Research Article

Short-Term Wind Speed Forecasting Using the Data Processing Approach and the Support Vector Machine Model Optimized by the Improved Cuckoo Search Parameter Estimation Algorithm

Chen Wang,¹ Jie Wu,² Jianzhou Wang,³ and Zhongjin Hu¹

¹School of Mathematics & Statistics, Lanzhou University, Lanzhou 730000, China

²School of Mathematics and Computer Science, Northwest University for Nationalities, Lanzhou 730030, China

³School of Statistics, Dongbei University of Finance and Economics, Dalian 116025, China

Correspondence should be addressed to Jie Wu; 13519619146@126.com

Received 28 October 2015; Revised 12 February 2016; Accepted 11 May 2016

Academic Editor: Vida Maliene

Copyright © 2016 Chen Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Power systems could be at risk when the power-grid collapse accident occurs. As a clean and renewable resource, wind energy plays an increasingly vital role in reducing air pollution and wind power generation becomes an important way to produce electrical power. Therefore, accurate wind power and wind speed forecasting are in need. In this research, a novel short-term wind speed forecasting portfolio has been proposed using the following three procedures: (I) data preprocessing: apart from the regular normalization preprocessing, the data are preprocessed through empirical model decomposition (EMD), which reduces the effect of noise on the wind speed data; (II) artificially intelligent parameter optimization introduction: the unknown parameters in the support vector machine (SVM) model are optimized by the cuckoo search (CS) algorithm; (III) parameter optimization approach modification: an improved parameter optimization approach, called the SDCS model, based on the CS algorithm and the steepest descent (SD) method is proposed. The comparison results show that the simple and effective portfolio EMD-SDCS-SVM produces promising predictions and has better performance than the individual forecasting components, with very small root mean squared errors and mean absolute percentage errors.

1. Introduction

The demand for clean and renewable energy resources has increased significantly since the acid emissions and air pollution caused by burning fossil fuels have heavily polluted the world environment. As a clean and renewable resource, wind energy plays an increasingly vital role in energy supply and wind power generation becomes an important way to generate electrical power. However, the stochastic fluctuation of wind makes it problematic to forecast [1–3]. Therefore, effort to improve the accuracy of wind speed forecasting continues so as to lower the possibility of the power-grid collapse accident occurrence.

Wind speed forecasting is an important foundation and prerequisite for the prediction of wind power generation. The more accurate wind speed forecasting result can reduce wind rotating equipment and operation cost and improve limitation of wind power penetration. At the same time the precise

prediction of wind speed helps dispatching department timely adjustments to the program, so as to reduce the impact of wind power on the grid and effectively avoid the adverse effect of wind farm on the power system, enhancing the competitiveness of wind power in the electricity market.

In literature studies, statistically based and neural network-based methods are two models pervasively used to forecast the wind speed [4–7]. With the development of artificial intelligent techniques, some artificial intelligent methods have been presented, such as Artificial Neural Networks, fuzzy logic methods, and support vector machine. Guo et al. [8] presented a wind speed strategy based on the chaotic time series modeling technique and the Apriori algorithm. Barbounis et al. [9] employed three different types of neural network (NN) models to forecast the hourly wind speed (up to 3 days) in a wind park located on the Greek island of Crete. However, there are several unknown parameters in the NN model. Thus, many researchers have indicated the need to

optimize the parameters in the NN model to improve wind speed forecasting accuracy. Wang and Hu [10] improved the performance of the back propagation (BP) NN model in the wind speed forecasting field by optimizing the parameters in the BP model. Both models, that is, the statistical and the NN-based models, have been used by Azad et al. [11] to solve the long-term wind speed forecasting problem for two stations in Malaysia. However, wind speed forecasting results obtained by the neural network models are not always superior to those obtained by other models. Chen and Yu [12] developed a new model by integrating the unscented Kalman filter with the support vector regression-based state-space model. Comparison results indicated that the new proposed model outperforms the NN model. Apart from the NN models, the parameter optimization strategy has also been applied to other wind speed forecasting models. Gani et al. [13] proposed that firefly algorithm combines with SVM algorithm for a problem of short-term wind speed forecast, where firefly algorithm is used to optimize the parameters of SVMs and successfully obtain the accuracy forecasting result. Compared with artificial intelligent models, statistical approaches are less expensive and intrusive and, hence, more practical in forecasting wind power generation. Statistical models are widely used to forecast model for short-term wind forecasting, predicting wind conditions several hours in advance, which is particularly useful for wind power generation [14]. But for the nonlinear wind speed time series is often not satisfactory, especially in multistep prediction, and the error will be significantly increased with the extension of the prediction time. The new paradigm of big data stream mobile computing is quickly gaining momentum [15], while wind speed forecasting results have been applied to many different areas [16].

It is found that the existing wind speed forecasting models have the following disadvantages: (1) some of the existing models have taken no account of the randomness, instability, and the large fluctuation of the wind speed data, which may lead to a high forecasting error. Therefore, in this research, a model based on the ensemble empirical mode decomposition (EEMD) technique is utilized to adaptively decompose the original wind speed data into a finite number of intrinsic mode functions with a similarity property to modeling. (2) The existing traditional parameter estimation methods, such as the moment estimation or the likelihood estimation, are not dynamic and need to solve some equations with a great deal of calculations. Therefore, the artificial intelligent parameter estimation method named the cuckoo search (CS) algorithm is used in this paper to estimate the unknown parameters in the forecasting model. (3) Though some researchers applied the artificial intelligent parameter estimation approaches to the parameter estimation, they just adopted the original approach without considering the deficiency of the approach. Thus, in this paper, the steepest descent (SD) method is used to optimize the CS algorithm so as to enhance the convergence rate. Based on the above motivations, in this research, a new short-term wind speed forecasting portfolio which not only can maintain the characteristics of the wind speed data but can also automatically estimate the unknown parameters in the forecasting model with a considerable convergence rate has been proposed through

the following three procedures: (I) data preprocessing: apart from regular normalization preprocessing, the data are preprocessed through the EMD model, which reduces the effect of the noise on the wind speed data; (II) artificially intelligent parameter optimization introduction: the unknown parameters in the support vector machine (SVM) model are optimized by the cuckoo search (CS) algorithm; (III) parameter optimization approach modification: although the original CS algorithm is simple and efficient, it has disadvantages such as insufficient search vigor and slow search speed during the latter part of the search. Therefore, this paper proposes an improved parameter optimization approach based on the CS algorithm and the steepest descent (SD) method, which is abbreviated as the SDCS model. The performance of the developed EMD-SDCS-SVM model has been compared with those obtained by the individual forecasting components using the following two error evaluation criteria: the root mean squared error and the mean absolute percentage error.

The paper is organized as follows: Section 2 introduces related methodologies, Section 3 presents the simulation examples and discussions, and the last section presents concluding remarks.

2. Related Methodologies

2.1. Data Preprocessing Approach. Data preprocessing is a common way to improve forecasting accuracy, especially for data with high noise and different scales. This paper focuses on handling these two problems by using the EMD model and the normalization preprocessing approach, respectively.

2.1.1. Empirical Mode Decomposition Model. The EMD model is an adaptive decomposition approach proposed by Baccarelli et al. [15]. It is used in a wide range of applications, especially in dealing with nonlinear time series. The EMD model decomposes the original time series into several different sequences with different scales (also called the intrinsic mode function (IMF)) as well as a residual sequence. All IMFs must satisfy two requirements:

- (a) The number of extreme points (all maximum and minimum points are included) must be equal to the number of zero crossings or differ by no more than one.
- (b) In all cases, the average of the envelopes defined by the local maxima and minima must be zero.

With the above two limitations, a signal sequence $x(t)$ can be decomposed with the assistance of the EMD method [16] through the following steps.

Step 1. Calculate all the local extrema (including all the minimum and maximum values).

Step 2. Connect the local maxima by a cubic spline line to generate the upper envelope and similarly produce the lower envelope by connecting all the local minima with a cubic spline interpolation, represented by e_{upper} and e_{lower} , respectively.

Step 3. Calculate the average value of the two envelopes m_1 by

$$m_1 = \frac{(e_{\text{upper}} + e_{\text{lower}})}{2}. \quad (1)$$

Step 4. Calculate the difference (h_1) between the data and m_1 by

$$h_1 = x(t) - m_1. \quad (2)$$

Step 5. Judge whether h_1 satisfies the two requirements of the IMFs. If not, regard h_1 as the original signal sequence; then $h_{11} = h_1 - m_{11}$. Repeat this process k times until h_{1k} which is calculated by $h_{1k} = h_{1(k-1)} - m_{1k}$ is an IMF. The first IMF sequence is obtained by

$$\text{IMF}_1 = h_{1k}. \quad (3)$$

Step 6. Calculate the first residual sequence according to

$$r_1 = x(t) - \text{IMF}_1. \quad (4)$$

Step 7. Regard r_1 as the raw data and return to Step 1 to repeat this procedure unless the final residue r_n turns into either a monotonic function or a function from which no more IMF sequences can be extracted.

Finally, the original signal sequence is decomposed into

$$x(t) = \sum_{i=1}^n \text{IMF}_i + r_n. \quad (5)$$

2.1.2. Normalization Preprocessing. To improve the training efficiency and the generalization ability of the SVM model, normalization preprocessing is used to address the IMF sequences obtained by the SVM model. Normalization preprocessing is defined as follows:

$$x_{\text{processed}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \quad (6)$$

where x and $x_{\text{processed}}$ represent the original data sequence and the preprocessed data sequence, respectively, and $x_{\min}$ and $x_{\max}$ denote the minimum and the maximum data in the original data sequence, respectively.

2.2. Support Vector Machine Model. The SVM model is the core of statistical machine learning theories. It can surmount difficulties that appear in the traditional machine learning methods, such as the curse of dimensionality, easily falling into local optima and overlearning. In addition, it has great generalization ability [17]. Therefore, the SVM model has long been an attractive tool with powerful capabilities in solving classification and regression problems. In this paper, we mainly focus on the SVM model for regression.

Suppose that there are m in-sample data points (or training samples) $(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)$ where $x_i \in R^n$ denotes the input vector and $y_i \in R$ is the targeted output corresponding to the input vector x_i . The main purpose of the SVM for regression is to find a function $f(x)$ which satisfies

the following two requirements: (a) the deviation between $f(x_i)$ and y_i is no greater than a given positive real number ε , for all $i = 1, 2, \dots, m$, and (b) $f(x)$ is as flat as possible. In the SVM algorithm, f is defined by the formula

$$f(x) = \omega^T \phi(x) + b, \quad (7)$$

where $\phi : R^n \rightarrow F$ is a nonlinear mapping, b is the threshold value, and the unknown coefficients ω and b can be estimated by solving the following optimization problem:

$$\min_{\omega, b, \xi, \xi^*} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^m (\xi_i + \xi_i^*) \quad (8)$$

$$\text{subject to: } \begin{aligned} [\omega^T \phi(x) + b] - y_i &\leq \varepsilon + \xi_i \\ y_i - [\omega^T \phi(x) + b] &\leq \varepsilon + \xi_i^* \end{aligned} \quad (9)$$

$$\xi_i, \xi_i^* \geq 0, \quad i = 1, 2, \dots, m,$$

where C denotes the penalty coefficient, ξ_i and ξ_i^* are two slack variables, and ε is the tube size. Problem (8) can be solved by introducing two Lagrange multipliers α_i and α_i^* and minimizing the following Lagrange function L [14]:

$$\begin{aligned} L &= -\frac{1}{2} \sum_{i,j=1}^m (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) \kappa(x_i, x_j) \\ &\quad - \varepsilon \sum_{i=1}^m (\alpha_i + \alpha_i^*) + \sum_{i=1}^m y_i (\alpha_i - \alpha_i^*) \end{aligned} \quad (10)$$

$$\begin{aligned} L &= -\frac{1}{2} \sum_{i,j=1}^m (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) \kappa(x_i, x_j) \\ &\quad - \varepsilon \sum_{i=1}^m (\alpha_i + \alpha_i^*) + \sum_{i=1}^m y_i (\alpha_i - \alpha_i^*) \end{aligned}$$

subject to

$$\sum_{i=1}^m (\alpha_i - \alpha_i^*) = 0, \quad 0 \leq \alpha_i, \alpha_i^* \leq C. \quad (11)$$

This calculation results in

$$\omega = \sum_{i=1}^m (\alpha_i - \alpha_i^*) \phi(x_i), \quad (12)$$

$$f(x) = \sum_{i=1}^m (\alpha_i - \alpha_i^*) \kappa(x_i, x) + b,$$

where $\kappa(\cdot, \cdot)$ is called the kernel function. The following four types of kernel functions are commonly used [18, 19]: (a) linear kernel function: $\kappa(x, y) = x^T y$, (b) polynomial kernel function: $\kappa(x, y) = (x^T y + 1)^d$, (c) sigmoid kernel function: $\kappa(x, y) = \tanh(ax^T y + c)$, and (d) Gaussian kernel function: $\kappa(x, y) = \exp(-\|x - y\|^2 / 2\sigma^2)$, where a, c, d , and σ are kernel parameters.

```

(1) Determine the objective function  $g(x)$ ,  $x = (x_1, x_2, \dots, x_d)^T$ ;
(2) Initialize the population of  $N$  host nests  $x_i$  ( $i = 1, 2, \dots, N$ );
(3) WHILE (The stop criterion has not met)
(4)   Choose a cuckoo randomly by Lévy flights and evaluate its goodness of fit or quality  $G_i$ ;
(5)   Choose a nest among  $N$  (say  $j$ ) randomly;
(6)   IF ( $G_i > G_j$ )
(7)     Replace  $j$  by the new solution;
(8)   END IF
(9)   Abandon a fraction ( $p_a$ ) of worse nests and build new ones;
(10)  Keep the best solutions (or nests with quality solutions);
(11)  Rank the solutions and find the current best solution;
(12) END WHILE
(13) Post-process results;

```

ALGORITHM 1: CS algorithm.

2.3. Artificial Intelligent Parameter Optimization

2.3.1. Original Cuckoo Search Algorithm. The CS algorithm was first developed by Sun et al. [20] in 2007. It is derived from the action of cuckoos laying their eggs in the nests of other birds to let those birds hatch eggs for them. However, once the host birds discover the cuckoo eggs, these eggs will be thrown away or the host birds will abandon their nests and rebuild a new nest elsewhere. The CS algorithm is constructed based on three assumptions: (a) Only one egg is laid by each cuckoo in a randomly selected nest; (b) The best nests will be carried over to the following generations; and (c) The number of available host nests is a constant, and the probability value of an egg laid by a cuckoo being discovered by the host bird is p_a which has the range of 0 to 1.

In the CS algorithm, each nest represents a solution. The pseudo code of the CS technique [21] presented in Algorithm 1 can aid in understanding the CS process.

The Lévy flight mentioned in the pseudo code of Algorithm 1 is generated according to:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{Lévy}(\lambda), \quad (13)$$

where $\alpha > 0$ is the step size, which should be related to the scale of the problem of interest. The product $\oplus$ indicates entry-wise multiplication location. A Lévy flight is considered when the step-lengths are distributed according to the following probability distribution:

$$\text{Lévy} \sim u = t^{-\lambda}, \quad 1 < \lambda \leq 3 \quad (14)$$

which has an infinite variance. Here, the consecutive steps of a cuckoo search essentially form a random walk process that obeys a power-law step-length distribution with a heavy tail.

2.3.2. Modified Cuckoo Search Method. Similar to other meta-heuristic algorithms, the original CS algorithm is simple and efficient; however, it has disadvantages such as insufficient search vigor and slow search speed during the latter part of the search. As one of the oldest optimization algorithms, the steepest descent (SD) method [22] is simple and intuitive. Currently, there are many effective optimization algorithms

established on the basis of this algorithm. In order to overcome the CS's shortcoming of slow convergence rate, the SD method is used to modify the CS algorithm, and the modified model is abbreviated as the SDCS model. In the SDCS model, the following equation substitutes for (13):

$$x_i^{t+1} = x_i^t + d_i^t, \quad (15)$$

where d_i^t is defined by

$$d_i^t = -\nabla g(x_i^t). \quad (16)$$

The SDCS process can be expressed by the following procedures.

Step 1. Initialize the initial points x^0 , the end error $\varepsilon > 0$, and set $k = 0$.

Step 2. Calculate $\nabla f(x^k)$. If $\|\nabla f(x^k)\| = \|\nabla g(x_i^k)\| \leq \alpha \oplus \text{Lévy}(\lambda) \leq \varepsilon$, terminate the iteration and output the value of x^k . Otherwise, go to Step 3.

Step 3. Set $p^k = -\nabla f(x^k)$.

Step 4. Conduct one-dimensional search. Get the value of t_k which satisfied equation $f(x^k + t_k p^k) = \min_{t \geq 0} f(x^k + t p^k)$; then set $x^{k+1} = x^k + t_k p^k$, $k := k + 1$, and return to Step 2.

The step size and step-length distribution function of the CS algorithm can be improved by using steepest descent due to its simplicity and flexibility. The final optimal solution can be obtained by modifying the step size and step-length distribution function constantly.

2.4. Proposed Novel Model. Based on the above methodologies, we propose a novel short-term wind speed forecasting portfolio with three steps (Figure 1): (I) data preprocessing: both the regular normalization preprocessing model and the EMD approach are used for data preprocessing, which reduces the effect of noise and different scales on the wind speed data; (II) artificially intelligent parameter optimization

FIGURE 1: Flowchart of the SDCS algorithm.

introduction: the unknown parameters in the SVM model are optimized by the CS algorithm; (III) parameter optimization approach modification: although the original CS algorithm is simple and efficient, it has disadvantages such as insufficient search vigor and slow search speed during the latter part of the search. Therefore, this paper proposes an improved parameter optimization approach based on the CS algorithm and the steepest descent (SD) method, which we call the SDCS model. The final forecasting model is called the EMD-SDCS-SVM model.

The performance of SVM depends on a good set of parameters, including the penalty parameter and the parameter of the kernel function. The parameter adjustment and selection of support vector machine is still a difficult issue in the research field. The generalization performance of support vector machine is closely related to the selection of specific parameters in the model. The parameter of penalty coefficient C and kernel parameters σ must be selected by the users. However, in practical applications, the forecasting complexity control is more difficult, because the parameters of C and σ must be adjusted simultaneously.

(1) *The Penalty Coefficient C* . The penalty coefficient C is to balance the model between the complexity and the training error, so that the model has better extending ability. Furthermore, the parameter C can control the robustness of the forecasting model. The different training groups have different

optimal values. For forecasting problems, if the parameter C is smaller, the punishment for miscalculation samples in the sample data is smaller. As a result, the training error becomes larger, and the system's generalization ability is poorer. When new data is forecasted by the model, the fitting error will be very high, and the phenomenon of "less learning" will appear. On the contrary, if the parameter C is too large, the weight of $(1/2)\|\omega\|^2$ will be smaller. Although the fitting error of the available data is very low, the fitting error of the new data is also very high. It is the so-called "overlearning" phenomenon. The generalization ability of the model is still very poor. Each sample data group has at least one suitable C , which makes the SVM generalization performance the best. Therefore, the correct choice of parameter C can improve the prediction accuracy of the model.

(2) *The Kernel Function σ* . For the kernel function of the SVM, the linear kernel function, polynomial kernel function, radial basis kernel function, and sigmoid function are usually the most used. The width σ of radial basis function is the same to all kernel functions, and σ can reflect corresponding width of inner product kernel for input. If σ is too small, it will lead to overfitting or memory of the training group. If σ is too large, it will make SVM discriminant function too gentle. Width of kernel function σ and the penalty coefficient C affect the shape of prediction curve of the support vector machine from different angles. In practical applications, too large or

FIGURE 2: Process of objective model.

too small penalty coefficient C and kernel function σ will make the generalization performance of the support vector machine worse.

Based on the analysis of influences of each parameter on the performance of SVM, we put forward the time series forecasting model by using modified cuckoo search (SDCS) algorithm to optimize SVM parameters. It not only maintains the characteristics of time series, but also can select the parameters of SVM automatically, which eliminates the blindness and randomness caused by artificial selection. The main procedures of this EMD-SDCS-SVM are as follows.

Procedure 1. Collect wind speed time series data. Use the EMD to preprocess the wind speed data and reconstitute the new wind speed time series, which will be treated as the training sample of the SVM model.

Procedure 2. Determine the range of C and σ , the maximum step ($\text{step}_{\max}$), the minimum step ($\text{step}_{\min}$), and the maximum number of iterations $N_{\max}$. Set the probability of an egg laid by a cuckoo being discovered by the host bird as $p_a = 0.25$, and initialize the number of the host nests as $N = 25$. Each nest corresponds to a two-dimensional vector (C, σ) .

Procedure 3. Search the optimum value of the two-dimensional vector (C, σ) according to the SDCS algorithm, and the detailed steps that need to be implemented in this procedure are shown in Figure 2.

Procedure 4. Use the optimum parameter values obtained in Procedure 3 and the processed data obtained in Procedure 1 to construct the forecasting model and obtain the forecasting results.

3. Simulation Examples and Discussions

3.1. Data Division and Parameter Initialization. Wind speed data recorded by four wind turbines (numbered #1, #2, #3, and #4) during the period from Jan 2, 2011, to Jan 6, 2011, with a time resolution of 10 minutes are used to verify the effectiveness of the new proposed hybrid model. The data from Jan 2 to Jan 5 are adopted as the in-sample data (i.e., training data), while those on Jan 6 are used as the out-of-sample data (i.e., testing data).

Step 1. The original wind speed series are decomposed into a high-frequency component and a low-frequency component,

FIGURE 3: Procedures of the proposed forecasting portfolio.

FIGURE 4: IMFs and residue results obtained by the EMD method.

which represents the noise signal and main features of the wind speed series (see Figures 3(a)–3(c)).

Step 2 (data splitting and normalization). The available wind speed series after noise reduction are split into the training set and the test set, which are denoted by including input sets and output sets for training parameters of SVM and consisting of inputs and outputs for the testing model's forecasting effectiveness, respectively. To establish the model, the training datasets and the input test sets are normalized with the same setting (see Figure 3(d)).

Step 3 (initialization: a SVM with two parameters). The penalty coefficient C and the kernel function σ are shown in Figure 3(e). The number of connection weights of the SVM is the size of the cities in the SDCS algorithm, namely, the dimension of the optimized parameters.

Step 4 (optimization). The objective function of the SDCS algorithm is given as follows:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{Lévy}(\lambda). \quad (17)$$

Step 5 (SVM construction). The best solution obtained by the SDCS algorithm is set to be the final connection weights of

SVM training and construction. The terminal condition of network training is set to be the reach of maximum iterations or no further improvement (see Figures 3(d)–3(e)).

Step 6 (EMD-SDCS-SVM construction for the test dataset). The forecasting data of the output test sets are generated by importing the input test sets based on the established optimal SVM (see Figure 3(e)).

Step 7 (evaluation). The quality of the EMD-SDCS-SVM is assessed by the indices SDCS and SVM, which presents the validity and informativeness of EMD, respectively. With the aim of comprehensive evaluation, MAPE is calculated as well.

To employ the methodologies introduced in Section 2 of this paper, the parameters contained in the models are initialized as follows: in the CS algorithm, the number of the host nests is initialized as $N = 25$, and the probability of an egg laid by a cuckoo being discovered by the host bird is given as $p_a = 0.25$. The Gaussian kernel function is chosen for the SVM method. In the GA algorithm, the maximum number of iterations is initialized as 50, and the population size is 100. The probability of cross is 0.3 and the probability of mutation is 0.1. When the CS algorithm and GA are adapted for SVM

FIGURE 5: Denoised and normalized results of the original data.

optimization, the search interval of the penalty coefficient C is set to $[0.1, 100]$, while the search interval of the kernel function is set to $[0.01, 1000]$.

3.2. Data Preprocessing Results. Wind speed data are first preprocessed by the EMD method. Figure 4 shows the IMFs and residue results obtained by the EMD method for the four wind turbines. As indicated in Figure 4, for the #2 and #3 wind turbines, 7 IMF sequences are extracted from the original wind speed training dataset, while 6 IMF sequences are extracted for the other two wind turbines. According to the principle of denoising, eliminating the high-frequency sequence from the IMF sequences can assist in obtaining cleaner data sequence, that is, data sequence with lower noise. For this paper, the first IMF sequence obtained by the EMD method is eliminated from the original data sequence to improve the accuracy of wind speed forecasting. The visualization of the denoise preprocessing of the EMD method of the four wind turbines is shown in Figure 5. The final results after denoise processing with the EMD method and the normalization operation are also presented in Figure 5.

3.3. Forecasting Results. To validate the effectiveness of the EMD-SDCS-SVM model in wind speed forecasting, the model is used to forecast wind speed with four horizons: 1-step-ahead, 2-step-ahead, 4-step-ahead, and 6-step-ahead. The forecasting results obtained by this model are compared with those obtained by the nonparameterization method

EMD-SVM, the unmodified parameterization method EMD-CS-SVM, and another parameterization method, EMD-GA-SVM, where GA is the abbreviation for the Genetic Algorithm [23].

Figure 6 presents the forecasting results of the four EMD-based models. In this figure, the wind speed data in the center of the circular rings with the value of 0 is the smallest, while the bigger the radius, the larger the wind speed value. The difference of the radius between each adjacent two circular rings is 5. As shown in Figure 6, the forecasting results obtained by these EMD-based models fit the actual wind speed data best when the forecasting horizon is 1-step-ahead, while the fit is the worst in the 6-step-ahead situation; that is, the deviation between the wind speed data forecast by the models and the actual wind speed data becomes larger as the forecasting horizon increases. In addition, the EMD-SVM and the EMD-GA-SVM methods deviate much more significantly from the actual data when compared to the other models.

In addition, the forecast results obtained by these models are analyzed according to the Quantile-Quantile (Q-Q) plot. The f quantile corresponding to a datum $q(f)$ means that approximately a decimal fraction f of the data can be found below the datum. The f quantile is calculated in the following manner: sort the data in a sequence $\{x_j\}_{j=1,2,\dots,n}$ in an ascending order. The sorted data $\{x_{(i)}\}_{i=1,2,\dots,n}$ have rank $i = 1, 2, \dots, n$. Then, the quantile value f_i for the datum $x_{(i)}$ is computed by

$$f_i = \frac{i - 0.5}{n}. \quad (18)$$

FIGURE 6: Forecasting results comparisons among EMD-modified models.

The 0.25, 0.5, and 0.75 quantiles are called the lower quantile, the median, and the upper quantile, respectively. The Q-Q plot is used to compare the quantiles of two samples. If the two samples come from the same type of distribution, the plot will be a straight line. A straight reference line that passes through the lower quantile and the upper quantile is helpful for assessing the Q-Q plot. The greater the distance from this reference line, the more likely it is that the two samples come from populations with different distributions.

The vertical and the horizontal axes of the Q-Q plot are the estimated quantiles from the two samples, respectively. If the sizes of these two samples are the same, the Q-Q plot is just a plot of the sorted data in the first sample against the sorted data in the second sample. As an example, Figure 7 provides an empirical Q-Q plot of the quantiles of the actual wind speed sequence versus the quantiles of the forecast data for the #4 wind turbine, where X represents the actual wind speed data sequence, and $Y1, Y2, Y3$, and $Y4$ denote the wind

FIGURE 7: Empirical Q-Q plot of the quantiles of the actual wind speed sequence versus the quantiles of the forecasted data for the #4 wind turbine.

speed data sequences forecasted by the EMD-SVM model, the EMD-CS-SVM model, the EMD-GA-SVM model, and the EMD-SDCS-SVM model, respectively. The straight line shown in each subplot is just the extrapolated line which joins the lower and the upper quantiles, and the vertical axis and the horizontal axis in each subplot are the estimated quantiles from the corresponding forecast data sequence and the actual data sequence. As observed from Figure 7, the forecast values sometimes are larger than the actual values (corresponding to the plus symbol located above the straight line), while sometimes they are smaller than the actual values (corresponding to the plus symbol located below the straight line). Figure 7 also reveals that the EMD-SDCS-SVM model fits the actual wind speed data best when compared to the other three models.

3.4. Forecasting Error Comparison. Results presented in Section 3.2 provide graphical visualization of the performance of the different forecasting models. In this section, the superior performance of the EMD-SDCS-SVM model is shown quantitatively. To do this, two error evaluation criteria named the root mean squared error (RMSE) and the mean absolute percentage error (MAPE) are adopted and defined as follows:

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2}, \\ \text{MAPE} &= \frac{1}{n} \sum_{i=1}^n \left| \frac{x_i - \hat{x}_i}{x_i} \right| \times 100\%, \end{aligned} \quad (19)$$

where n is the number of data points in the out-of-sample data and x_i and $\hat{x}_i$ are the actual value and the forecasted value, respectively.

From Table 1 and Figure 8, it can be seen that compared to the ANN forecasting models, the SVM models perform favorable forecasting accuracy; in particular in four- and six-step-ahead forecasting result, the SVM is superior to the ANN model of BPNN, Elman NN, and WNN.

The forecasting error results with different forecasting horizons of these 4 models are given in Table 2 and Figure 9. As observed from Table 2 and Figure 9, the forecasting error values become larger as the forecasting horizon increases. For the #1, #2, and #4 wind turbines, the EMD-SDCS-SVM model always obtains more accurate wind speed forecasting results than the other three models. In addition, for the #3 wind turbine, the EMD-SDCS-SVM model is superior to both the EMD-SVM and the EMD-CS-SVM models, which means that the proposed novel model EMD-SDCS-SVM has made promising predictions and has better performance than its individual forecasting components.

4. Conclusions

Wind speed forecasting plays a significant part in the economy and security of wind farm systems' operation. Accurate forecasting results have significant influence on the economy. Recently, academia and industry have paid more attention to wind speed forecasting. More accurate forecasting could reduce costs and risks, improve the security of power systems, and help administrators develop an optimal action program,

TABLE 1: Forecasting error results between SVM and ANN forecasting model.

Horizon	Wind turbine	Error type	SVM	BPNN	Elman NN	WNN	ARIMA	Horizon	Wind turbine	Error type	SVM	BPNN	Elman NN	WNN	ARIMA
1-step-ahead	1#	MSE	0.7306	1.9341	1.5646	2.3015	0.8544	4-step-ahead	1#	MSE	0.8133	2.1649	2.181	3.1542	1.2014
		MAPE	6.72	11.22	11.18	13.71	4.741			MAPE	7.01	12.25	14.03	17.81	6.882
	2#	MSE	0.4339	0.9645	1.1323	2.6048	0.9006		2#	MSE	0.8471	3.2995	2.1403	5.0735	1.5609
		MAPE	5.36	8.87	8.99	13.06	5.889			MAPE	7.58	14.08	12.99	20.23	7.373
	3#	MSE	0.4838	2.5618	1.4261	1.7689	0.9942		3#	MSE	0.8871	3.1718	2.7741	4.7698	1.8086
		MAPE	5.37	13.73	10.65	12.73	6.218			MAPE	7.19	14.28	15.76	19.77	7.280%
	4#	MSE	0.7954	1.8717	1.5713	3.2243	2.4862		4#	MSE	1.4467	3.6119	3.1221	6.4129	1.7779
		MAPE	8.19	12.73	12.11	14.13	7.283			MAPE	11.26	17.69	16.59	19.94	7.821
2-step-ahead	1#	MSE	0.639	1.2661	1.8282	3.6953	1.0663	6-step-ahead	1#	MSE	1.0364	2.708	2.6968	4.1962	1.3271
		MAPE	6.19	9.31	12.70	17.74	5.922			MAPE	8.03	13.56	16.90	18.40	6.709
	2#	MSE	0.5788	1.2924	1.6292	3.2163	1.0885		2#	MSE	1.0511	3.5101	2.5112	5.6064	2.0692
		MAPE	6.31	10.74	10.66	16.69	6.154			MAPE	8.44	14.90	14.27	22.60	7.954
	3#	MSE	0.6497	2.9984	1.8137	2.5426	1.5840		3#	MSE	1.0549	2.6744	2.9361	6.5295	1.3540
		MAPE	6.27	13.72	13.08	13.14	5.648			MAPE	7.84	12.49	16.29	20.57	7.380
	4#	MSE	1.0798	3.8907	1.8167	2.671	1.4168		4#	MSE	1.7788	4.7215	3.7118	4.8947	2.7558
		MAPE	9.58	18.82	13.46	15.94	6.955			MAPE	12.40	22.09	19.21	20.59	10.446

FIGURE 8: Forecasting results of SVM and others ANN.

TABLE 2: Forecasting error results with different forecasting horizons.

Horizon	Wind turbine	Error type	EMD-SVM	EMD-GA-SVM	EMD-CS-SVM	EMD-FA-SVM	EMD-SDCS-SVM	Horizon	Wind turbine	Error type	EMD-SVM	EMD-GA-SVM	EMD-CS-SVM	EMD-FA-SVM	EMD-SDCS-SVM
1-step-ahead	#1	MSE	0.6798	0.6798	0.6798	0.6460	0.6382	#1	MSE	0.9262	0.9262	0.9262	0.9262	1.5420	0.7883
		MAPE	4.6937	4.6937	4.6937	4.924	4.4052			MAPE	6.3854	6.3854	6.3854	7.103%	5.5685
	#2	MSE	0.8628	0.8226	0.8534	0.9262	0.7959	#2	MSE	1.256	1.256	1.0524	1.1339	0.9147	1.01
		MAPE	6.0093	5.8184	5.9602	5.327	5.597			MAPE	8.2016	7.279	7.8073	6.197	6.8343
	#3	MSE	0.8256	0.7907	0.8304	0.9553	0.803	#3	MSE	1.0914	0.9676	1.1172	1.1185	1.0714	1.0714
		MAPE	5.5431	5.2876	5.5765	6.234	5.2886			MAPE	7.2059	6.4716	7.3004	6.897	6.8158
	#4	MSE	0.8657	0.8514	0.9065	0.7401	0.8135	#4	MSE	1.1689	1.1094	1.3419	1.0754	1.0149	1.0149
		MAPE	5.5708	5.5101	5.8196	5.493	5.1737			MAPE	6.8888	6.526	7.7831	6.548%	6.0562
2-step-ahead	#1	MSE	0.7707	0.7707	0.7707	0.9932	0.7184	#1	MSE	0.9982	0.9982	0.9982	0.9982	1.8679	0.8342
		MAPE	5.3944	5.3944	5.3944	5.006	5.0049			MAPE	6.9568	6.9568	6.9568	9.435	5.898
	#2	MSE	1.0091	0.9242	0.9806	0.7437	0.8885	#2	MSE	1.4805	1.0856	1.2166	1.2166	2.2042	1.0627
		MAPE	6.8987	6.4877	6.7938	5.892	6.0704			MAPE	8.9118	7.6722	8.3732	8.378%	7.0088
	#3	MSE	0.8876	0.8259	0.8963	1.2746	0.8674	#3	MSE	1.1505	0.9688	1.1615	1.1615	1.7628	1.0857
		MAPE	5.9299	5.5404	5.9339	7.069	5.5753			MAPE	7.8968	6.643	7.9065	6.910	6.949
	#4	MSE	0.9916	0.9558	1.0946	1.0891	0.8804	#4	MSE	1.2147	1.0537	1.5362	1.5362	2.8053	0.981
		MAPE	6.2854	5.8933	6.8798	6.587	5.4089			MAPE	7.3785	6.423	8.9868	8.532	6.1465

FIGURE 9: MSE and MAPE visualization of the four wind turbines.

thereby enhancing the economic social benefits of power-grid management. Therefore, it is highly desirable to develop techniques for wind speed forecasting to improve accuracy. However, individual models do not always achieve a desirable performance. The proper selection method of a hybrid model can reduce certain negative effects that are inherent to each of these individual models; moreover, the hybrid forecasting model can make full use of the advantages of each of the individual models and is less sensitive, in certain cases, to the factors that make the individual models perform in an undesirable manner.

In this paper, to enhance the forecasting capacity of the proposed combined model, consisting of three procedures, the data preprocessing procedure, the artificial intelligent parameter optimization introduction procedure, and the parameter optimization approach modification procedure were integrated. The SVM model used in this paper can handle data with nonlinear features, and the SD technique is adopted to enhance the convergence speed of the CS algorithm, which is utilized to optimize the parameters in the SVM model. The effectiveness and robustness of the proposed approach has been successfully tested by the real wind speed data sampled at four wind turbines. Based on the Q-Q plot and the error comparison, results show that the developed portfolio EMD-SDCS-SVM has made promising predictions and has better performance than its individual forecasting components despite very small MAPE and MSE values. For instance, the average MAPE values of the combined model were 0.7138%, 1.0281%, 4.8394%, 0.9239%, and 7.3367%, which are lower than those of BPNN, WNN, and Elman NN. By improving forecasting accuracy and stability, in the wind farm, a large amount of money and energy could be saved. The hybrid model can be applied to forecast the wind speed that can be used in wind power scheduling to produce various benefits, saving on economic dispatching, reducing production costs, and reducing the spinning reserve capacity of electrical power system. This model is also useful for supporting wind farm decision making in practice. The combined forecasting model, which has high precision, is a promising model for use in the future. In addition, this hybrid model can be utilized in other forecasting fields, such as product sales forecasting, tourism demand forecasting, early warning and flood forecasting, and traffic-flow forecasting.

Competing Interests

The authors declare that they have no competing interests.

References

- [1] A. Tascikaraoglu, B. M. Sanandaji, K. Poolla, and P. Varaiya, "Exploiting sparsity of interconnections in spatio-temporal wind speed forecasting using Wavelet Transform," *Applied Energy*, vol. 165, pp. 735–747, 2016.
- [2] M. Lydia, S. Suresh Kumar, A. I. Selvakumar, and G. E. P. Kumar, "Linear and non-linear autoregressive models for short-term wind speed forecasting," *Energy Conversion and Management*, vol. 112, pp. 115–124, 2016.
- [3] Z. Men, E. Yee, F.-S. Lien, D. Wen, and Y. Chen, "Short-term wind speed and power forecasting using an ensemble of mixture density neural networks," *Renewable Energy*, vol. 87, pp. 203–211, 2016.
- [4] H. Liu, H.-Q. Tian, Y.-F. Li, and L. Zhang, "Comparison of four Adaboost algorithm based artificial neural networks in wind speed predictions," *Energy Conversion and Management*, vol. 92, pp. 67–81, 2015.
- [5] H. Liu, H. Q. Tian, X. F. Liang, and Y. F. Li, "New wind speed forecasting approaches using fast ensemble empirical model decomposition, genetic algorithm, Mind Evolutionary Algorithm and Artificial Neural Networks," *Renewable Energy*, vol. 83, pp. 1066–1075, 2015.
- [6] Q. Hu, R. Zhang, and Y. Zhou, "Transfer learning for short-term wind speed prediction with deep neural networks," *Renewable Energy*, vol. 85, pp. 83–95, 2016.
- [7] H. Liu, H.-Q. Tian, X.-F. Liang, and Y.-F. Li, "Wind speed forecasting approach using secondary decomposition algorithm and Elman neural networks," *Applied Energy*, vol. 157, pp. 183–194, 2015.
- [8] Z. Guo, D. Chi, J. Wu, and W. Zhang, "A new wind speed forecasting strategy based on the chaotic time series modelling technique and the Apriori algorithm," *Energy Conversion and Management*, vol. 84, pp. 140–151, 2014.
- [9] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 273–284, 2006.
- [10] J. Wang and J. Hu, "A robust combination approach for short-term wind speed forecasting and analysis—combination of the ARIMA (Autoregressive Integrated Moving Average), ELM (Extreme Learning Machine), SVM (Support Vector Machine) and LSSVM (Least Square SVM) forecasts using a GPR (Gaussian Process Regression) model," *Energy*, vol. 93, pp. 41–56, 2015.
- [11] H. B. Azad, S. Mekhilef, and V. G. Ganapathy, "Long-term wind speed forecasting and general pattern recognition using neural networks," *IEEE Transactions on Sustainable Energy*, vol. 5, no. 2, pp. 546–553, 2014.
- [12] K. Chen and J. Yu, "Short-term wind speed prediction using an unscented Kalman filter based state-space support vector regression approach," *Applied Energy*, vol. 113, pp. 690–705, 2014.
- [13] A. Gani, K. Mohammadi, S. Shamshirband, T. A. Altameem, D. Petković, and S. Ch, "A combined method to estimate wind speed distribution based on integrating the support vector machine with firefly algorithm," *Environmental Progress & Sustainable Energy*, vol. 35, no. 3, pp. 867–875, 2016.
- [14] Y. J. Lin, U. Kruger, J. Zhang, Q. Wang, L. Lamont, and L. El Chaar, "Seasonal analysis and prediction of wind energy using random forests and ARX model structures," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 1994–2002, 2015.
- [15] E. Baccarelli, N. Cordeschi, A. Mei, M. Panella, M. Shojafar, and J. Stefa, "Energy-efficient dynamic traffic offloading and reconfiguration of networked data centers for big data stream mobile computing: review, challenges, and a case study," *IEEE Network*, vol. 30, no. 2, pp. 54–61, 2016.
- [16] D. Liu, D. Niu, H. Wang, and L. Fan, "Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm," *Renewable Energy*, vol. 62, pp. 592–597, 2014.

- [17] S. J. Watson, L. Landberg, and J. A. Halliday, "Application of wind speed forecasting to the integration of wind energy into a large scale power system," *IEEE Proceedings: Generation, Transmission and Distribution*, vol. 141, no. 4, pp. 357–362, 1994.
- [18] N. E. Huang, Z. Shen, S. R. Long et al., "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London A*, vol. 454, no. 1971, pp. 903–995, 1998.
- [19] Y. Gan, L. Sui, J. Wu, B. Wang, Q. Zhang, and G. Xiao, "An EMD threshold de-noising method for inertial sensors," *Measurement*, vol. 49, no. 1, pp. 34–41, 2014.
- [20] H.-X. Sun, N.-N. Zhao, and X.-H. Xu, "Text region localization using wavelet transform in combination with support vector machine," *Journal of Northeastern University (Natural Science)*, vol. 28, no. 2, pp. 165–168, 2007.
- [21] M. Bouzerdoun, A. Mellit, and A. M. Pavan, "A hybrid model (SARIMA-SVM) for short-term power forecasting of a small-scale grid-connected photovoltaic plant," *Solar Energy*, vol. 98, pp. 226–235, 2013.
- [22] X. Wang, J. Wen, Y. Zhang, and Y. Wang, "Real estate price forecasting based on SVM optimized by PSO," *Optik*, vol. 125, no. 3, pp. 1439–1443, 2014.
- [23] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proceedings of the World Congress on Nature & Biologically Inspired Computing (NaBIC '09)*, pp. 210–214, Coimbatore, India, December 2009.

Hindawi

Submit your manuscripts at
<http://www.hindawi.com>