is the maximum lag. Obviously, $\hat{\gamma}_{-k} = \hat{\gamma}_k$.

Then the ACF (autocorrelation function) at lag k, denoted by ρ_k, can be estimated according to Eq. (18).

$$\hat{\rho}_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0} \tag{18}$$

Based on the Covariance and the resulting ACF, we present the calculation for the PACF at lag k, denoted by $\alpha_{kk,}$ as follows

$$\left.\begin{aligned} \hat{\alpha}_{11} &= \hat{\rho}_1 \\ \hat{\alpha}_{k+1,k+1} &= \frac{\hat{\rho}_{k+1} - \sum_{j=1}^{k}\hat{\rho}_{k+1-j}\hat{\alpha}_{kj}}{1 - \sum_{j=1}^{k}\hat{\rho}_j\hat{\alpha}_{kj}} \\ \hat{\alpha}_{k+1,j} &= \hat{\alpha}_{kj} - \hat{\alpha}_{k+1,k+1}\cdot\hat{\alpha}_{k,k-j+1} \\ & (j = 1, 2, \ldots, k) \end{aligned}\right\} \tag{19}$$

where $k = 1, 2, \ldots, M$.

In addition, we simply mark the model which has above two improvements for EMD-FNN as M-EMDFNN where "M" represents "Modified". Depending on the previous techniques and methods, the M-EMDFNN ensemble paradigm is illustrated in Fig. 4.

5. Simulation

Northwest China has a vast open land, hence offers good opportunities for harnessing the power of wind. Meanwhile Chinese government has formulated a number of preferential policies for the development of wind electricity in this area, so case studies are carried out using wind speed data from Gansu Province of this area in this section.

We collect the mean monthly wind speed data in years of 2003 ~ 2006 and mean daily wind speed data from May to Aug in 2006 of Zhangye which is located in Gansu Province. Fig. 5 shows the mean monthly data and Fig. 6 presents the mean daily values. Then we implement multi-step forecasting for the mean monthly and daily wind speed using the above-mentioned three models (FNN, EMD-FNN and M-EMDFNN) respectively.

5.1. Statistics measures to determine the accuracy of the forecast

In order to quantitatively determine the best model, three criteria are used for evaluating the proposed models. They are the

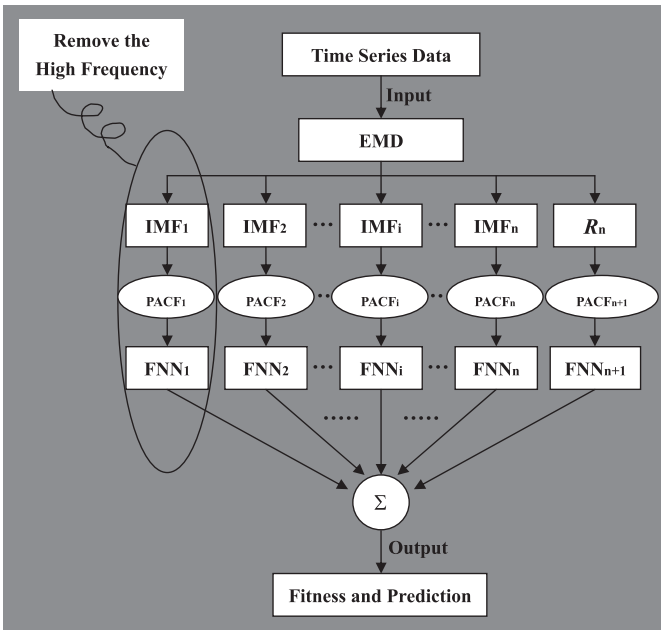

Fig. 4. The overall process of the M-EMDFNN ensemble methodology.

mean square error (*MSE*), mean absolute error (*MAE*) and mean absolute percentage error (*MAPE*) which are calculated as

$$MSE = \frac{1}{n}\sum_{t=1}^{n} e_t^2 \tag{20}$$

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |e_t| \tag{21}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \left|\frac{e_t}{x_t}\right| \times 100\% \tag{22}$$

where n is the number of periods of time, $e_t = x_t - \hat{x}_t$ and x_t is the actual observation for the time period t, $\hat{x}_t$ is the forecast for the same period. Currently, the wind speed forecasting *MAPEs* are ranging from 25% to 40%, which is not only concerned with the forecasting methods, but also with the forecasting horizon and the

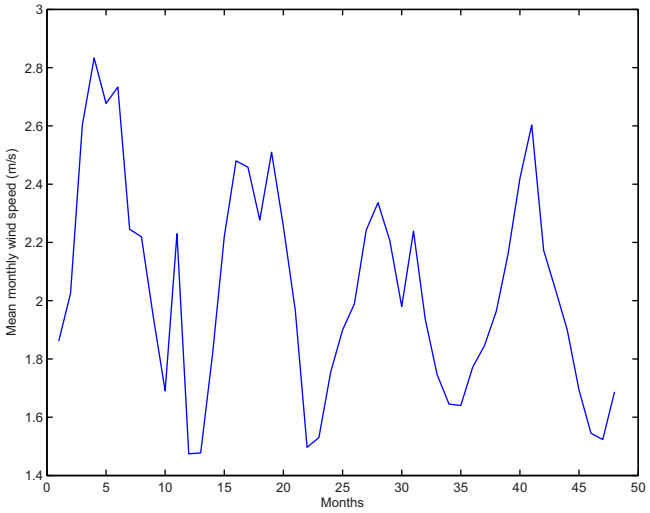

Fig. 5. Mean monthly wind speed during the period from 2003 to 2006 for Zhangye.

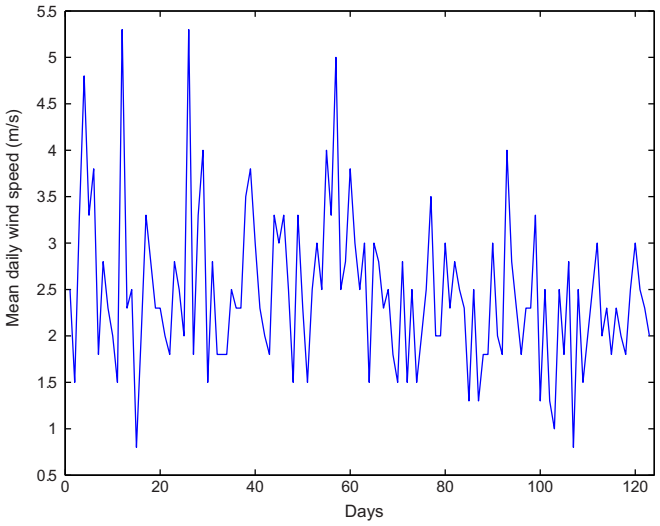

Fig. 6. Mean daily wind speed during the period from May to Aug 2006 for Zhangye.

characteristics of the wind speed in a location. In general, the shorter the forecasting horizon is or the more relaxative the wind speed variation is, the smaller the forecasting error will be, otherwise, the forecasting error will be larger [29].

5.2. Multi-step forecasting

The so-called multi-step forecasting (predicting more than one day in advance) is an iterative process where the output of the system is fed again as input, in other words, it takes the previous forecasting data instead of the true values to predict the next values when there are no previous true values. And it is commonly used for short forecast horizons [15].

In this section, for the mean monthly wind speed forecasting, we take the data from January 2003 to December 2005 with a total of 36 observations as the training set and the remainder is used as the testing set (12 observations which are from January to December, 2006) for multi-forecasting. For the mean daily wind speed forecasting, we take the data from May 1 to August 21, 2006 with a total of 113 observations as the training set and the remainder is used as the testing set (10 observations which are from August 22 to 31, 2006).

We first take the mean monthly wind speed for multi-forecasting. Using EMD technique, the original series is first decomposed into several independent IMFs and one residue. In the decomposing process a monotonic residue cannot be obtained, so it is stopped until the residue is less than a predetermined value (0.01 in this paper) according to the first sifting stoppage criterion in Section 2. As a result, the original series is decomposed into five IMFs and one residue which can be ignored since it is so small (equals zero approximately, between -10^{-15} and 10^{-15}) that computers regard it as zero in the modeling process which can only obtain zero forecasting results. Then we get the graphical representations of these decomposed sub-series, as illustrated in Fig. 7. Conveniently, after normalizing these series by Eq. (15), we can get the partial autocorrelogram of these normalized IMFs and the normalized original series which are shown in Fig. 8 where PACF0 stands for PACF of the normalized original series and the others (PACF1 ~ PACF5) stand for PACF of each normalized IMF respectively. According to the input selection method shown in Section 4, through observing Fig. 8, with the output variable x_t, it is obvious that the input variables of these six normalized series for FNN modeling are as follows

- Original: (x_{t-1});
- IMF1: (x_{t-1});
- IMF2: $(x_{t-3}, x_{t-2}, x_{t-1})$;
- IMF3: (x_{t-2}, x_{t-1});

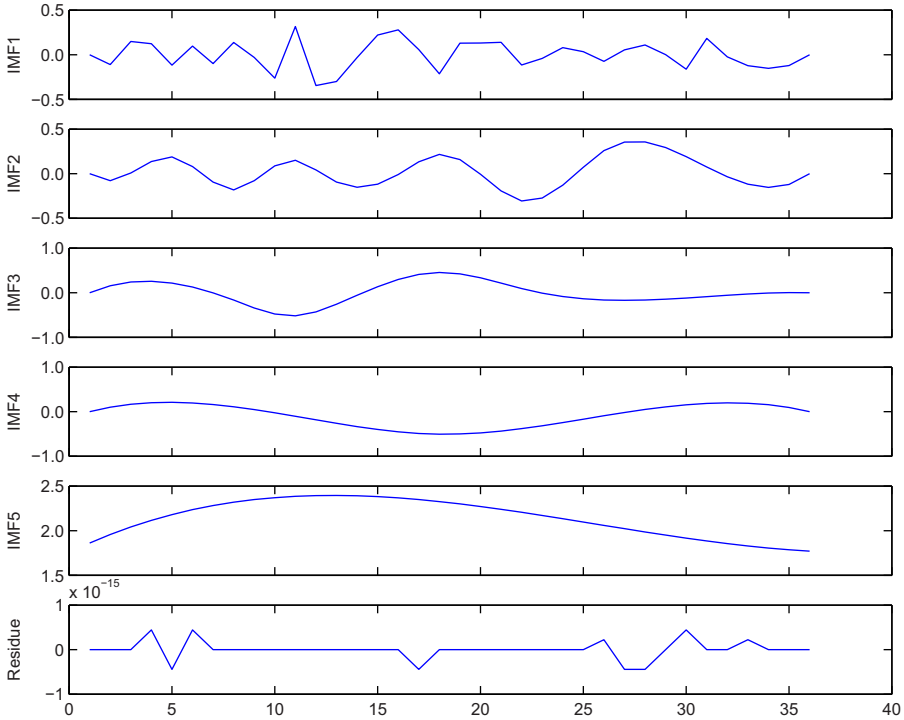

Fig. 7. The decomposition of the mean monthly wind speed during the period from 2003 to 2005 for Zhangye.

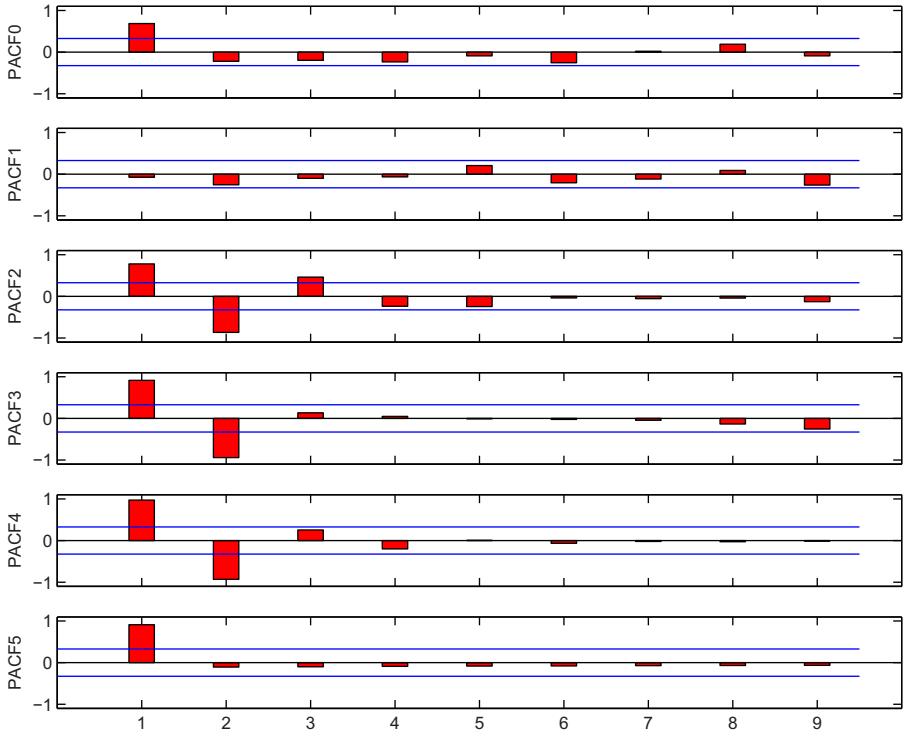

Fig. 8. The PACFs of the normalized original series and normalized IMFs of the mean monthly wind speed during the period from 2003 to 2005 for Zhangye.

- IMF4: (x_{t-2}, x_{t-1});
- IMF5: (x_{t-1}).

where the series of $\{x_i\}$ represents those six normalized series respectively. In the iterative process of multi-step forecasting, x_i represents the corresponding forecasting value of each series when i exceeds the length of the series.

After simulating those six normalized series by FNN method and re-scaling them, we obtain the simulation of these five IMFs and the original series. Obviously, the basic FNN model for original series has been completed. Meanwhile we aggregate the prediction results of IMF1 ~ IMF5 to produce an ensemble fitness and forecasting result, which is the EMD-FNN modeling process. Furthermore the M-EMDFNN model is similar to EMD-FNN, which only aggregates the results of IMF2 ~ IMF5. And the evaluation of prediction results obtaining from those three models for the mean monthly wind speed is presented in Table 1.

For the mean daily wind speed, in the same way of monthly case, we can decompose the original series to seven IMFs with one residue which are shown in Fig. 9 where the residue can also be ignored for equaling zero approximately, and then obtain the PACFs of the normalized original series and the normalized IMFs (see Fig. 10), so the inputs of those eight normalized series for FNN modeling are as follows

- Original: (x_{t-9});
- IMF1: $(x_{t-17}, x_{t-8}, x_{t-2}, x_{t-1})$;
- IMF2: $(x_{t-6}, x_{t-4}, x_{t-3}, x_{t-2}, x_{t-1})$;
- IMF3: $(x_{t-3}, x_{t-2}, x_{t-1})$;
- IMF4: $(x_{t-4}, x_{t-3}, x_{t-2}, x_{t-1})$;
- IMF5: (x_{t-2}, x_{t-1});
- IMF6: $(x_{t-4}, x_{t-3}, x_{t-2}, x_{t-1})$;
- IMF7: (x_{t-1}).

Through the same process above-mentioned, with the simulation results of those eight normalized series by FNN method and resulting re-scaled results, the basic FNN model, EMD-FNN model and M-EMDFNN for original series can be completed. Respectively, their results are the fitness and forecasting results of original series, the aggregation of IMF1 ~ IMF7 and the aggregation of IMF2 ~ IMF7, which are presented in Table 2.

5.3. Comparative analysis

From Table 1, we can see that FNN has high forecasting precision in the front eight steps but has very bad results after the 9th step forecasting. Quantitatively, FNN's averaged relative error of the first half of the year is 6.53% which is lower than 13.82% of EMD-FNN and 13.43% of M-EMDFNN, but this figure of

Table 1
Compare the performance of the above-mentioned three models for mean monthly wind speed.

Month	Actual value	FNN	R.E. (%)	EMD-FNN	R.E. (%)	Modified	R.E. (%)
Jan.	1.8452	1.9117	3.6	2.2018	19.32	1.9184	3.97
Feb.	1.9643	2.0113	2.39	1.8157	7.57	2.0082	2.24
Mar.	2.1613	2.0809	3.72	2.0807	3.73	2.0246	6.33
Apr.	2.42	2.129	12.02	1.9381	19.91	1.962	18.93
May	2.6032	2.162	16.95	2.0944	19.55	1.8499	28.94
Jun.	2.1733	2.1845	0.52	1.8946	12.82	1.7357	20.14
Averaged R.E.			**6.53**		**13.82**		**13.43**
Jul.	2.0387	2.1998	7.9	1.6214	20.47	1.6648	18.34
Aug.	1.9	2.2101	16.32	1.3313	29.93	1.6646	12.39
Sep.	1.6933	2.217	30.93	1.9322	14.11	1.734	2.41
Oct.	1.5452	2.2217	43.78	1.792	15.97	1.8432	19.28
Nov.	1.5233	2.2248	46.05	2.0071	31.76	1.9452	27.69
Dec.	1.6871	2.227	32	2.1766	29.02	1.9979	18.42
Averaged R.E.			**29.50**		**23.54**		**16.42**
MSE		0.1608		0.1511		0.1296	
MAE		0.3208		0.3583		0.2986	
MAPE		18.02%		18.68%		14.92%	

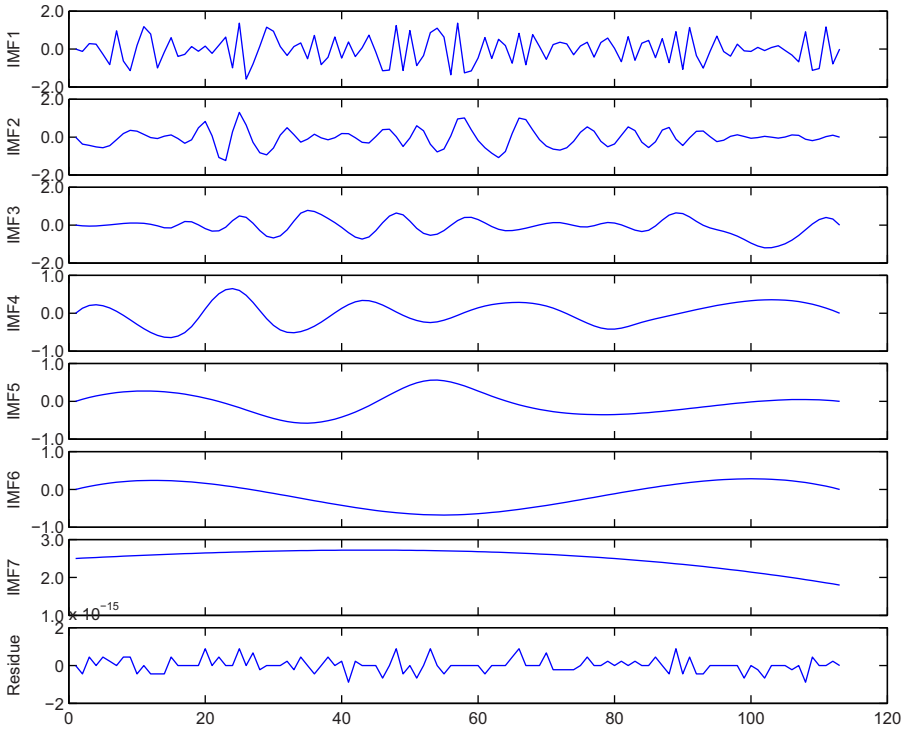

Fig. 9. The decomposition of the mean daily wind speed during the period from May 1 to August 21 2006 for Zhangye.

the second half of the year turns to 29.5% which is higher than 23.54% and 16.42% which are the figures of EMD-FNN and M-EMDFNN respectively. That indicates FNN is not proper for multi-step forecasting for a poor stability. Relatively, EMD-FNN has a better stability in multi-step forecasting but has little improvement in the holistic performance with the MSE reducing by 0.0097 but the MAE and MAPE increasing by 0.0375 and 0.66% respectively. The best performance belongs to M-EMDFNN, which produces better results than such two models in terms of lower MSE, MAE and MAPE. Besides, with the similar errors of the first and second half of the year, the modified model has a good forecast stability as well.

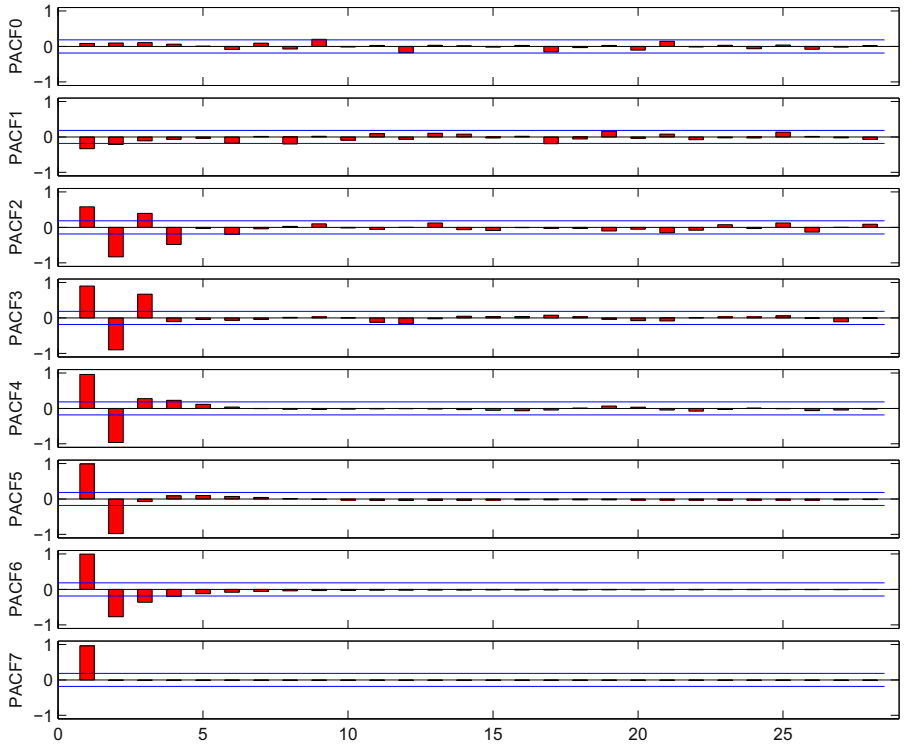

Fig. 10. The PACFs of the normalized original series and normalized IMFs of the mean daily wind speed during the period from May 1 to August 21 2006 for Zhangye.

Table 2
Compare the performance of the above-mentioned three models for mean daily wind speed.

Date	Actual value	FNN	R.E. (%)	EMD-FNN	R.E. (%)	Modified	R.E. (%)
22nd	1.7	2.2032	29.6	1.9052	12.07	1.646	3.18
23rd	1.6	2.2032	37.7	1.4285	10.72	1.4229	11.07
24th	1.7	2.1919	28.94	1.1406	32.91	1.4187	16.55
25th	2	2.2732	13.66	1.8298	8.51	1.4177	29.12
26th	2	2.1745	8.73	1.5241	23.79	1.4327	28.37
Averaged R.E.			**23.73**		**17.60**		**17.66**
27th	1.6	2.1919	37	1.001	37.44	1.4099	11.88
28th	1.5	2.3022	53.48	1.7934	19.56	1.442	3.87
29th	2.9	2.2032	24.03	1.3851	52.24	1.4945	48.47
30th	1.5	2.2141	47.61	1.4667	2.22	1.5556	3.71
31st	1.9	2.2352	17.64	1.6647	12.38	1.5825	16.71
Averaged R.E.			**35.95**		**24.77**		**16.93**
MSE		0.3066		0.3436		0.2893	
MAE		0.5186		0.4258		0.3689	
MAPE		29.84%		21.18%		17.29%	

With the obvious non-stationarity of the daily data (see Fig. 6), the advantage of EMD technique is shown perfectly in the multi-step forecasting for mean daily wind speed. From Table 2, we can see the forecasting precision of EMD-FNN has a great improvement to FNN where the MAE and MAPE reduce by 0.0928 and 8.76% respectively though the MSE is a little high. Especially, the precision of the first and second half of these days are both improved observably with the averaged relative error reducing by 6.13% and 11.18% respectively. Further more, the M-EMDFNN is the best model absolutely with reducing MSE, MAE and MAPE by 0.0173, 0.1497 and 12.55% respectively compared with FNN. And its two averaged relative errors are also reduced observably. Especially, The gap of them is only 0.73 (16.93 subtracted from 17.66) percentage point which is much lower than 12.22 of FNN and 7.17 of EMD-FNN. We can say M-EMDFNN is the most stability model of the three models.

It should be pointed out that the magnitude of forecasting precision improvement from EMD-FNN to M-EMDFNN for mean daily wind speed is larger than that for mean monthly wind speed, which proves that the more irregular IMF1 (here it is the IMF1 of the mean daily wind speed) will more probably result in worse forecasting precision.

6. Conclusions

This paper develops a modified EMD-based FNN technique for wind speed prediction. In terms of empirical results, we find that across different forecasting models, for the two kinds of wind speed series - mean monthly wind speed and mean daily wind speed - in terms of different criteria (MSE, MAE and MAPE), the M-EMDFNN model performs the best. In both testing cases, the MSE, MAE and MAPE are the lowest, indicating that the M-EMDFNN forecasting paradigm can be used as a very promising methodology for wind speed prediction under both long-term and short-term scales. With the significant improvement and the good stability of wind speed forecasting precision, the developed model is an efficient tool for operation, planning and dispatching of wind farms. In addition, no matter how complex the calculation of FNN is, the computation time is negligible for the three models with the advancing computers.

Acknowledgements

This research was supported by the National Basic Research Program of China (Grant No. 2009CB421402).

References

[1] Mohandes MA, Rehman S, Halawani TO. A neural networks approach for wind speed prediction. Renewable Energy 1998;13(3):345–54.
[2] Flores P, Tapia A, Tapia G. Application of a control algorithm for wind speed prediction and active power generation. Renewable Energy 2005; 30:523–36.
[3] Kavasseri RG, Seetharaman K. Day-ahead wind speed forecasting using f-ARIMA models. Renewable Energy 2009;34:1388–93.
[4] Mohandes M, Halawani T, Rehman S, Hussain AA. Support vector machines for wind speed prediction. Renewable Energy 2004;29:939–47.
[5] Abdel-Aal R, Elhadidy M, Shaahid S. Modeling and forecasting the mean hourly wind speed time series using GMDH-based abductive networks. Renewable Energy 2009;34:1686–99.
[6] Bilgili M, Sahin B, Yasar A. Application of artificial neural networks for the wind speed prediction of target station using reference stations data. Renewable Energy 2007;32:2350–60.
[7] Moghram I, Rahman S. Analysis and evaluation of five short-term load forecasting techniques. IEEE Transactions on Power Systems 1989;4(4): 1484–91.
[8] Loredo EN. Annual electrical peak load forecasting methods with measures of prediction error, engineering, industrial. Arizona State University; 12 2002.
[9] Torres J, García A, Blas MD, Francisco AD. Forecast of hourly average wind speed with ARMA models in Navarre (Spain). Solar Energy 2005;79: 65–77.
[10] Poggi P, Muselli M, Notton G, Cristofari C, Louche A. Forecasting and simulating wind speed in Corsica by using an autoregressive model. Energy Conversion and Management 2003;44:3177–96.
[11] Kamal L, Jafri YZ. Time series models to simulate and forecast hourly averaged wind speed in Quetta, Pakistan. Solar Energy 1997;61(1):23–32.
[12] Huang Z, Chalabi Z. Use of time-series analysis to model and forecast wind speed. Journal of Wind Engineering and Industrial Aerodynamics 1995;56:311–22.
[13] Malmberg A, Holst U, Holst J. Forecasting near-surface ocean winds with Kalman filter techniques. Ocean Engineering 2005;32:273–91.
[14] Louka P, Galanis G, Siebert N, Kariniotakis G, Katsafados P, Pytharoulis I, et al. Improvements in wind speed forecasts for wind power prediction purposes using Kalman filtering. Journal of Wind Engineering and Industrial Aerodynamics 2008;96:2348–62.
[15] Cadenas E, Rivera W. Short term wind speed forecasting in La Venta, Oaxaca, México, using artificial neural networks. Renewable Energy 2009;34:274–8.
[16] Barbounis T, Theocharis J. A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation. Neurocomputing 2007;70:1525–42.
[17] Barbounis T, Theocharis J. Locally recurrent neural networks for long-term wind speed and power prediction. Neurocomputing 2006;69:466–96.
[18] Barbounis T, Theocharis J. Locally recurrent neural networks for wind speed prediction using spatial correlation. Information Sciences 2007;177: 5775–97.
[19] Yu L, Wang S, Lai KK. Forecasting crude oil price with an EMD-based neural network ensemble learning paradigm. Energy Economics 2008;30: 2623–35.
[20] Dätig M, Schlurmann T. Performance and limitations of the Hilbert-Huang transformation (HHT) with an application to irregular water waves. Ocean Engineering 2004;31:1783–834.
[21] Huang NE, Shen Z, Long SR, Wu MC, Shih HH, Zheng Q, et al. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. Proceedings of the Royal Society A: Mathematical, Physical & Engineering Sciences 1998;454:903–95.
[22] Qi K, He Z, Zi Y. Cosine window-based boundary processing method for EMD and its application in rubbing fault diagnosis. Mechanical Systems and Signal Processing 2007;21:2750–60.
[23] Wei Y-C, Lee C-J, Hung W-Y, Chen H-T. Application of Hilbert-Huang transform to characterize soil liquefaction and quay wall seismic responses modeled in centrifuge shaking-table tests. Soil Dynamics and Earthquake Engineering 2010;30:614–29.
[24] Law R. Back-propagation learning in improving the accuracy of neural network-based tourism demand forecasting. Tourism Management 2000;21: 331–40.
[25] More A, Deo M. Forecasting wind with neural networks. Marine Structures 2003;16:35–49.
[26] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors, vol. 323. Nature Publishing Group; 1986. 9, 533–536.
[27] Plumb AP, Rowe RC, York P, Brown M. Optimisation of the predictive ability of artificial neural network (ANN) models: a comparison of three ANN programs and four classes of training algorithm. European Journal of Pharmaceutical Sciences 2005;25:395–405.
[28] Wang H, Zhao W. ARIMA model estimated by Particle Swarm optimization algorithm for Consumer price index forecasting, Lecture notes in computer Science. Artificial Intelligence and Computational Intelligence 2009;5855: 48–58.
[29] Yang X, Xiao Y, Chen S. Wind speed and generated power forecasting in wind farm (in Chinese). Proceedings of the CSEE 2005;25(11):1–5.