energies

MDPI

*Article*

# Ultra-Short-Term Wind Speed Forecasting Using the Hybrid Model of Subseries Reconstruction and Broad Learning System

**Ming Pang [1,*], Lei Zhang [2], Yajun Zhang [1], Ao Zhou [1], Jianming Dou [1] and Zhepeng Deng [3]**

[1] School of Mechanical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China; zhangyajun323@163.com (Y.Z.); cn_zhouao@163.com (A.Z.); lgy_116@163.com (J.D.)
[2] School of Mechanical and Electrical Engineering, Xi'an Polytechnic University, Xi'an 710048, China; 20201205@xpu.edu.cn
[3] School of Chemical and Chemical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China; dengzp@163.com
* Correspondence: aksu_141@163.com

**Abstract:** The traditional decomposition–combination wind speed forecasting model has high complexity and a long calculation time. As a result, an ultra-short-term wind speed hybrid forecasting model based on a broad learning system (BLS) that combines improved variational mode decomposition (EPSO-VMD, EVMD) and subseries reconstruction (SR) is proposed in this work. The values of $K$ and $\alpha$ in the EVMD are determined by minimum mean envelope entropy (MMEE) and enhanced particle swarm optimization (EPSO), and EVMD is used to decompose the original wind speed data. SR is applied to recombine the subseries obtained by EVMD to improve the forecasting efficiency. The sample entropy (SE) is used to quantify the subseries' complexity, and they are then adaptively divided into high-entropy and low-entropy subseries. Adjacent high-entropy subseries of approximate entropy values are merged to obtain a new group of reconstructed high-entropy subseries, while the low-entropy subseries merge into a new subseries as well. Then, the forecasting results of the reconstructed high- and low-entropy subseries are calculated via the BLS and ARIMA models. Numerical simulation results show that the proposed method is more effective than traditional methods.

**Keywords:** ultra-short-term wind speed forecasting; broad learning system; variational mode decomposition; subseries reconstruction; sample entropy

check for updates

**Citation:** Pang, M.; Zhang, L.; Zhang, Y.; Zhou, A.; Dou, J.; Deng, Z. Ultra-Short-Term Wind Speed Forecasting Using the Hybrid Model of Subseries Reconstruction and Broad Learning System. *Energies* **2022**, *15*, 4492. https://doi.org/10.3390/en15124492

Academic Editor: John Boland

Received: 17 April 2022
Accepted: 19 June 2022
Published: 20 June 2022

## 1. Introduction

Climate change and reducing greenhouse gas emissions have become central issues on the global sustainability agenda. For this reason, wind energy, as a renewable and clean energy source in nature, is undergoing rapid development worldwide [1]. However, variable wind speed has restricted its power quality and related system stability in terms of the output power of wind turbine generators (WTG) [2]. This phenomenon significantly restricts wind energy's integration with the grid [3]. Comparatively speaking, great significance is therefore attached to the accurate prediction of wind power and wind speed as regards the exploitation of renewable energy sources [4].

According to the technical specifications of power systems, wind speed forecasting can be segmented into four different timescales, namely, ultra-short-term (<30 min), short-term (0.5~6 h), medium-term (6~24 h), and long-term (1~7 days). Turbine control and load tracking are based on ultra-short-term forecasting. Preload sharing is based on short-term forecasting. For power system management and energy trading, medium-term forecasting is utilized. Maintenance schedules for wind turbines are based on long-term forecasting.

Over the past several years, numerous researchers have devoted their efforts to proposing more accurate and stable statistical models for ultra-short-term wind speed forecasting. Recently, decomposition–combination forecasting models based on time–frequency analysis have been studied and proposed. The time–frequency analysis method is used to decompose different feature information for wind speed time series, which reduces the influence of non-stationarity on the prediction result and improves prediction accuracy. A. Tascikaraoglu [5] incorporated a forecasting model based on wavelet transform (WT) to improve the forecasting accuracy of electrical price and wind speed; however, the wavelet transform effect depends on the wavelet base and decomposition scale, and the adaptability is poor. Wavelet transform was combined with a deep belief network (DBN) to improve effectively prediction accuracy, as suggested by [6]; however, this model usually includes multiple neural layers, and each layer contains hundreds to thousands of node nerves. The interplay of neurons between the layers is intricate, resulting in a dramatic increase in modeling complexity and computational time. Empirical mode decomposition (EMD) has been used to decompose wind speed data into a series of intrinsic mode functions (IMF), after which the improved ARIMA model was used for each IMF and margin, as proposed in [7]. Ning [8] introduced an EMD-based signal filtering method that is fully data-driven in order to reduce the noise signals of electricity demand series, a method that could noticeably improve forecasting accuracy. Huang [9] applied ensemble empirical mode decomposition (EEMD) to extract a series of IMFs from an initial wind data sequence, and created the LSTM-GRNN-BSO model for multi-step wind speed forecasting. While this method, which is based on EMD, can improve forecasting results EMD has an end effect and over-enveloping problem. Sun [10] suggested local mean decomposition (LMD) and improved least-square support machine (LSSVM) to predict short-term wind speed, which can improve the modeling accuracy of LSSVM. However, the ability of the LMD algorithm to judge the FM signal needs to be tested. If the selection of the sliding span is improper, the function will not converge, and the structural accuracy will be affected by over-smoothing.

In the process of obtaining the decomposition component, variational mode decomposition (VMD) uses the optimal solution of the iterative search variational model to determine the frequency center and bandwidth of each component. In essence, it is a plurality of adaptive Wiener filter groups which are more robust or perform better than their counterparts [11]. Sun [12] suggested the VMD-P-ARIMA-BP-PSOLSSVM prediction-optimization model to forecast wind speed in the short term. Han [13] proposed a variational mode decomposition long short-term memory (VMD-LSTM) method to improve the accuracy of multi-step wind power forecasting. While VMD is an effective decomposition technology, the parameters K and $\alpha$ need to be determined in advance, and empirical values are used in most cases.

The broad learning system (BLS) is based on the input of the Map Vector as a Random Vector Functional Link Neural Network (RVFLNN) used to generate enhanced nodes. The network system can be quickly obtained through pseudo-inverse operations. This avoids the time-consuming training of a large number of BLS parameters with complex structures. The BLS has been introduced into wind speed forecasting in [14,15] to improve precision and computational efficiency.

In view of the above research situation, we propose an EVMD-SR-BLS-ARIMA hybrid wind speed forecasting model, which is based on BLS combined with improved variational mode decomposition (EPSO-VMD, EVMD), subseries reconstruction (SR), and ARIMA. The main contributions of this paper are summarized as follows.

(1) In the proposed EVMD, the minimum mean envelope entropy (MMEE) and enhanced particle swarm optimization (EPSO) algorithm are introduced in order to solve the optimal value of $K$ and $\alpha$ and improve the computational convergence.

(2) SR is applied to recombine the subseries obtained by EVMD. The sample entropy (SE) is used to quantify the complexity of the subseries, which are then adaptively divided into new high-entropy and low-entropy subseries. The adjacent high-entropy subseries of approximate entropy values are merged to obtain a new group of reconstructed high-entropy subseries, while the low-entropy subseries are merged into a new subseries as well.

(3) A novel and robust hybrid prediction model (EVMD-SR-BLS-ARIMA) is proposed and employed for ultra-short-term wind power forecasting. This avoids the time-consuming training of a deep structures model. This paper introduces BLS, which is time-efficient and constantly updates the parameters of the forecasting model, to ultra-short-term wind speed forecasting. The forecasting results of nascent high- and low-entropy subseries are calculated using the BLS and ARIMA models. In order to improve the forecasting accuracy, an error-corrected EVMD-SR-BLS-ARIMA model is developed to post-process the errors.

The rest of the paper is organized as follows: Section 2 briefly reviews the related approaches involved in this paper; Section 3 describes the EVMD-SR-BLS-ARIMA model for wind speed forecasting in detail; Section 4 presents experimental simulations and comparative examples to demonstrate the performance of the proposed algorithm; finally, we present our conclusions in Section 5.

## 2. Related Work

In this section, the proposed methods are introduced, including the VMD and BLS network.

### *2.1. Variational Mode Decomposition*

Variational mode decomposition (VMD) is an adaptive signal processing method first proposed by Konstantin Dragomiretskiy and Dominique Zosso [11]. VMD can decompose the original input signal, $x(t)$, into a $K$ intrinsic modes function (IMF), $u_k(t)$ ($k$ = 1, 2, . . . , $K$). In this process, the bandwidth sum of each mode is minimized, and the aggregation of modes $u(t)$ is equal to the input signal, $x(t)$. Compared with EMD, the instantaneous frequency of $u(t)$ is physically meaningful, and is written as in Equation (1):

$$u_k(t) = A_k(t)\cos(\Phi_k(t)) \tag{1}$$

where $A_k(t)$ is the non-negative envelope of $u_k(t)$, $\Phi_k$ is the phase, and the instantaneous frequency is the derivative of $\Phi_k$.

VMD, completely different from EMD, transforms the signal decomposition process into a variational problem. Its adaptive decomposition is based on the optimal solution of the constrained variational problem, as described in Equation (2):

$$\min_{\{u_k\},\{\omega_k\}} \left\{ \sum_{k=1}^{K} \|\partial_t \left[ \left( \delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] \mathrm{e}^{-j\omega_k t} \|_2^2 \right\} \quad \text{s.t.} \sum_{k=1}^{K} u_k = x \tag{2}$$

where $u_k$ is the $k$-th IMF mode function, $\omega_k$ is the $k$-th center frequency, $\partial_t$ is the partial derivative of the function for time $t$, $\delta(t)$ is the Dirac distribution, and $*$ indicates the convolution.

In order to find the optimal solution, the Lagrange multiplier $\lambda$ and the quadratic penalty term $\alpha$ are introduced to Equation (2). Then, the constrained variational problem is turned into an unconstrained one, which can be expressed as in Equation (3):

$$\begin{aligned} L(\{u_k\},\{\omega_k\},\lambda) &= \alpha \sum_{k=1}^{K} \|\partial_t \left[ \left( \delta(t) + \tfrac{j}{\pi t} \right) * u_k(t) \right] \mathrm{e}^{-j\omega_k t} \|_2^2 \\ &+ \|x(t) - \sum_{k=1}^{K} u_k(t)\|_2^2 + \left\langle \lambda(t), x(t) - \sum_{k=1}^{K} u_k(t) \right\rangle \end{aligned} \tag{3}$$

Meanwhile, the alternating direction multiplier algorithm is employed to update the parameters of Equation (3) iteratively in order to calculate the saddle point of the Lagrange function. This process can be represented as in Equations (4) and (5) [16]:

$$\hat{u}_k^{n+1}(\omega) = \frac{\hat{x}(\omega) - \sum\limits_{i \neq k} \hat{u}_i(\omega) + \frac{\hat{\lambda}(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2} \tag{4}$$

$$\hat{\omega}_k^{n+1} = \frac{\int_0^{\infty} \omega |\hat{u}_k(\omega)|_2 \mathrm{d}\omega}{\int_0^{\infty} |\hat{u}_k(\omega)|_2 \mathrm{d}\omega} \tag{5}$$

where $\hat{u}_k^{n+1}\ (\omega)$, $\hat{u}_i\ (\omega)$, $\hat{x}\ (\omega)$ and $\hat{\lambda}\ (\omega)$ are the Fourier transforms of $u_k^{n+1}\ (t)$, $u_i\ (t)$, $x\ (t)$ and $\lambda(t)$, respectively, and $n$ is the iteration number.

### *2.2. Broad Learning System*

The broad learning system (BLS) is a kind of flat function-link neural network proposed by C. L. Philip Chen and Zhuliu Liu [17] which can overcome the time-consuming training process of deep-learning networks. The core concept of BLS is to transform the original input to the mapped feature (the mapped feature being an input of a function-link neural network) and to establish the enhancement node. In the BLS network, all mapping features and enhancement nodes are directly connected to the output; its structure is shown in Figure 1. The BLS network consists of mapped features, enhancement nodes, and an output layer.

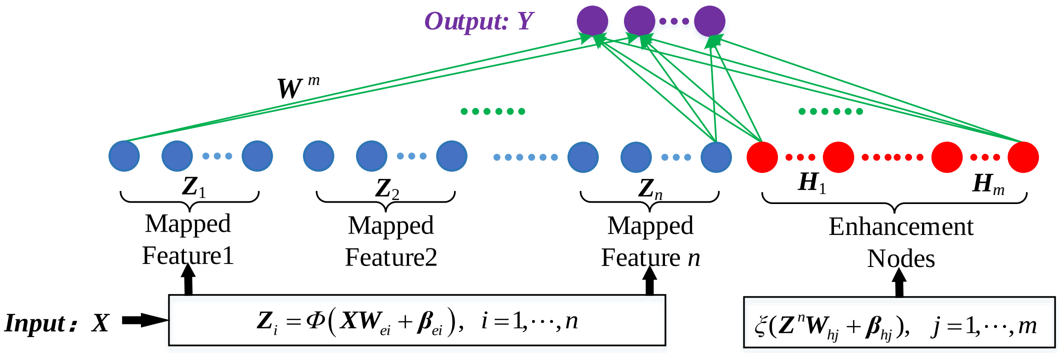


**Figure 1.** Structure of the broad learning system.

Assuming that the BLS network sets $n$ mapped features and each mapped feature is equipped with $k$ nodes, the $i$-th mapped feature $\boldsymbol{Z}_i$ can be represented as in Equation (6):

$$\boldsymbol{Z}_i = \Phi(\boldsymbol{X}\boldsymbol{W}_{ei} + \boldsymbol{\beta}_{ei}), i = 1, \cdots, n \tag{6}$$

where $\boldsymbol{X}$ is the input data, $\boldsymbol{W}_{ei}$ and $\boldsymbol{\beta}_{ei}$ are the random weights matrix and the random bias matrix of mapping feature, respectively, $\Phi(\cdot)$ is the nonlinear activation function, and $\boldsymbol{Z}_i$ is the output matrix.

Denoting all of the feature nodes of BLS as $\boldsymbol{Z}_n \equiv [\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_n]$, the $j$-th enhancement node $\boldsymbol{H}_j$ can be defined as in Equation (7):

$$\boldsymbol{H}_j = \xi(\boldsymbol{Z}^n \boldsymbol{W}_{hj} + \boldsymbol{\beta}_{hj}), j = 1, \cdots, m \tag{7}$$

where $\boldsymbol{W}_{hj}$ and $\boldsymbol{\beta}_{hj}$ are the random weights matrix and the random bias matrix of enhancement node, respectively, and $\xi(\cdot)$ is the nonlinear activation function. Hence, in mathematical terms, the network output $\boldsymbol{Y}$ of the BLS can be denoted as in Equation (8):

$$\begin{aligned} \boldsymbol{Y} &= [\boldsymbol{Z}_1, \cdots, \boldsymbol{Z}_n | \xi(\boldsymbol{Z}^n \boldsymbol{W}_{h1} + \boldsymbol{\beta}_{h1}), \cdots, \xi(\boldsymbol{Z}^n \boldsymbol{W}_{hm} + \boldsymbol{\beta}_{hm})] \boldsymbol{W}^m \\ &= [\boldsymbol{Z}_1, \cdots, \boldsymbol{Z}_n | \boldsymbol{H}_1, \cdots, \boldsymbol{H}_m] \boldsymbol{W}^m = [\boldsymbol{Z}^n | \boldsymbol{H}^m] \boldsymbol{W}^m \end{aligned} \tag{8}$$

where $\boldsymbol{W}_m$ is the network connection weight matrix and $\boldsymbol{W}_m = [\boldsymbol{Z}_n \mid \boldsymbol{H}_m]^+\boldsymbol{Y}$. $[\boldsymbol{Z}_n \mid \boldsymbol{H}_m]^+$ is the Moore–Penrose inverse matrix of the matrix $[\boldsymbol{Z}_n \mid \boldsymbol{H}_m]$, the value of which is estimated by ridge regression. Meanwhile, $[\boldsymbol{Z}_n \mid \boldsymbol{H}_m]^+$ can be calculated using Equation (9):

$$[\boldsymbol{Z}^n|\boldsymbol{H}^m]^+ = \lim_{\lambda\to 0}(\lambda \boldsymbol{I} + [\boldsymbol{Z}^n|\boldsymbol{H}^m][\boldsymbol{Z}^n|\boldsymbol{H}^m]^+)[\boldsymbol{Z}^n|\boldsymbol{H}^m]^T \tag{9}$$

As in a random vector function-link neural network, the network weights of BLS can be updated dynamically and quickly. Assuming that the initial network has $n$ mapping features and $m$ enhancement nodes and that the network input features matrix is $\boldsymbol{A}_n^m = [\boldsymbol{Z}^n \mid \boldsymbol{H}^m]$, when denoting $\boldsymbol{X}_a$ as the new added data of BLS network the mapping features and enhancement nodes of the new added data are represented as in Equation (10):

$$\boldsymbol{A}_x = [\Phi(\boldsymbol{X}_a\boldsymbol{W}_{e1} + \boldsymbol{\beta}_{e1}), \cdots, \Phi(\boldsymbol{X}_a\boldsymbol{W}_{en} + \boldsymbol{\beta}_{en})|\xi(\boldsymbol{Z}_x^n\boldsymbol{W}_{h1} + \boldsymbol{\beta}_{h1}), \cdots, \xi(\boldsymbol{Z}_x^n\boldsymbol{W}_{hm} + \boldsymbol{\beta}_{hm})] \tag{10}$$

where $\boldsymbol{Z}_x^n = [\Phi(\boldsymbol{X}_a\boldsymbol{W}_{e1} + \boldsymbol{\beta}_{e1}), \ldots, \Phi(\boldsymbol{X}_a\boldsymbol{W}_{en} + \boldsymbol{\beta}_{en})]$ is the mapping features matrix corresponding to $\boldsymbol{X}_a$ and $\boldsymbol{W}_{ei}$, $\boldsymbol{\beta}_{ei}$ and $\boldsymbol{W}_{hj}$, $\boldsymbol{\beta}_{hj}$ are randomly generated during the initialization of the network. Hence, the updated matrix of the output features, ${}^x\boldsymbol{A}_n^m$, is provided by Equation (11):

$${}^x\boldsymbol{A}_n^m = \begin{bmatrix} \boldsymbol{A}_n^m \\ \boldsymbol{A}_x^T \end{bmatrix} \tag{11}$$

According to the Moore–Penrose inverse matrix recursive algorithm, the Moore–Penrose inverse matrix of the output features update matrix ${}^x\boldsymbol{A}_n^m$ can be obtained as in Equation (12):

$$({}^x\boldsymbol{A}_n^m)^+ = [(\boldsymbol{A}_n^m)^+ + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{D}^{\mathrm{T}}\Big|\boldsymbol{B}^{\mathrm{T}}] \tag{12}$$

where $\boldsymbol{D}^T = \boldsymbol{A}_x^T(\boldsymbol{A}_n^m)^+, \boldsymbol{B}^{\mathrm{T}} = \begin{cases} (\boldsymbol{A}_x^T - \boldsymbol{D}^T\boldsymbol{A}_n^m)^+, \boldsymbol{A}_x^T - \boldsymbol{D}^T\boldsymbol{A}_m \neq 0 \\ (1 + \boldsymbol{D}^T\boldsymbol{D})^{-1}(\boldsymbol{A}_n^m)^+\boldsymbol{D}, \boldsymbol{A}_x^T - \boldsymbol{D}^T\boldsymbol{A}_n^m = 0 \end{cases}$.

Therefore, the new network weight matrix ${}^x\boldsymbol{W}_x^m$ can be represented as in Equation (13):

$${}^x\boldsymbol{W}_n^m = ({}^x\boldsymbol{A}_n^m)^+\boldsymbol{Y}_x \tag{13}$$

where $\boldsymbol{Y}_x$ is the network output and can be expressed as in Equation (14):

$$\boldsymbol{Y}_x = \begin{bmatrix} \boldsymbol{Y} \\ \boldsymbol{Y}_a \end{bmatrix} \tag{14}$$

where $\boldsymbol{Y}_a$ is the output data corresponding to the new added data, $\boldsymbol{X}_a$.

Based on Equations (8), (11), (12), and (14), the network weight matrix ${}^x\boldsymbol{W}_x^m$ can be transformed as in Equation (15):

$${}^x\boldsymbol{W}_n^m = \boldsymbol{W}^m + \left(\boldsymbol{Y}_a^T - \boldsymbol{A}_x^T\boldsymbol{W}^m\right)\boldsymbol{B} \tag{15}$$

## 3. EVMD-SR-BLS-ARIMA Hybrid Wind Speed Forecasting Model

In this section, a novel EVMD-SR-BLS-ARIMA hybrid wind speed forecasting model is developed that is able to attain high precision and efficiency.

### 3.1. Improved Variational Mode Decomposition (EVMD)

The non-linearity, non-smoothness and randomness of wind speed are the main factors that decrease accuracy in wind speed forecasting. VMD has the advantages of high computational efficiency and strong robustness, and can improve the smoothness of the original signal and reduce the complexity of modelling. Compared with the EMD, the values of IMF $K$ and the quadratic penalty term $\alpha$ must be pre-set before using VMD. If the $K$ value is too small, the feature of the original signal cannot be extracted efficiently. Conversely, excessive decomposition leads to added complexity in the model [18]. Furthermore, a smaller value of $\alpha$ results in the bandwidth of each obtained mode function being larger. Therefore, the choice of $K$ and $\alpha$ directly affects the decomposition effect and reconstruction accuracy.

In order to address this problem, improved variational mode decomposition (EPSO-VMD, EVMD) is proposed, in which the minimum mean envelope entropy (MMEE) and enhanced particle swarm optimization (EPSO) algorithms are introduced in order to solve the optimal value of $K$ and $\alpha$ and improve the computational convergence.

According to the characteristics of entropy, it is possible to use it to evaluate the sparsity of a signal. The amount of entropy can reflect the uniformity of a probability distribution [19]. Based on the entropy, the minimum mean envelope entropy (MMEE) of IMF is the objective function of the VMD parameter optimization algorithm. This objective function can be expressed as in Equation (16):

$$\min_{K,\alpha} F(K,\alpha) = -\frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{N} p_i \log_2(p_i), \qquad p_i = h(i)/\sum_{i=1}^{N} h(i) \tag{16}$$

where $h$ is the envelope of the mode function $u_k$ generated by the Hilbert transform and $p_i$ is the probability at the $i$-th point of the envelope.

In order to improve convergence, an enhanced particle swarm optimization (EPSO) algorithm is proposed to solve the optimal parameters of VMD in this paper. In ordinary PSO, the velocity and position of each particle in PSO-updating rules are provided as in Equation (17) [20]:

$$\begin{cases} v_i(g+1 = wv_i(g) + c_1r_1(pbest_i - x_i(g) + c_2r_2(gbest - x_i(g)) \\ x_i(g+1 = x_i(g) + v_i(g+1) \end{cases} \tag{17}$$

where $pbest_i$ is the best solution of particle $i$, $gbest$ is the optimal solution of the entire swarm, $g$ is an iteration number, $v_i$ $(g)$ is velocity, $x_i$ $(g)$ is the position of particle $i$ at iteration $g$, $w$ is an inertia weight, $c_1$ and $c_2$ are learning factors, and $r_1$ and $r_2$ are random numbers in [0, 1].

In order to improve the global search ability, the EPSO adds mutation and jump operations in the search process. At each iteration, this algorithm can randomly determine the mutation number and position of an individual optimal solution *pbest* with a certain mutation probability, and this probability gradually decreases as the iterations proceed. Thus, the *pbest* updating rule for population number $N_p$ is defined as in Equation (18):

$$\begin{aligned} &pbest = \begin{cases} pbest, & r > p_{\mathrm{m}} \\ R_{pbest}, & r \le p_{\mathrm{m}} \end{cases} \\ &R_{pbest}(i) = \begin{cases} pbest(i), & i \notin M_{\mathrm{p}} \\ rand, & i \in M_{\mathrm{p}} \end{cases}, \qquad i = 1, 2, \cdots N_{\mathrm{p}} \end{aligned} \tag{18}$$

where $p_{\mathrm{m}}$ is the mutation probability, $M_{\mathrm{p}}$ is the randomly determined set of mutation particle positions in *pbest*, and $r$ is a random number in [0, 1].

At each iteration, the number of elements nm in the set $M_{\mathrm{p}}$ is generated randomly, and $n_m \le N_p/2$. Then, the mutation probability $p_{\mathrm{m}}$ is satisfied as in Equation (19):

$$p_{\mathrm{m}} = p_{\mathrm{m,max}} - \frac{(g-1)(p_{\mathrm{m,max}} - p_{\mathrm{m,min}})}{(g_{\max} - 1)} \tag{19}$$

where $p_{\mathrm{m,min}}$ is the minimum mutation probability, $p_{\mathrm{m,max}}$ is the maximum mutation probability, and $g_{\max}$ is the maximum iteration number.

In later iterations a large number of particles concentrates in a narrow area and the optimization result easily falls into local optimization, from which it is difficult to emerge. Compared with PSO, the multiple uniformly distributed particle populations are regenerated near the current optimal solution via the jump operation and the new optimal solution is obtained via multi-step iterative competition. In this paper, the later iteration is defined as $g > 0.7\, g_{\max}$. Meanwhile, the particle aggregation level is used as the trigger threshold in the jump operation, which can be expressed as in Equation (20):

$$\begin{aligned} &\rho = \tfrac{1}{N_{\mathrm{p}} d} \sum_{i=1}^{N_{\mathrm{p}}} \sum_{j=1}^{d} \eta(x_{i,j}(g)) \\ &\eta(x_{i,j}(g)) = \begin{cases} 1, & (\left|x_{i,j}(g) - gbest\right| / \left|gbest\right|) < \varepsilon \\ 0, & \text{others} \end{cases} \end{aligned} \tag{20}$$

where $\rho$ is the particle aggregation level and $\varepsilon$ is a small positive number.

If $\rho > \delta$, $N_J$ particle swarms are generated, the $h$-th particle swarm is as provided in Equation (21):

$$x_{i,j}^{h}(g+1) = \begin{cases} gbest + r(x_{\max}(g) - gbest), & (x_{\max}(g) - gbest) > (gbest - x_{\min}(g)) \\ gbest + r(gbest - x_{\min}(g)), & \text{others} \end{cases} \tag{21}$$

where $h$ is the number of the new generated particle swarms ($h$ = 1,2, . . . , $N_J$), $r$ is the random number in [0, 1], $x_{\max}(g)$ is the maximum value of $x_{i,j}$ at the $g$-th iteration, and $x_{\min}(g)$ is the minimum value of $x_{i,j}$ at the $g$-th iteration.

After $g_x$ iterations and optimizations of $N_J$ particle swarms by Equation (21), the population with the best solution continues to the next iteration to update the global best solution. The introduced random value within a range can increase the population diversity and avoid early maturity. A flowchart representing the EPSO optimization algorithm is shown in Figure 2.

### 3.2. Subseries Reconstruction Method (SR)

The structure of the traditional decomposition–combination wind speed forecasting model is shown in Figure 3. Here, each subseries must be modelled independently, which increases the model's complexity and calculation time. In order to improve the accuracy and efficiency of the forecasting model, this paper proposes a subseries reconstruction (SR) method to satisfy the accuracy and timeliness requirements for ultra-short-term wind speed forecasting. In SR, the sample entropy (SE) is introduced to quantify the EVMD subseries complexity, while the reconstruction method of adjacent subseries with approximate entropy is proposed to decrease the number of subseries.

Formally, the sample entropy of time series $\{x_i\}$ ($i$ = 1, . . . , $N$) can be expressed as in Equation (22):

$$En_{\mathrm{samp}} = -\ln\left[\frac{B^{m+1}(r)}{B^{m}(r)}\right] \tag{22}$$

where $m$ is the dimension of vector sequences, $r$ is the similarity tolerance, and $B^m(r)$ is the mean of the number of distances between all vectors less than or equal to $r$, which can be expressed as in Equation (23):

$$\begin{aligned} B^{m}(r) &= \tfrac{1}{N-m+1} \sum_{i=1}^{N-m+1} B_i^{m}(r) \\ B_i^{m}(r) &= \tfrac{1}{N-m} C\{\mathrm{d}[X(i), X(j)\big] < r\}, i = 1, 2, \cdots N-m+1, i \neq j \end{aligned} \tag{23}$$

where $X(i)$ is $m$ dimensional sequences composed of $\{x_i\}$ in order, $X(i) = [x(i), \ldots, x(i+m-1)]$ ($i = 1, 2, \ldots, N-m+1$), $\mathrm{d}[X(i), X(j)]$ are the absolute maximum distance between vector $X(i)$ and $X(j)$, and $C\{\cdot\}$ is a function that counts the number of vectors that satisfy the condition.

Based on Equation (23), $m$ and $r$ are critical in determining the outcome of sample entropy. In most conditions, the value of $m$ is 1 or 2, and $r$ is in [0.1SD, 0.25SD], where SD is the standard deviation of the original time series $\{x_i\}$ [21]. The sample entropy value of $\{x_i\}$ is related to the series complexity; the higher the series complexity, the higher the sample entropy value.

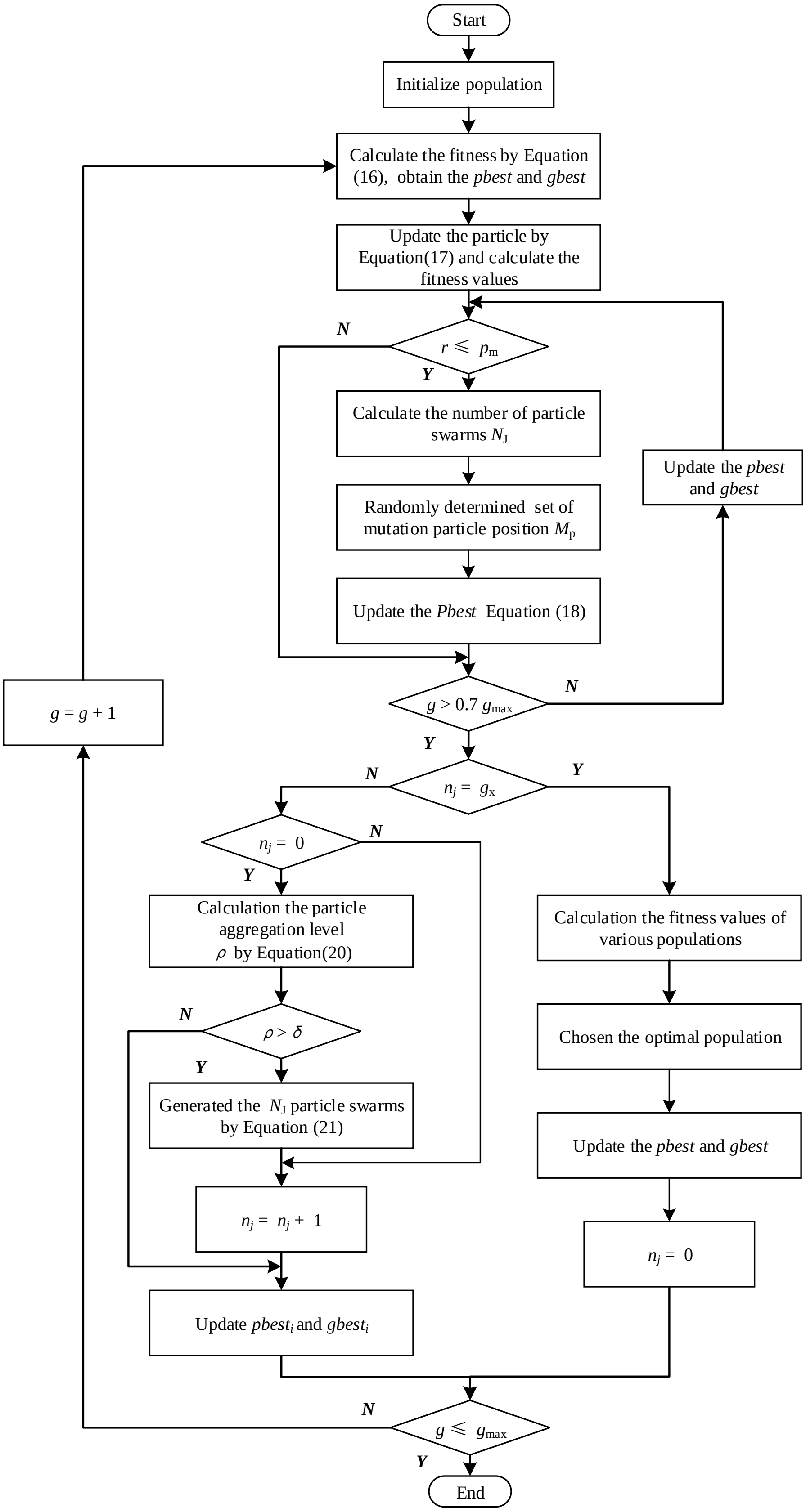

Start
Initialize population
Calculate the fitness by Equation (16), obtain the pbest and gbest
Update the particle by Equation(17) and calculate the fitness values
$r \leqslant p_m$
N
Y
Calculate the number of particle swarms $N_J$
Randomly determined set of mutation particle position $M_p$
Update the Pbest Equation (18)
Update the pbest and gbest
$g > 0.7\ g_{max}$
N
Y
$g = g + 1$
$n_j = g_x$
N
Y
$n_j = 0$
N
Y
Calculation the particle aggregation level $\rho$ by Equation(20)
$\rho > \delta$
N
Y
Generated the $N_J$ particle swarms by Equation (21)
$n_j = n_j + 1$
Update $pbest_i$ and $gbest_i$
Calculation the fitness values of various populations
Chosen the optimal population
Update the pbest and gbest
$n_j = 0$
$g \leqslant g_{max}$
N
Y
End

**Figure 2.** Flowchart of EPSO optimization algorithm.

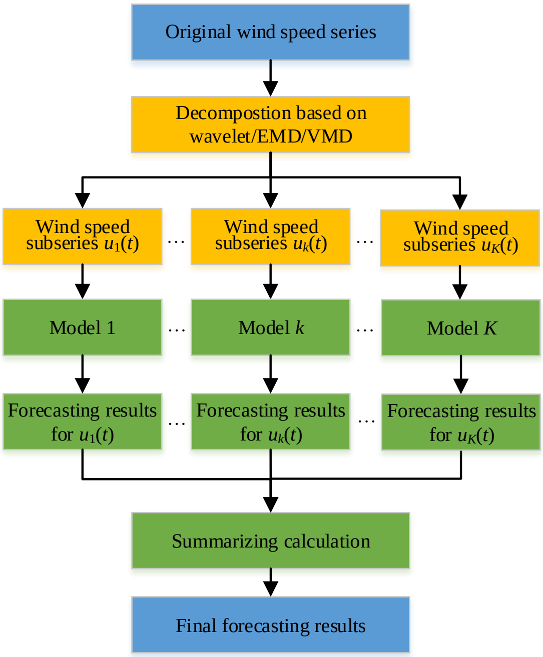

Original wind speed series
Decompostion based on wavelet/EMD/VMD
Wind speed subseries $u_1(t)$
…
Wind speed subseries $u_k(t)$
…
Wind speed subseries $u_K(t)$
Model 1
…
Model $k$
…
Model $K$
Forecasting results for $u_1(t)$
…
Forecasting results for $u_k(t)$
…
Forecasting results for $u_K(t)$
Summarizing calculation
Final forecasting results

**Figure 3.** Framework of traditional decomposition–combination wind speed forecasting model.

In the series reconstruction method, the complexity of EVMD subseries of the original wind speed time series is assessed by sample entropy and the subseries are sorted from lowest to highest according to entropy value. Setting the threshold of entropy, the EVMD subseries are adaptively divided into high-entropy subseries, $S_{\mathrm{H}}$, and low-entropy subseries, $S_{\mathrm{L}}$. Evaluating the entropy of adjacent high-entropy subseries, the adjacent subseries with approximate entropy values are merged, then a new group of reconstructed high-entropy subseries, *SrH*, is obtained. Meanwhile, the low-entropy subseries merge into another new subseries, *SrL*. Because the complexity of the merged subseries approximates the original subseries, this reconstructed subseries method can improve accuracy and efficiency. The framework of this method is shown in Figure 4.

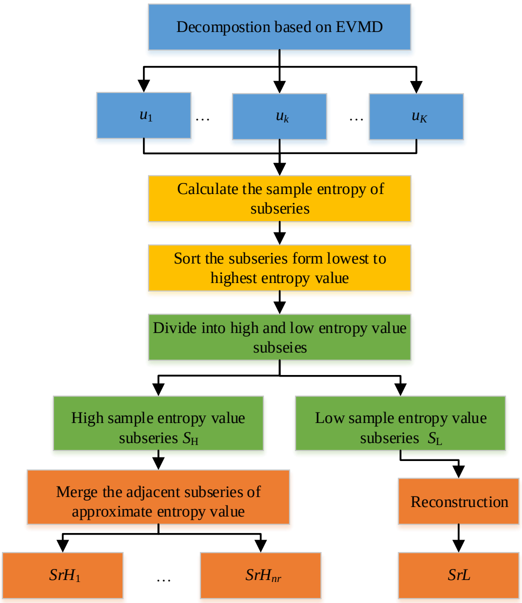

Decompostion based on EVMD
$u_1$
…
$u_k$
…
$u_K$
Calculate the sample entropy of subseries
Sort the subseries form lowest to highest entropy value
Divide into high and low entropy value subseies
High sample entropy value subseries $S_{\mathrm{H}}$
Low sample entropy value subseries $S_{\mathrm{L}}$
Merge the adjacent subseries of approximate entropy value
Reconstruction
$SrH_1$
…
$SrH_{nr}$
$SrL$

**Figure 4.** Framework of subseries reconstruction method.

### 3.3. Hybrid Wind Speed Forecasting Framework

The EVMD-SR-BLS-ARIMA hybrid wind speed forecasting model proposed in this paper mainly includes data on time–frequency analysis, subseries reconstruction, hybrid model construction, and linear superposition of prediction results in four major parts; the framework is shown in Figure 5. First, the EVMD is utilized to decompose the original wind speed time series in order to obtain $K$ subseries, thereby reducing irregularities. Second, SR is applied to reconstruct the subseries in order to decrease their number and improve forecasting efficiency. Subsequently, the BLS-ARIMA model is used to forecast different reconstructed subseries. Finally, the final forecasting result is obtained by linear superposition of the forecasting values of each subseries.

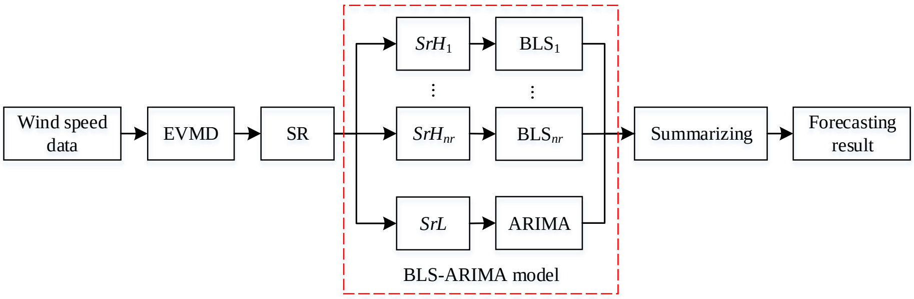


**Figure 5.** The framework of the EVMD-SR-BLS-ARIMA hybrid wind speed forecasting model.

As shown in Figure 6, the forecasting process of the EVMD-SR-BLS-ARIMA forecasting model is as follows:

Step 1: Initialize the number of forecasting cycles $k$, $k_u$, $n_0$, $n_1$, and $n_r$ to 0; set the input model wind speed time series length, $N_s$, and forecasting times, $M$.

Step 2: Using the EVMD method, the wind speed time series $[X(1 + k), \ldots, X(N_s + k)]$ is decomposed into K subseries $u(k)$; calculate the sample entropy of each subseries and find the minimum sample entropy, $En_{\text{samp,min}}$.

Step 3: Using $1.2En_{\text{samp,min}}$ as the threshold, divide the subseries $u(k)$ into the high-entropy subseries $S_H$ and low-entropy subseries $S_L$, then count the numbers of high-entropy subseries $n_0$ and low-entropy subseries $n_1$.

Step 4: The adjacent subseries with no more than 10% $En_{\text{samp}}$ variation in $S_H$ are reconstructed to obtain a new group of high-entropy subseries $SrH$ (the number of $SrH$ is $n_r$) and the forecasting results $\hat{Y}_B(i)$ ($I = 1, \ldots, n_r$) are calculated via the BLS model. The $n_1$ low-entropy subseries, $S_L$, merge into a new subseries, $SrL$, and an ARIMA model for $SrL$ is built to obtain the forecasting result $\hat{Y}_A$; then, the $k$-th wind speed forecasting result is $\hat{Y}(k) = \hat{Y}_A + \sum_{i=1}^{n_r} \hat{Y}_B(i)$.

Step 5: $k = k + 1$. Repeat steps 2–4 until $k = M$ is reached at the end of the forecasting process and the $M$-step forecasting results are obtained.

In order to improve forecasting accuracy, the error-corrected EVMD-SR-BLS-ARIMA model is developed to post-process the errors. This error-corrected model is essentially a two-stage forecasting model. The first stage is composed of EVMD-SR-BLS-ARIMA for wind speed basis data, which calculates the forecasting error calculated, while the second stage uses ARIMA to calculate the forecasting error of the EVMD- SR-BLS-ARIMA model. In the forecasting process, the second-level model is used to correct the first-level prediction data in order to obtain the final prediction result.

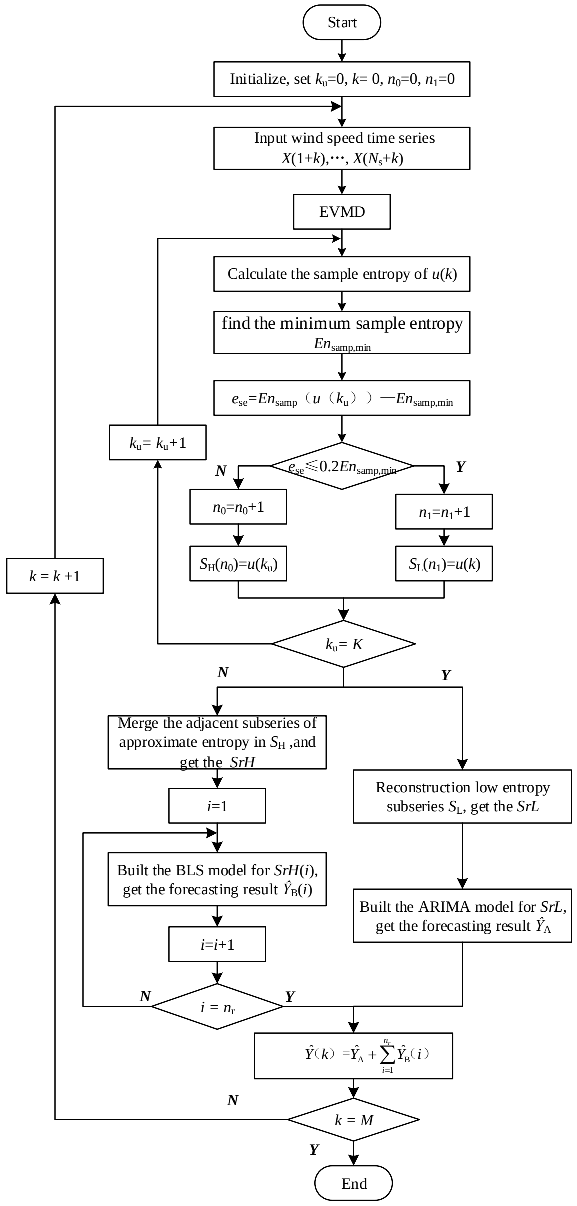

Start
Initialize, set $k_u$=0, $k$= 0, $n_0$=0, $n_1$=0
Input wind speed time series $X(1+k),\cdots, X(N_s+k)$
EVMD
Calculate the sample entropy of $u(k)$
find the minimum sample entropy $En_{\text{samp,min}}$
$e_{se}=En_{\text{samp}}(u(k_u))-En_{\text{samp,min}}$
$k_u= k_u+1$
N
$e_{se}\leqslant 0.2En_{\text{samp,min}}$
Y
$n_0=n_0+1$
$n_1=n_1+1$
$S_H(n_0)=u(k_u)$
$S_L(n_1)=u(k)$
$k = k+1$
$k_u= K$
N
Y
Merge the adjacent subseries of approximate entropy in $S_H$ ,and get the $SrH$
$i$=1
Reconstruction low entropy subseries $S_L$, get the $SrL$
Built the BLS model for $SrH(i)$, get the forecasting result $\hat{Y}_B(i)$
Built the ARIMA model for $SrL$, get the forecasting result $\hat{Y}_A$
$i$=$i$+1
N
$i = n_r$
Y
$\hat{Y}(k)=\hat{Y}_A+\sum_{i=1}^{n_r}\hat{Y}_B(i)$
N
$k = M$
Y
End

**Figure 6.** Flowchart of EVMD-SR-BLS-ARIMA for ultra-short-term wind speed forecasting.

### 3.4. Evaluation Index

In order to quantitatively evaluate the performance of the forecasting model, the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and symmetric Mean Absolute Percentage Error (sMAPE) are introduced. The discrepancy between anticipated and actual values is measured by these three indicators, with lesser values indicating better forecast accuracy. Respectively, the three evaluation indexes are defined as in Equations (24)–(26):

$$\delta_{\text{RMSE}}=\sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t-\hat{y}_t)^2} \tag{24}$$

$$\delta_{\text{MAE}}=\frac{1}{N}\sum_{t=1}^{N}|y_t-\hat{y}_t| \tag{25}$$

$$\delta_{\text{sMAPE}}=\frac{1}{N}\sum_{t=1}^{N}\frac{2|y_t-\hat{y}_t|}{y_t+\hat{y}_t}\times 100\% \tag{26}$$

where $y_t$ is the actual value at time $t$, $\hat{y}_t$ is the predicted value at time $t$, and $N$ is the observation size.

## 4. Case Study

In this section, original wind speed data from a Huaneng Group wind farm in Shandong province from 1 January 2016 were used to test the accuracy of the ultra-short-term wind speed forecasting method. The wind forecasting model was constructed using m-files in MATLAB R2016b running on a Windows PC with an Intel I5-8500 CPU.

### *4.1. Data Collection*

The sample data were based on actual test data from a 1.5 MW wind turbine supervisory control and data acquisition (SCADA) system with a the sampling period of 10 min (24 h total, 144 sampling points $[X_1, \ldots, X_{144}]$). The sample wind speed time series is shown in Figure 7. The average wind speed in the sample was 6.72 m/s, the maximum wind speed was 11.80 m/s, and the minimum wind speed was 2.08 m/s. The wind speed sequence fluctuated noticeably with no discernible pattern. In this case study, the first 100 sampling points (sampling points 1~100) of the wind speed series were utilized as modeling data, while the next 44 points (sampling points 101~144) were used as test samples to evaluate the prediction model's accuracy and timeliness.

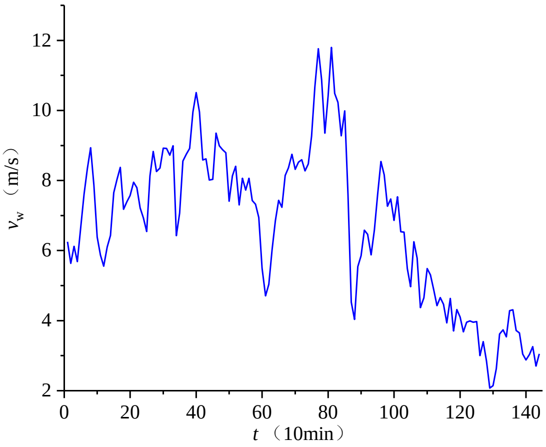


**Figure 7.** Sample of wind speed time series.

### *4.2. Experiment 1: EVMD-BLS Forecasting Model*

Experiment 1 mainly studied the ultra-short-term wind speed forecasting performance of the EVMD-BLS decomposition–combination model. In this model, the EVMD divides the wind speed time series into subseries with simple frequency components and more regular variation, creates the BLS models of each subseries, and then constructs the error correction model for increased accuracy. Meanwhile, when the wind speed time series $N_s$ in the input model is set to 70, the first 100 wind speed samples are divided into two parts: the first 70 data points are used for wind speed forecasting, and the last 30 are used for error modeling.

According to the method introduced in Section 3.1, the EPSO algorithm was used to solve the optimal values of $K$ and $\alpha$ in the EVMD method. In order to prevent excessive variation from destroying the algorithm's convergence efficiency, the size of the set of variation particles, $M_p$, was limited to within 50% of the population particle number $N_p$. Taking into account the algorithm's computational efficiency, the start point of the jump operation was set to 70% of the maximum number of iterations. Additionally, the particle aggregation level calculation limit $\varepsilon$ in Equation (20) was set to 0.01, the aggregation trigger limit, $\delta$, was 0.5, IMF $K$ was an integer between 1 and 10, and the quadratic penalty term $\alpha$ was a value between 1 and 50. The EPSO optimization algorithm parameters were set as shown in Table 1.

**Table 1.** Parameters of the EPSO optimization algorithm.

| Parameter | $N_p$ | $w_{min}$ | $w_{max}$ | $c_1$ | $c_2$ | $p_{m,min}$ | $p_{m,max}$ |
|---|---|---|---|---|---|---|---|
| Value | 30 | 0.4 | 0.9 | 1.497 | 1.497 | 0.1 | 0.6 |

Figure 8 depicts the variation curve of the minimum average envelope entropy value in the EPSO optimal value search with the number of iterations, $g$; the parameters of variational mode decomposition are then determined as $K = 6$ and $\alpha = 5.67$.

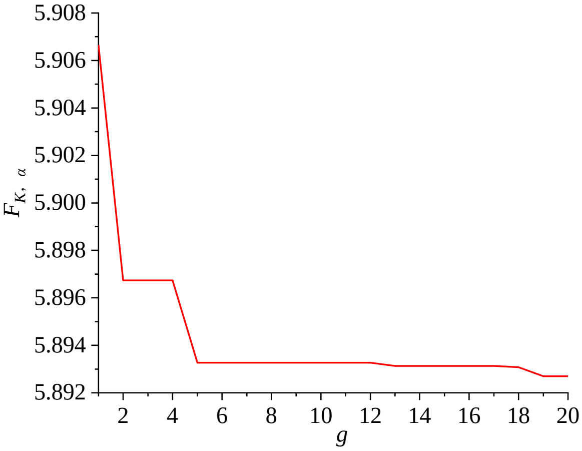


**Figure 8.** Iterative process of variational mode decomposition parameter optimization.

The first 70 wind speed data points were subjected to EVMD to generate six mode functions ($u_1, \ldots, u_6$), as shown in Figure 9. In order to compare the validity of decomposition, the wind speed data were decomposed using ensemble empirical mode decomposition (EEMD). When the standard deviation of Gaussian white noise was set to 0.4 and the noise order was set to 50 in EEMD, the wind speed data were decomposed into five IMF components and one residual component, $r$, as shown in Figure 10.

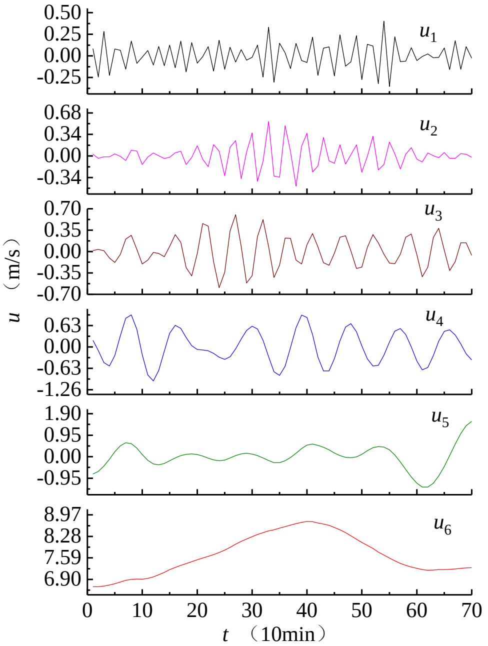


**Figure 9.** EVMD results of the first 70 points in the wind speed series.

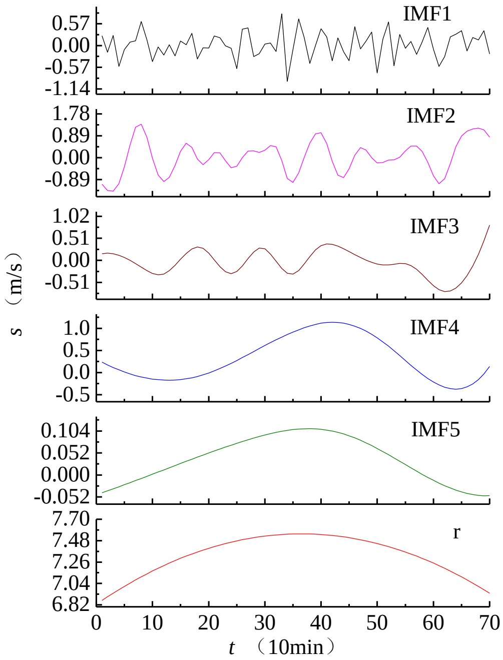


**Figure 10.** Ensemble empirical mode decomposition results of the first 70 points of the wind speed series.

Analyzing the characteristics of each component of EVMD and EEMD, the highest frequency component's amplitude in EVMD is between [−0.35, 0.40], with a component mean of −0.01, and the highest frequency component's amplitude in EEMD is between [−0.93, 0.83], with a component mean of 0.03. Compared with EEMD, it is evident that EVMD decomposes the high-frequency component more completely. When the waveforms of the lowest frequency component generated by the two approaches are compared, the lowest frequency component of EVMD can better depict the long-term fluctuation features of the wind speed.

The BLS forecasting model was used for each EVMD subseries of the first 70 wind speed data points, and the forecasting result at the 71st point was derived. We updated the input data and performed the EVMD-BLS forecasting procedure 30 times to obtain the expected wind speeds at the 71st to 100th points, calculate the forecast error at the 71st to 100th points, and obtain the forecasting error series. The ARIMA model for the error series was constructed in order to calculate the error forecasting value at the 101st point. Meanwhile, the corrected forecasting result of the 101st point was obtained by superimposing the 101st point of EVMD-BLS and the error forecasting results for the same point.

To validate the prediction performance of the EVMD-BLS model, the ARIMA, BP, EEMD-SVM, EEMD-DBN, EEMD-BLS, and EVMD-DBN models were employed for comparison.

In this paper, the mapping features of the BLS model are set to 30, each mapping feature contains 100 nodes, and the enhanced nodes are set to 300. The DBN network structure is set to 100-500-500-100. The simulation experiment employed these seven models to predict ultra-short-term wind speed at the 101st to 144th points, with the results are presented in Figure 11.

The wind speed forecast results of the seven models are generally similar to the actual wind speed fluctuation trends, and EEMD-DBN, EEMD-BLS, EVMD-DBN, and EVMD-BLS are better at tracking actual wind speed variation. We used the RMSE, MAE, and sMAPE to evaluate the forecasting performance of these models; the results are shown in Table 2.

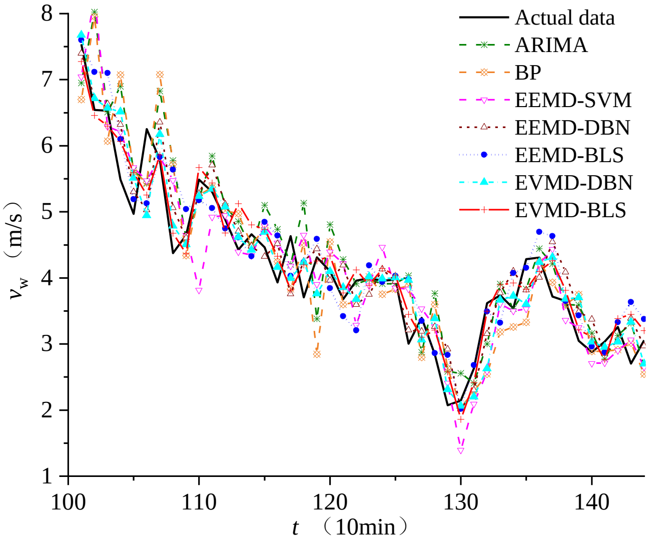


**Figure 11.** Wind speed forecasting results of seven different models.

**Table 2.** Evaluation result of seven forecasting models.

| Model | $\delta_{\mathbf{RMSE}}$ (m/s) | $\delta_{\mathbf{MAE}}$ (m/s) | $\delta_{\mathbf{sMAPE}}$ (%) | $t$ (s) |
|---|---|---|---|---|
| ARIMA | 0.64 | 0.54 | 12.19 | 91.07 |
| BP | 0.68 | 0.53 | 12.77 | 101.85 |
| EEMD-SVM | 0.62 | 0.48 | 12.24 | 226.43 |
| EEMD-DBN | 0.46 | 0.37 | 9.49 | 478.99 |
| EEMD-BLS | 0.49 | 0.37 | 9.20 | 203.01 |
| EVMD-DBN | 0.46 | 0.34 | 8.85 | 235.92 |
| EVMD-BLS | 0.38 | 0.31 | 7.85 | 190.45 |

In comparing the evaluation index of the seven models, the forecasting models using time–frequency analysis perform significantly better than the ARIMA and BP models. Among the three EEMD forecasting models, the EEMD-DBN and EEMD-BLS models outperform the EEMD-SVM model, while the EEMD-BLS model has the faster running speed. In both of the models using DBN, the EVMD-DBN model has higher prediction accuracy and efficiency than the EEMD-DBN model; however, the amplitude of the high-frequency component of the EEMD is large, and affects overall prediction accuracy. Furthermore, the Gaussian white noise introduced by the EEMD in signal decomposition increases computation time. Compared to the EVMD-DBN model, the EVMD-BLS model has higher wind speed forecasting accuracy and can greatly reduce computing time. Statistical analysis of the forecasting errors of the seven models was performed, and the results are presented in Figure 12.

For the seven combined models, the distribution of ultra-short-term wind speed prediction errors is as follows: (1) The percentage of prediction errors in the (−0.5, 0.5) interval is 29.54% for the ARIMA model, 29.54% for the EEMD-SVM model, 38.63% for the EEMD-DBN model, 43.18% for the EEMD-BLS model, 50% for the EVMD-DBN model, and 45.45% for the EVMD-BLS model; (2) The percentage of prediction errors in the (−1.0, 1.0) interval is 81.81% for the ARIMA model, 86.36% for the EEMD-SVM model, 93.18% for the EEMD-DBN model, 90.91% for the EEMD-BLS model, 93.18% for the EVMD-DBN model, and 100% for the EVMD-BLS model. Therefore, the prediction errors of the EVMD-BLS model are all less than 1 m/s, and the computing time is short.

The Diebold–Mariano (DM) test is introduced to further evaluate the validity of the forecasting model in this paper. The DM test results of the seven forecasting models are shown in Table 3.

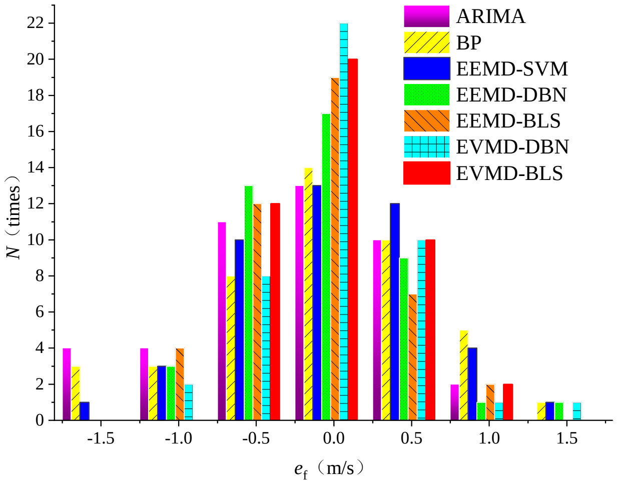


**Figure 12.** Forecasting-error statistics of seven models.

**Table 3.** Evaluation result of forecasting models.

| | | **BP** | **EEMD-SVM** | **EEMD-DBN** | **EEMD-BLS** | **EVMD-DBN** | **EVMD-BLS** |
|---|---|---|---|---|---|---|---|
| ***p*-value** | ARIMA | 0.647 | 0.597 | 0.018 | 0.03 | 0.015 | 0.003 |
| | BP | / | 0.476 | 0.015 | 0.036 | 0.007 | 0.003 |
| | EEMD-SVM | / | / | 0.107 | 0.17 | 0.102 | 0.021 |
| | EEMD-DBN | / | / | / | 0.445 | 0.986 | 0.014 |
| | EEMD-BLS | / | / | / | / | 0.559 | 0.034 |
| | EVMD-DBN | / | / | / | / | / | 0.088 |
| **DM-value** | ARIMA | −0.461 | 0.533 | 2.467 | 2.239 | 2.524 | 3.195 |
| | BP | / | 0.719 | 2.547 | 2.167 | 2.807 | 3.195 |
| | EEMD-SVM | / | / | 1.646 | 1.397 | 1.67 | 2.389 |
| | EEMD-DBN | / | / | / | −0.771 | −0.017 | 2.565 |
| | EEMD-BLS | / | / | / | / | 0.589 | 2.191 |
| | EVMD-DBN | / | / | / | / | / | 1.744 |

According to the DM test method, the tested model's accuracy is better than the reference model when the *p*-value is <0.05 and the DM value is >0. As shown in Table 3, except for the BP model, the forecasting accuracy of ARIMA is lower than the other five models. In particular, the EVMD-BLS method proposed in this paper significantly outperforms the ARIMA, BP, EEMD-SVM, EEMD-DBN, and EEMD-BLS models. Compared with the EVMD-DBN model, the *p*-value of 0.088 from the DM test of the EVMD-BLS model is very close to the threshold of significant difference (0.05), and its time, error statistics, and three evaluated indexes are better than those of EVMD-DBN.

### 4.3. Experiment 2: EVMD-SR-BLS-ARIMA Hybrid Forecasting Model

According to the method introduced in Sections 3.2 and 3.3, the EVMD-SR-BLS-ARIMA hybrid model was used to forecast the ultra-short-term wind speed of wind farms, with the 101st point in the forecasting process used as an example to demonstrate the modeling.

With the input wind speed data length $N_s$ set to 100, the input wind speed data are $[X_1, \ldots, X_{100}]$.

The EPSO algorithm was used to solve the optimization parameters $K$ and $\alpha$ as 7 and 6.40, respectively, and the input data were decomposed by the EVMD to obtain seven subseries ($u_1, \ldots, u_7$), as shown in Figure 13a. Using sample entropy (SE) to evaluate the complexity of the decomposed subseries, the SE evaluation parameters $m$ and $r$ were set to 2 and 0.2 SD (standard deviation), respectively; the SE results for each subseries are presented in Figure 14.

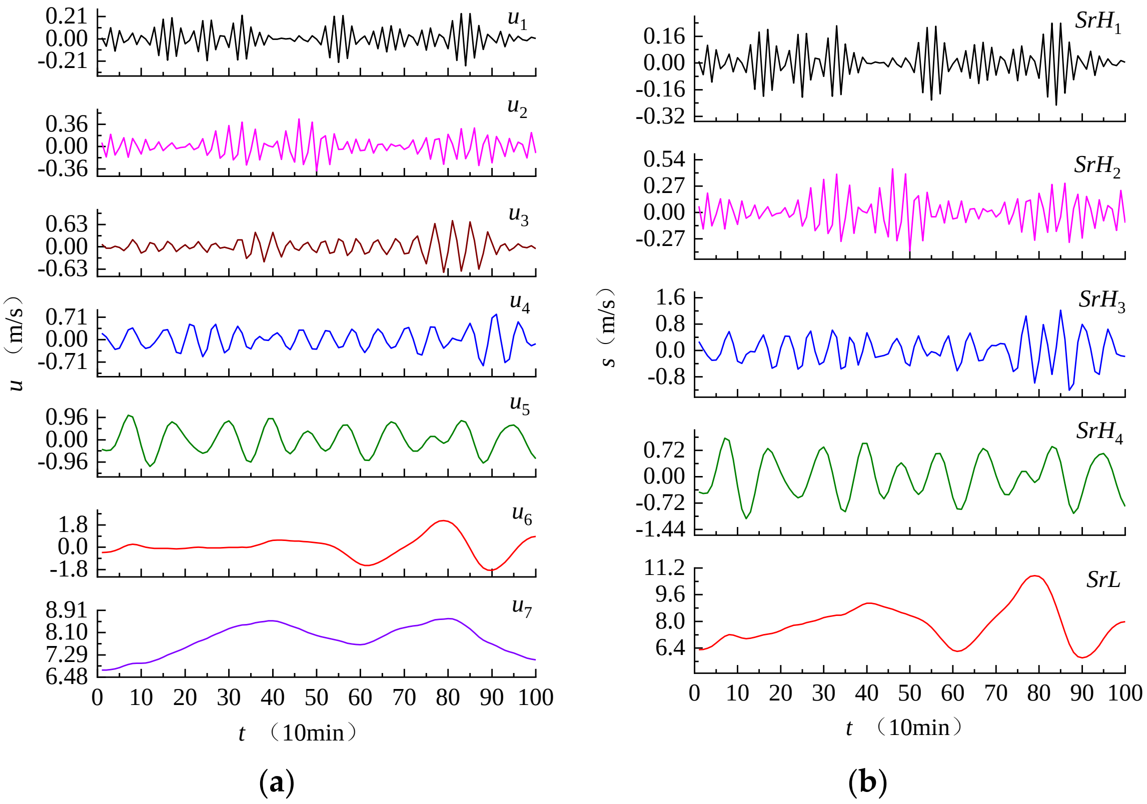


**Figure 13.** Decomposition result of the first 100 points in the wind speed series: (**a**) result of EVMD and (**b**) result of subseries reconstruction.

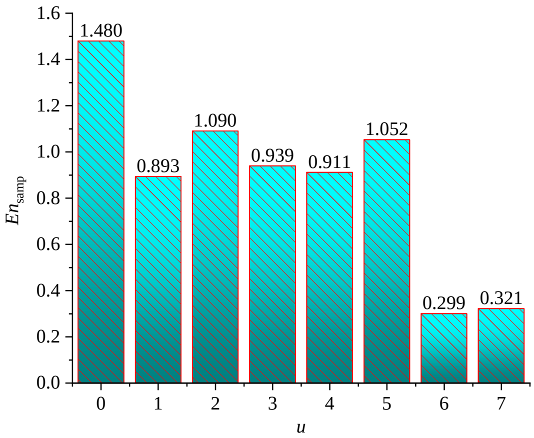


**Figure 14.** Subseries sample entropy of EVMD.

In Figure 14, the horizontal axis $u$ value of 0 represents the sample entropy of the undecomposed wind speed data, $u_0$. The sample entropy values of each subseries $u_1$~$u_7$ are 60.32%, 73.68%, 63.47%, 61.56%, 71.10%, 20.22% and 21.69% of the sample entropy value of $u_0$, respectively, and the minimum sample entropy, $En_{samp,min}$, is 0.299. The high-entropy-value subseries were screened as $u_1$~$u_5$ and the low-entropy-value sequences as u6 and $u_7$ using the method shown in Figure 6, with 1.2 $En_{samp,min}$ as the threshold. The subseries $u_6$ and $u_7$ were reconstructed as the new subseries $SrL$, and the adjacent high-entropy subseries were reconstructed with the requirement that the change in $En_{samp}$ did not exceed 10%. The reconstructed subseries are shown in Figure 13b. The nascent subsequence $SrH_1$ is $u_1$, $SrH_2$ is $u_2$, $SrH_3$ is $u_3$ combined with $u_4$, $SrH_4$ is $u_5$, and $SrL$ is $u_6$ fused with $u_7$. In the EVMD-SR-BLS-ARIMA model, the forecasting results of the nascent high-entropy subseries *SrH* are calculated by the BLS model, while the ARIMA model obtains the forecasting result of the new low-entropy subseries.

The simulation experiment employed the EVMD-SR-BLS-ARIMA model to predict ultra-short-term wind speed at the 101st to 144th points, with the results presented in Figure 15. The green dotted lines with triangle symbols represent the EVMD-SR-BLS-ARIMA model's forecasting results, while the red dashed lines with dots represent the results of the error-corrected EVMD-SR-BLS-ARIMA model. Wind speed variations can be tracked with excellent precision using these two models; their forecasting evaluation results are shown in Table 4.

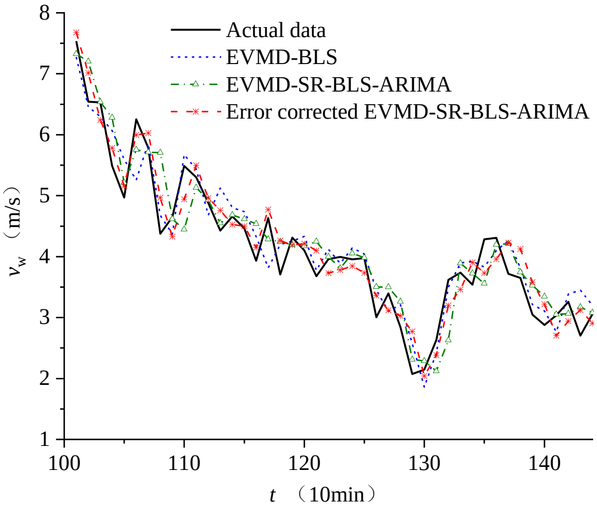


**Figure 15.** Wind speed forecasting results of EVMD-SR-BLS-ARIMA model.

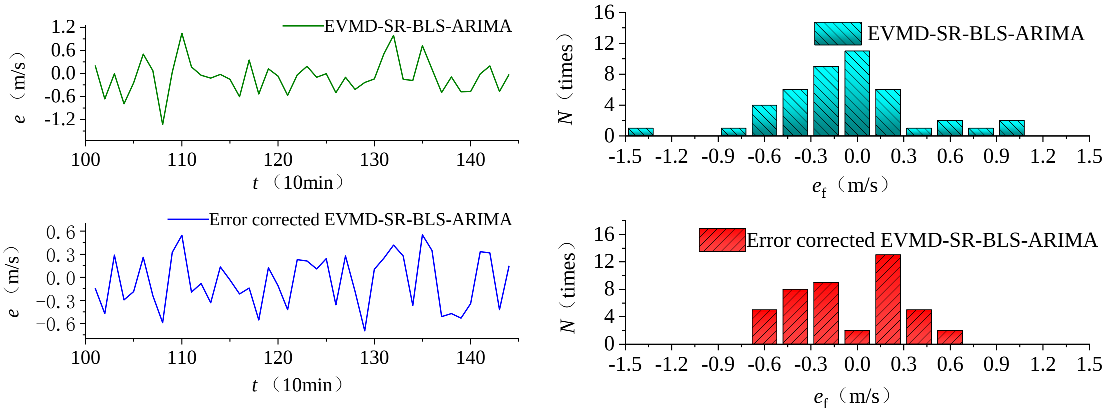
**Table 4.** Evaluation indexes by EVMD-SR-BLS-ARIMA model.

| Model | $\delta_{\text{RMSE}}$ (m/s) | $\delta_{\text{MAE}}$ (m/s) | $\delta_{\text{sMAPE}}$ (%) | $t$ (s) |
|---|---|---|---|---|
| EVMD-SR-BLS-ARIMA | 0.45 | 0.32 | 8.25 | 126.94 |
| Error-corrected EVMD-SR-BLS-ARIMA | 0.34 | 0.30 | 7.99 | 172.67 |

The forecasting accuracy of the EVMD-SR-BLS-ARIMA model is similar to that of the EVMD-BLS model, and is better than the other models, as shown in Table 3. Due to the complexity of evaluating the wind speed decomposition subseries using sample entropy, the low-entropy series is passed to the ARIMA model for processing, and the adjacent approximate high-entropy series are reconstructed to reduce the model size and computational time.

The model calculation time is decreased from 190.45 s in the EVMD-BLS model to 126.94 s, which is just 66.65% of the EVMD-BLS model's calculation time. The $\delta_{\text{RMSE}}$, $\delta_{\text{MAP}}$, and $\delta_{\text{MAPE}}$ of the error-corrected EVMD-SR-BLS-ARIMA model are 0.34 m/s, 0.30 m/s, and 8.38%, respectively, and its computation time is 90.66% of that of the EVMD-BLS model. To summarize, the EVMD-SR-BLS-ARIMA model can provide an excellent balance of accuracy and efficiency in wind speed forecasting.

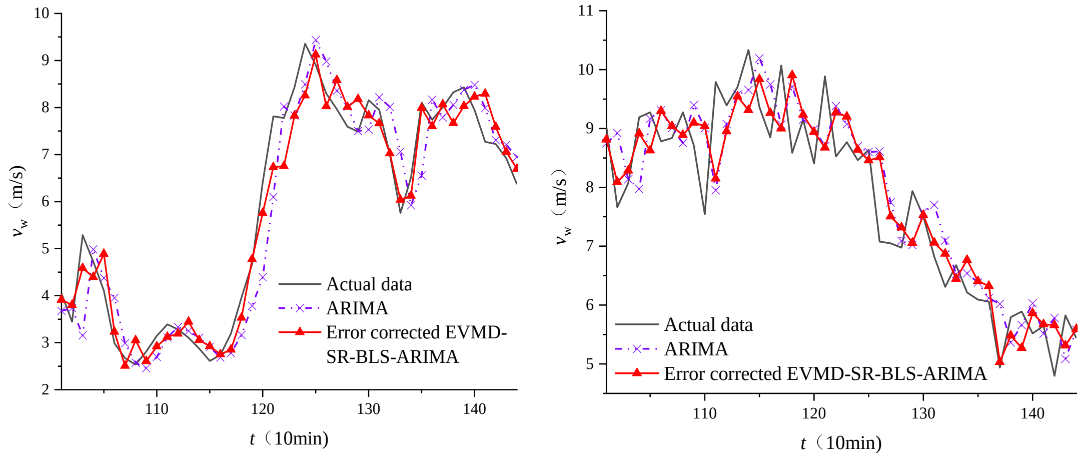
The statistical analysis results of the forecasting error of the two reconstructed models are shown in Figure 16. It can be observed that (1) the EVMD-SR-BLS-ARIMA model's forecasting error is [−1.33, 1.04], the probability that the forecasting error is in the (−1.0, 1.0) interval is 93.18%, and the probability that the forecasting error is in the (−0.5, 0.5) interval is 68.18%; (2) the error-corrected EVMD-SR-BLS-ARIMA model's forecasting error is [−0.68, 0.55], and the probability that the forecasting error is in the (−0.5, 0.5) interval is 84.09%.



(**a**) (**b**)

**Figure 16.** Forecasting errors and statistics of EVMD-SR-BLS-ARIMA models: (**a**) forecasting error and (**b**) error statistics.

This paper used two different days of wind speed data from the same wind farm (collected on 6 and 13 January 2016) to verify the applicability of the model; the results are shown in Figure 17. Compared with the ARIMA model, the *p*-value and DM-value of the error-corrected EVMD-SR-BLS-ARIMA model are 0.045 and 2.057, and its $\delta_{\text{sMAPE}}$ is 7.41% less than that of ARIMA (10.58%) for 6 January 2016; the *p*-value and DM-value of the error-corrected EVMD-SR-BLS-ARIMA model are 0.033 and 2.20, and its $\delta_{\text{sMAPE}}$ is 6.62% less than that of ARIMA (7.73%) for 13 January 2016. Hence, the proposed model has better applicability.



(**a**) (**b**)

**Figure 17.** Wind speed forecasting results at different times: (**a**) 6 January 2016 and (**b**) 13 January 2016.

According to the above experimental simulation results, it can be observed that (1) the proposed model has higher prediction accuracy and stronger prediction performance; and (2) the proposed model can significantly improve computational operation efficiency and reduce the time required to completion.

## 5. Conclusions

In view of the fact that actual wind speed is affected by weather, temperature, and random factors, there are a large number of outliers in wind speed prediction and it is strongly nonlinear. In this study, a hybrid intelligent forecasting model for ultra-short-term wind speed forecasting based on EVMD, SR, and BLS is proposed. The conclusions of this paper can be listed as follows:

- EVMD was used to decompose the wind speed time series, while the minimum mean envelope entropy (MMEE) and enhanced particle swarm optimization (EPSO) algorithms were introduced to attain the optimal values of $K$ and $\alpha$ of EVMD.
- SR was applied to recombine the subseries obtained by EVMD. The subseries of EVMD were adaptively divided into high-entropy and low-entropy subseries. Adjacent high-entropy subseries of approximate entropy values were merged to obtain a new group of reconstructed high-entropy subseries, and the low-entropy subseries were merged into a new subseries as well.
- The EVMD-SR-BLS-ARIMA hybrid wind speed forecasting model was constructed to obtain the reconstructed subseries forecasting results. Experimental results showed that the proposed method can significantly improve the forecasting accuracy and reduce the time required.

In future research, we intend to select more appropriate models for wind speed datasets with different features, investigate forecasting results with a nonlinear weighted combination of components, improve forecasting accuracy characteristics from the standpoint of BLS structure, optimize this model to carry out wind power forecasting, and identify the optimal configuration of the energy storage system capacity of wind power systems.

**Author Contributions:** Conceptualization, M.P.; methodology, M.P. and L.Z.; software, M.P. and L.Z.; validation, M.P., A.Z. and J.D.; formal analysis, M.P. and A.Z.; data curation, Y.Z.; writing—original draft preparation, M.P., Y.Z. and A.Z.; writing—review and editing, M.P. and Z.D.; visualization, M.P. and Y.Z.; supervision, M.P.; project administration, M.P.; funding acquisition, M.P., L.Z., A.Z., J.D. and Z.D. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Gansu Science and Technology Project in China, grant number 21JR1RA245, grant number 21JR1RA253, and grant number 20JR5RA382; Young Science Foundation of Lanzhou Jiaotong University, grant number 2020041; Natural Science Basic Research Plan in Shaanxi Province of China, grant number 2022JQ-397; and the Tianyou innovation team of Lanzhou Jiaotong University, grant number TY202004.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Liu, H.; Chen, C. Data Processing Strategies in Wind Energy Forecasting Models and Applications: A Comprehensive Review. *Appl. Energy* **2019**, *249*, 392–408. [CrossRef]
2. Zuo, H.; Bi, K.; Hao, H. A State-of-the-Art Review on the Vibration Mitigation of Wind Turbines. *Renew. Sustain. Energy Rev.* **2020**, *121*, 109710. [CrossRef]
3. Tascikaraoglu, A.; Uzunoglu, M. A Review of Combined Approaches for Prediction of Short-Term Wind Speed and Power. *Renew. Sustain. Energy Rev.* **2014**, *34*, 243–254. [CrossRef]
4. Okumus, I.; Dinler, A. Current Status of Wind Energy Forecasting and a Hybrid Method for Hourly Predictions. *Energ. Convers. Manag.* **2016**, *123*, 362–371. [CrossRef]
5. Tascikaraoglu, A.; Sanandaji, B.M.; Poolla, K.; Varaiya, P. Exploiting Sparsity of Interconnections in Spatio-Temporal Wind Speed Forecasting Using Wavelet Transform. *Appl. Energy* **2016**, *165*, 735–747. [CrossRef]
6. Wang, H.Z.; Wang, G.B.; Li, G.Q.; Peng, J.C.; Liu, Y.T. Deep Belief Network Based Deterministic and Probabilistic Wind Speed Forecasting Approach. *Appl. Energy* **2016**, *182*, 80–93. [CrossRef]
7. Liu, H.; Tian, H.; Li, Y. An Emd-Recursive Arima Method to Predict Wind Speed for Railway Strong Wind Warning System. *J. Wind Eng. Ind. Aerod.* **2015**, *141*, 27–38. [CrossRef]
8. An, N.; Zhao, W.; Wang, J.; Shang, D.; Zhao, E. Using Multi-Output Feedforward Neural Network with Empirical Mode Decomposition Based Signal Filtering for Electricity Demand Forecasting. *Energy* **2013**, *49*, 279–288. [CrossRef]
9. Huang, Y.; Yang, L.; Liu, S.; Wang, G. Multi-Step Wind Speed Forecasting Based on Ensemble Empirical Mode Decomposition, Long Short Term Memory Network and Error Correction Strategy. *Energies* **2019**, *12*, 1822. [CrossRef]
10. Sun, B.; Yao, H. Short-Term Wind Speed Forecasting Based on Local Mean Decomposition and Multi-Kernel Support Vector Machine. *Acta Energ. Solaris Sin.* **2013**, *34*, 1567–1573.
11. Dragomiretskiy, K.; Zosso, D. Variational Mode Decomposition. *IEEE Trans. Signal. Process* **2014**, *62*, 531–544. [CrossRef]

12. Sun, W.; Gao, Q. Short-Term Wind Speed Prediction Based on Variational Mode Decomposition and Linear–Nonlinear Combination Optimization Model. *Energies* **2019**, *12*, 2322. [CrossRef]
13. Han, L.; Zhang, R.; Wang, X.; Bao, A.; Jing, H. Multi-Step Wind Power Forecast Based on Vmd-Lstm. *Iet Renew. Power Gen.* **2019**, *13*, 1690–1700. [CrossRef]
14. Zhu, L.; Lian, C. *Wind Speed Forecasting Based on a Hybrid Emd-Bls Method*; Institute of Electrical and Electronics Engineers Inc.: Hangzhou, China, 2019; pp. 2191–2195.
15. Zhu, L.; Lian, C.; Zeng, Z.; Su, Y. A Broad Learning System with Ensemble and Classification Methods for Multi-Step-Ahead Wind Speed Prediction. *Cogn. Comput.* **2020**, *12*, 654–666. [CrossRef]
16. Bai, Y.; Liu, M.; Ding, L.; Ma, Y. Double-Layer Staged Training Echo-State Networks for Wind Speed Prediction Using Variational Mode Decomposition. *Appl. Energy* **2021**, *301*, 117461. [CrossRef]
17. Chen, C.L.P.; Liu, Z. Broad Learning System: An Effective and Efficient Incremental Learning System without the Need for Deep Architecture. *IEEE Trans Neur. Net. Learn.* **2018**, *29*, 10–24. [CrossRef] [PubMed]
18. Zhang, G.; Liu, H.; Zhang, J.; Yan, Y.; Zhang, L.; Wu, C.; Hua, X.; Wang, Y. Wind Power Prediction Based on Variational Mode Decomposition Multi-Frequency Combinations. *J. Mod. Power Syst. Clean* **2019**, *7*, 281–288. [CrossRef]
19. Tang, G.; Wang, X. Parameter Optimized Variational Mode Decomposition Method with Applicaltion to Incipient Fault Diagnosis of Rolling Bearing. *J. Xi'an Jiaotang Univ.* **2015**, *49*, 73–81.
20. Ramadan, H.S.; Bendary, A.F.; Nagy, S. Particle Swarm Optimization Algorithm for Capacitor Allocation Problem in Distribution Systems with Wind Turbine Generators. *Int. J. Electr. Power Energy Syst.* **2017**, *84*, 143–152. [CrossRef]
21. Alcaraz, R.; Rieta, J.J. A Review on Sample Entropy Applications for the Non-Invasive Analysis of Atrial Fibrillation Electrocardiograms. *Biomed. Signal. Proces.* **2010**, *5*, 1–14. [CrossRef]