Article:
Yang, R, Liu, H, Nikitas, N et al. (2022) Short-term wind speed forecasting using deep reinforcement learning with improved multiple error correction approach. Energy, 239 (Part B). 122128. ISSN: 0360-5442

https://doi.org/10.1016/j.energy.2021.122128

Short-term wind speed forecasting using deep reinforcement learning with improved multiple error correction approach

Rui Yang[a], Hui Liu*[a], Nikolaos Nikitas[b], Zhu Duan[a], Yanfei Li[c], Ye Li[a]

a. Institute of Artificial Intelligence and Robotics (IAIR), Key Laboratory of Traffic Safety on Track of Ministry of Education, School of Traffic and Transportation Engineering, Central South University, Changsha 410075, Hunan, China

b. School of Civil Engineering, University of Leeds, LS2 9JT, Leeds, UK

c. School of Mechatronic Engineering, Hunan Agricultural University, Changsha 410128, Hunan, China

Abstract

The safe and stable operation of wind power systems requires the support of wind speed prediction. To ensure the controllability and stability of smart grid dispatching, a novel hybrid model consisting of data-adaptive decomposition, reinforcement learning ensemble, and improved error correction is established for short-term wind speed forecasting. In decomposition module, empirical wavelet transform algorithm is used to adaptively disassemble and reconstruct the wind speed series. In ensemble module, Q-learning is utilized to integrate gated recurrent unit, bidirectional long short-term memory, and deep belief network. In error correction module, wavelet packet decomposition and outlier-robust extreme learning machine are combined to developing predictable components. An appropriate correction shrinkage rate is used to obtain the best correction effect. Ljung-Box Q-Test is utilized to judge the termination of the error correction iteration. Four real data are utilized to validate model performance in the case study. Experimental results show that: (a) The proposed hybrid model can accurately capture the changes of wind data. Taking 1-step prediction results as an example, the mean absolute errors for site #1, #2, #3, and #4 are 0.0829m/s, 0.0661m/s, 0.0906m/s, and 0.0803m/s, respectively; (b) Compared with several state-of-the-art models, the proposed model has the best prediction performance.

Keywords: Short-Term Wind Speed Prediction; Adaptive Data Decomposition; Q-Learning Ensemble Strategy; Improved Multiple Error Correction Technique.

Highlights

- A reinforcement learning ensemble strategy is developed to determine the best model fusion weights.
- Three deep learning benchmark predictors suitable for different environments are combined to learn the law of wind speed changes.
- An improved multiple error correction technique based on decomposition-prediction is proposed.
- A correction shrinkage rate is introduced to reduce the risk of overfitting.

Abbreviations

AI	Artificial Intelligence	**LSTM**	Long Short-Term Memory
ARMA	Auto-Regression Moving Averaging	**LBQ-test**	Ljung-Box Q-Test
ARIMA	Auto-Regressive Integrated Moving Average	**MPDQEM**	Multi-Predictor Deep Q Ensemble Model
ANN	Artificial Neural Network	**MPDQDEM**	Multi-Predictor Deep Q Decomposition Ensemble Model
AM-FM	Amplitude-Modulation and Frequency-Modulation	**MODWPT**	Maximal Overlap Discrete Wavelet Packet Transform
BiLSTM	Bidirectional Long Short-Term Memory	**MAE**	Mean Absolute Error
BPNN	Back Propagation Neural Network	**MAPE**	Mean Absolute Percentage Error
BFGS	Broyden Fletcher Goldfarb Shanno	**MOGOA**	Multi-Objective Grasshopper Optimization Algorithm
BiGRUNNs	Bidirectional Gated Recurrent Unit Neural Networks	**MOMVO**	Multi-Objective Multi-Verse Optimization
BSA	Backtracking Search Algorithm	**MMAdapGA**	Multiple Mutations Adaptive Genetic Algorithm
CNN	Convolutional Neural Network	**MEC**	Multiple Error Correction
CEEMD	Complete Ensemble Empirical Mode Decomposition	**MDP**	Markov Decision Processes
DBN	Deep Belief Network	**NWP**	Numerical Weather Prediction
DBM	Deep Boltzmann Machine	**NSGA-II**	Non-dominated Sorting Genetic Algorithm II
ELM	Extreme Learning Machine	**NAR**	Nonlinear Auto Regressive
EEMD	Ensemble Empirical Mode Decomposition	**ORELM**	Outlier-Robust Extreme Learning Machine
EWT	Empirical Wavelet Transform	**PSOGSA**	Particle Swarm Optimization and Gravitational Search Algorithm
ENN	Elman Neural Network	**PCC**	Pearson Correlation Coefficient
FEEMD	Fast Ensemble Empirical Mode Decomposition	**RMSE**	Root Mean Square Error
GRNN	Generalized Regression Neural Network	**RELM**	Regularized Extreme Learning Machine

GWO	Gray Wolf Optimization	**RNN**	Recurrent Neural Network
GOA	Grasshopper Optimization Algorithm	**RBMs**	Restricted Boltzmann Machines
GA	Genetic Algorithm	**SSA**	Singular Spectrum Analysis
GRU	Gated Recurrent Unit	**SSD**	Singular Spectrum Decomposition
GWEC	Global Wind Energy Council	**SARSA**	State Action Reward State Action
GBoost	Gradient Boosting	**SSAE**	Stacked Sparse Auto-Encoder
ISSD	Improved Singular Spectrum Decomposition	**SVM**	Support Vector Machine
IMF	Intrinsic Mode Function	**VMD**	Variational Mode Decomposition
ICEEMDAN	Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise	**WPD**	Wavelet Packet Decomposition
IEC	Improved Error Correction	**WNN**	Wavelet Neural Network
ICMPDQDEM	Improved Corrected Multi-Predictor Deep Q Decomposition Ensemble Model	**WOA**	Whale Optimization Algorithm

1 Introduction

The deterioration of the global environment and the exhaustion of traditional energy sources have strengthened people's desire to explore renewable energy [1]. As a green and low-carbon energy, wind energy has opened a new door for utilizing new energy [2]. Focusing on the sustainable supply characteristics of wind energy, the development, and utilization of wind energy has become an important research direction for universities and enterprises around the world [3]. The latest report of the Global Wind Energy Council (GWEC) in 2019 shows that the global wind power industry market will add 355GW of capacity between 2020 and 2024. It is estimated that by 2024, the global installed capacity of wind power will be about 71GW per year, which shows that the wind power industry has huge industry potential and room for development [4].

Nevertheless, the chaos, instability, randomness, and intermittency of wind energy bring great challenges to wind energy utilization [5]. The complexity of the temporal and spatial distribution of wind speeds in different regions exacerbates this difficulty. The efficient use of wind energy relies heavily on the accurate prediction of wind speed [6]. Inaccurate wind speed prediction will reduce the safety factor and power generation efficiency of the wind power system, resulting in huge economic losses and energy waste. To break through this dilemma, many researchers have invested a lot of time in the research of wind speed prediction models [7].

1.1 Related works

Wind speed prediction models can be broadly classified into four categories, namely physical model, statistical model, intelligent model, and hybrid model. The most common physical model is Numerical Weather Prediction (NWP) [8], which analyzes meteorological and geographic information such as temperature, density, speed, and spatial distribution to realize wind speed prediction. However, the NWP prediction method has very high requirements on data sources and processors. The slow updating speed makes it show shortcomings in short-term wind speed forecasting. In contrast, the simpler structure of the statistical model gives it a great advantage in time consumption. Common statistical models include the Auto-Regression Moving Averaging (ARMA) [9], Auto-Regressive Integrated Moving Average (ARIMA) [10], persistence model [11], classical Box-Jenkins methodology model [12], and so on. Statistical models can directly use historical wind speed data for rapid short-term prediction [13]. Unfortunately, statistical models such as ARMA are difficult to capture the strong nonlinear characteristics of real wind speed time series. This shortcoming makes its prediction accuracy difficult to meet strict engineering requirements. The development of Artificial Intelligence (AI) technology has promoted the birth of machine learning models that can handle nonlinear characteristic time series well [14]. The widely used machine learning models include Artificial Neural Network (ANN) [15], Back Propagation Neural Network (BPNN) [16], Extreme Learning Machine (ELM) [17], Long Short-Term Memory (LSTM) [18], Deep Boltzmann Machine (DBM) [19], Convolutional Neural Network (CNN) [20], etc. These models have excellent computing power. Further, to pursue models with more universality and higher prediction accuracy, the era of hybrid modeling has come [21]. A large number of surveys show that many professionals have devoted their efforts to the research of data decomposition methods, ensemble methods, and error correction methods. The literature review of these three aspects can be summarized as follows:

(a) Decomposition methods

According to research experience, it can be known that the original wind speed time series has many noises and unstable characteristics [22]. This makes it hard for predictive models to capture the changing law of wind. Directly using the original wind speed series to carry out data-driven modeling may lead to larger forecasting errors. The data preprocessing method based on decomposition solves this problem to a certain extent. *Sun et al.* use Fast Ensemble Empirical Mode Decomposition (FEEMD) to decompose the wind speed series into stationary subseries for data-driven modeling [23]. Experimental results show that the decomposition method effectively improves prediction accuracy. *Moreno et al.* utilize the Variational Mode Decomposition - Singular Spectrum Analysis (VMD-SSA) method to decompose and preprocess the wind speed time series in northeastern Brazil [24]. The combination of rough decomposition and fine decomposition makes non-stationary time series easier to be learned by ARIMA. *Yan et al.* improve Singular Spectrum Decomposition (SSD) and combine it with Grasshopper Optimization Algorithm (GOA), LSTM, and Deep Belief Network (DBN) to build a multi-step wind speed prediction model [25]. Improved Singular Spectrum Decomposition (ISSD) has stronger data preprocessing capabilities than algorithms such as VMD and Complete Ensemble Empirical Mode Decomposition (CEEMD). The decomposition method breaks the original wind speed series with high volatility into multiple stable subseries [26]. This idea has been proved to be one of the most effective data preprocessing methods [27]. However, for the decomposition algorithm, the choice of the decomposition layer has a great influence on decomposition efficiency. The previous method is to select the appropriate number of decomposition layers by combining the user's prior knowledge, but this will inevitably introduce human errors. Therefore, a data preprocessing method for adaptively determining the number of decomposition levels needs to be studied.

(b) Ensemble methods

There are two main factors that affect the effectiveness of the ensemble model, one is the benchmark predictor, and the other is the integrated optimization algorithm. The focus of different prediction models is different, which also causes the prediction performance of a single benchmark model to vary greatly on different data sets. To improve the robustness and universality of the model, ensemble methods based on meta-heuristic optimization algorithms are explored [28]. *Song et al.* use Gray Wolf Optimization (GWO) algorithm to do a weighted integration of traditional prediction models such as BPNN, Elman Neural Network (ENN), Wavelet Neural Network (WNN), and Generalized Regression Neural Network (GRNN) [29]. The combined model shows better prediction accuracy in multiple datasets. Deep learning models have stronger learning capabilities than traditional machine learning models. On this basis, *Liu et al.* use Multiple Mutations Adaptive Genetic Algorithm (MMAdapGA) to integrate ELM, Outlier-Robust Extreme Learning Machine (ORELM), and DBN [5]. The addition of DBN provides a breakthrough to improve the accuracy limit. More hidden layers enable the deep network to learn more wind speed information. It verifies the hypothesis that the benchmark predictor has a great influence on the prediction effect of the integrated model. In addition, the advantages of multi-objective optimization algorithms are reflected in the universality of prediction results. *Niu et al.* used Multi-Objective Grasshopper Optimization Algorithm (MOGOA) to ensemble five benchmark models such as BPNN, GRNN, ARIMA,

ENN, and ELM [30]. Compared with the single objective optimization algorithm, the Pareto surface formed by multiple non-dominated solutions makes the model perform well on multiple error indicators. The ensemble strategy integrates the advantages of multiple benchmark predictors and complements each other to maximize the performance [31]. Compared with common meta-heuristic optimization algorithms, reinforcement learning with strong learning ability has demonstrated strong strength in many fields [32]. If reinforcement learning can be used to integrate AI models, an excellent result may occur. Therefore, an optimized ensemble method that combines reinforcement learning and better benchmark predictor is worth investigating.

(c) Error correction methods

The randomness and intermittency of wind speed often make the machine learning model unable to fully fit its changes. At this time, data post-processing methods come in handy. The error correction is performed by calculating the error between the forecasting data and the real data to improve the numerical output accuracy of the model [33]. *Ding et al.* developed an NWP model for wind speed forecasting based on Bidirectional Gated Recurrent Unit Neural Networks (BiGRUNNs) error correction [34]. As a variant of Recurrent Neural Network (RNN), BiGRUNN's model structure effectively solves the problem of gradient disappearance. The introduction of the error correction system raises the model prediction accuracy to a higher level. *Duan et al.* proposed a decomposition-prediction correction method based on Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (ICEEMDAN) - ARIMA [35]. The wind speed error series is decomposed by ICEEMDAN into multiple Intrinsic Mode Function (IMF) error subseries, and these error series are predicted by ARIMA. Finally, these series are added to the original prediction results to obtain a final result with higher accuracy. *Liu et al.* constructed an error correction model combining Empirical Wavelet Transform (EWT) and ORELM [36]. This correction method can adaptively determine the number of decomposition levels. The addition of the decomposition method reduces the instability of the error series and makes it easier to train the correction model. The stronger robustness of ORELM can make the model converge quickly. The prediction result after correcting the error is closer to the fluctuation characteristics of wind speed [37]. Unfortunately, one-time error correction may lead to the risk of over-fitting. Too many corrections may introduce new errors, and too few corrections may not fully utilize the predictable components. This process can be controlled by obtaining an appropriate correction shrinkage rate. The introduction of correction shrinkage rate can slow down the entire fitting process and prevent over-fitting. At the same time, training multiple predictors to perform multiple iteration corrections can prevent the validation data from being input to the predictor at one time, and avoid the situation where the data used to detect predictability is consistent with the data used to train the model. In other words, if the data used to train the model is consistent with the data used to test the series predictability, the potentially predictable components in the test data may be detected as unpredictable. Then the prediction accuracy of the error correction model in the validation set will be abnormally higher than its accuracy in the test set. Using different parts of the data to train the correction predictor can balance the performance of the model on the training data and the testing data, and the predictability of the testing data can be developed to the greatest extent. Therefore, an improved adaptive multiple error correction method based on decomposition and prediction needs to be explored.

Reviewing the above decomposition methods, ensemble methods, and error correction methods, scholars have made outstanding contributions in the field of wind speed time series forecasting. **Table 1** summarizes the structure of the models, contributions, and the year of publication.

Table 1 A brief literature review of short-term wind speed prediction methods.

Author	Published year	Models	Contributions
Decomposition methods			
Sun et al. [23]	2019	FEEMD-Backtracking Search Algorithm (BSA)-Regularized Extreme Learning Machine (RELM)	The FEEMD is used to reduce the non-stationarity of the wind speed time series.
Moreno et al. [24]	2021	AM-FM-VMD-SSA-ARIMA	The Amplitude-Modulation and Frequency-Modulation (AM-FM) modem and VMD-SSA are combined into a hybrid decomposition model to suppress high-frequency noise, filter low-frequency noise, and remove non-periodic noise.
Yan et al. [25]	2020	ISSD-LSTM-GOA-DBN	The shortcomings of relying on expert experience to select SSD embedding dimensions are improved to promote decomposition performance.
Ensemble methods			
Song et al. [29]	2018	GWO-ICEEMDAN-BPNN/ENN/WNN/GRNN	The meta-heuristic GWO is used to integrate multiple benchmark models.
Liu et al. [5]	2021	Multiple Error Correction (MEC)-MMAdapGA-MODWPT-ELM/ORELM/DBN	Improved Genetic Algorithm (GA) is utilized to ensemble ELM, ORELM, and DBN.
Niu et al. [30]	2019	MOGOA-CEEMDAN-BPNN/GRNN/ARIMA/ENN/ELM	Multi-objective optimization algorithm completes the complementary advantages of BPNN, GRNN, ARIMA, ENN, and ELM models.
Error correction methods			
Ding et al. [34]	2019	NWP-BiGRUNNs	The NWP and BiGRUNNs are used to fully explore the statistical characteristics and dynamic time behavior of wind speed time series.
Duan et al. [35]	2021	ICEEMDAN-RNN-ICEEMDAN-ARIMA	The decomposition method is applied to the error correction module to improve the correction efficiency of the model.
Liu et al. [36]	2019	SSAE-BiLSTM-Multi-Objective Multi-Verse Optimization (MOMVO)-EWT-ORELM	The EWT is used in combination with ORELM to correct the wind speed prediction results under the feature extraction of Stacked Sparse Auto-Encoder (SSAE).

1.2 The innovations and contributions of this study

Section 1.1 discusses the existing research contributions and gaps in decomposition methods, ensemble methods, and error correction methods. Based on the comments of the gaps in the above research works and methods, we develop a novel reinforcement learning hybrid model called Improved Corrected Multi-Predictor Deep Q Decomposition Ensemble Model (ICMPDQDEM). The hybrid model innovatively combines reinforcement learning ensemble strategy and improved multiple error correction technique, and applies them to the field of wind speed prediction. After verifying the accuracy of the proposed model, we have obtained satisfactory prediction results. Experiments show that the proposed model can effectively capture the law of wind speed changes.

The primary innovations and contributions of this study are illustrated below:

- ***A reinforcement learning ensemble strategy is developed to determine the best model fusion weight.*** Q-learning is innovatively utilized for the ensemble weight calculation of wind speed prediction models. The reward and punishment mechanism of reinforcement learning shows excellent talent in weight optimization, which allows it to obtain more appropriate weighting coefficients. A good theoretical foundation of reinforcement learning ensures the reliability of its convergence.

- ***Three deep learning benchmark predictors are combined to learn the law of wind speed changes.*** The rich hidden layers in deep learning models Gated Recurrent Unit (GRU), Bidirectional Long Short-Term Memory (BiLSTM), and DBN give them stronger learning capabilities. This makes them more comfortable in dealing with linear and nonlinear sequences. The integrated hybrid model contains the advantages of each component, so that it can be competent for wind speed prediction tasks under more different conditions. The complementary advantages between the models solve the limitations of traditional methods that are only suitable for a single condition.

- ***A multiple error correction technique based on decomposition-prediction is proposed to improve the prediction accuracy of the model.*** The Wavelet Packet Decomposition (WPD) and ORELM are combined to predict residuals. Training multiple sub-ORELMs for adaptive iteration can prevent the validation data from being input to the model all at once, which balances the performance of the model in the validation data and the testing data. The generalization of the model is improved. The introduction of error correction shrinkage rate controls the speed of the entire correction iteration process. By slowing down the fitting process, over-fitting can be prevented effectively. Appropriate correction shrinkage rate avoids the additional human error introduced by excessive correction. The Ljung-box Q-test (LBQ-Test) is utilized to detect the predictable components of the residual series and decide whether the correction process is performed. Appropriate corrections develop the predictable components of the residual series to the greatest extent.

The organization of this article is arranged in detail below. ***Section 2*** elaborates the design of the proposed deep reinforcement learning ensemble model. ***Modules I~III*** explain the algorithm and operation mechanism of the proposed model in detail. To illustrate the feasibility and effectiveness of the proposed model in more detail, five

different comparative experiments in ***Section 3*** are performed. ***Section 3.1*** describes the wind speed time series data used in the experiments. ***Section 3.2*** gives the calculation method of the model performance evaluation index. ***Cases I~IV*** in ***Section 3.3*** discusses the role of each module in the proposed model. ***Case V*** reflects the advancement and superiority of the proposed model by comparing it against other state-of-the-art models. Specifically, ***Section 4*** analyzes the real-time property, application potentials, and application scenarios. Finally, the conclusions and future works are summarized in ***Section 5***. It is worth mentioning that ***Appendix. Supplementary materials*** list the parameter setting of the proposed model and the state-of-the-art models.

2 Design of the proposed deep reinforcement learning ensemble model

2.1 Structure of the proposed ensemble model

The proposed ICMPDQDEM model subsumes adaptive data preprocessing method, Q-learning optimization ensemble strategy, and Improved Error Correction (IEC) technique. The model introduces how the decomposition-ensemble-correction system captures the variation law of wind speed time series. **Appendix. Table A** lists the parameter settings of the proposed reinforcement learning model. **Figure 1** displays the main design method steps of the proposed model.

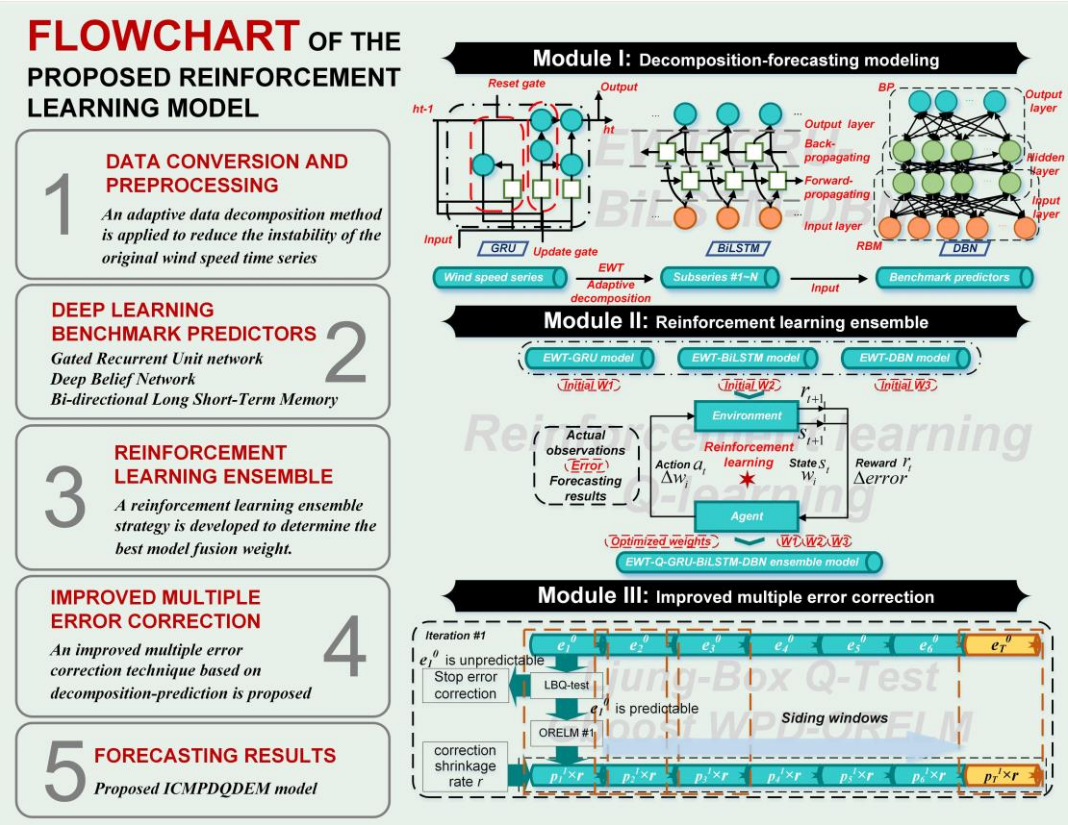


Figure 1 The proposed reinforcement learning ensemble model.

2.2 Module I: Decomposition-based benchmark predictors modeling

2.2.1 Empirical wavelet transform

According to the spectrum signal, EWT can realize the adaptive division of frequency bands. The EWT is composed of two parts of functions, including empirical scale function and empirical wavelet function. Generally, the empirical scale function is defined as a filter that filters low-pass signals. Specifically, its mathematical expression can be provided as follows [38]:

$$\psi_n(\omega)=\begin{cases}1,(1+\zeta)\omega_n\le|\omega|\le(1-\zeta)\omega_{n+1}\\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\zeta\omega_{n+1}}\left(|\omega|-(1-\zeta)\omega_{n+1}\right)\right)\right],(1-\zeta)\omega_{n+1}\le|\omega|\le(1+\zeta)\omega_{n+1}\\ \sin\left[\frac{\pi}{2}\beta\left(\frac{1}{2\zeta\omega_n}\left(|\omega|-(1-\zeta)\omega_n\right)\right)\right],(1-\zeta)\omega_n\le|\omega|\le(1+\zeta)\omega_n\\ 0,otherwise\end{cases} \tag{1}$$

The empirical wavelet function is regarded as a band-pass filter applied to each frequency band, and its mathematical expression is provided as follows:

$$\phi_n(\omega)=\begin{cases}1,|\omega|\le(1-\zeta)\omega_n\\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\zeta\omega_n}\left(|\omega|-(1-\zeta)\omega_n\right)\right)\right],(1-\zeta)\omega_n\le|\omega|\le(1+\zeta)\omega_n\\ 0,otherwise\end{cases} \tag{2}$$

Where ω_n is the boundary between adjacent frequency bands, $\omega_0=0$, $\omega_N=\pi$, and the width of the transition phase is controlled by $\zeta\in(0,1)$. To make sure that the EWT $\left\{\phi_1(t),\psi_n(t)_{n=1}^N\right\}$ is a set of orthogonal bases of $L^2(R)$, the ζ should be limited as:

$$\zeta<\min_n\left(\frac{\omega_{n+1}-\omega_n}{\omega_{n+1}+\omega_n}\right).$$

2.2.2 Gated recurrent unit

The internal units of GRU and LSTM are very similar, but the difference is that GRU merges the input gate and the forget gate into a single update gate. Finally, there are only two gate structures in the GRU, namely the update gate and the reset gate. The update gate is used to determine whether to retain the previous state information and the degree of retention. The reset gate is used to determine whether to combine the information of the current state and the previous state. The mathematical expression of the GRU neural network is provided as follows [39]:

$$\begin{aligned}z_t&=\sigma\left(W_z\left[h_{t-1},x_t\right]\right)\\ \Upsilon_t&=\sigma\left(W_\Upsilon\left[h_{t-1},x_t\right]\right)\\ h_t&=\tanh\left(W_h\left[\Upsilon_t*h_{t-1},x_t\right]\right)\\ h_t&=(1-z_t)*h_{t-1}+z_t*h_t\end{aligned} \tag{3}$$

Where W_z and W_Υ are the weight matrix of update gate and reset gate respectively, x_t is the input at the current time, h_{t-1} is the output of the hidden layer at $t-1$, z_t and Υ_t are the update gate and reset gate respectively, h_t is the output candidate value after the reset gate processing, σ is the nonlinear activation function, W_h is the weight matrix when calculating h_t, $[\ ]$ represents the connection of 2 vectors, $*$ represents the multiplication of corresponding elements between the matrices.

2.2.3 Bidirectional long short-term memory

BiLSTM enables recursive feedback of past and future hidden layer states through a two-way network. This can make the inner connection between current data and past-future time data be further explored. The hidden layer state ξ_t of BiLSTM is a combination of three parts, including the previous hidden layer output state ξ_{t-1} that propagates forward along the time axis, and the previous hidden layer output state ξ_{i-1} that propagates backward along the time axis, and the input amount x_t at the current moment. Specifically, the combined expression of each level of hidden layer state is provided as follows [40]:

$$\begin{cases} \xi_t = LSTM\left(x_t, \xi_{t-1}\right) \\ \xi_i = LSTM\left(x_t, \xi_{i-1}\right) \\ \xi_t = \chi^a{}_t \xi_t + \chi^b{}_t \xi_i + \chi^c{}_t \end{cases} \tag{4}$$

Where $LSTM$ represents the operation process of the traditional LSTM network [41], ξ_t is the forward hidden layer state, ξ_i is the backward hidden layer state, $\chi^a{}_t$ is the output weight of the hidden layer in the forward propagation unit, $\chi^b{}_t$ is the output weight of the hidden layer in the backward propagation unit, $\chi^c{}_t$ is the hidden layer bias optimization parameter at the current moment.

2.2.4 Deep belief network

Multiple Restricted Boltzmann Machines (RBMs) and a BP layer constitute the DBN. The training process of DBN consists of two parts. One is to train RBMs layer by layer in a greedy and unsupervised manner, and the other is to fine-tune the model structure parameters by training BPNN in a supervised manner. Specifically, the energy function expression of RBMs is given as follows [5]:

$$E\left(v,u \mid \theta\right) = -\sum_{i=1}^{n} a_i v_i - \sum_{j=1}^{m} \hat{b}_j u_j - \sum_{i=1}^{n} \sum_{j=1}^{m} v_i w_{ij} u_j \tag{5}$$

Where w_i represents the weights of the layer connectors, a and $\hat{b}$ are the bias of the visible and the number of hidden layer neurons, m represents the number of DBN layers, (v,u) is given randomly, initial weight $W = \{w_1, w_2, ..., w_n\}$, and $\theta = \left\{W_{ij}, a_i, \hat{b}_j\right\}$.

The original series $[X_1,...,X_{1499},X_{1500}]$ are divided into training data $[X_1,...,X_{599},X_{600}]$, validation data $[X_{601},...,X_{1199},X_{1200}]$, and test data $[X_{1201},...,X_{1499},X_{1500}]$. In the EWT phase [42], the training data $[X_1,...,X_{599},X_{600}]$ is adaptively decomposed into several subseries $[X_1^m,...,X_{599}^m,X_{600}^m], m \le N$, m represents the subseries decomposed in m-th layer, N represents the number of subseries layer.

After the decomposition is completed, the benchmark models are used for predictive modeling. The training data is input to the model and output *s-step* prediction results $\hat{X}_{GRU}^{m,s}$, $\hat{X}_{BiLSTM}^{m,s}$, $\hat{X}_{DBN}^{m,s}$ on the validation set, the S is the prediction step of the benchmark predictor. Taking GRU as an example, the predicted results of m layers of subseries are superimposed to obtain the predicted results $\hat{X}_{GRU}^{s}$. Finally, the trained benchmark predictors GRU [43], BiLSTM [44], and DBN [5] are obtained. **Figure 2** shows the mechanism of decomposition-prediction modeling.

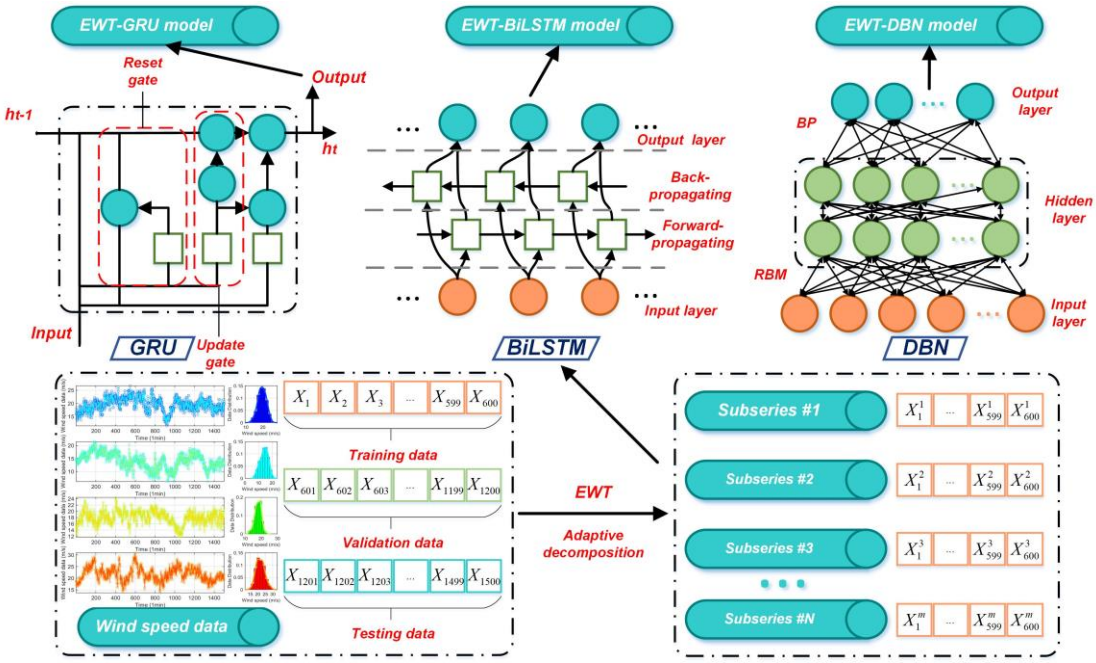


Figure 2 The mechanism of decomposition-prediction modeling.

2.3 Module II: Multi-predictor ensemble based on reinforcement learning

2.3.1 Q-learning

Reinforcement learning is a method that belongs to the field of machine learning. It emphasizes that actions change with the environment, and the purpose is to maximize the expected benefits. The algorithm is inspired by behaviorist theories in psychology [45]. Stimulated by the rewards and punishments of the external environment, the agent will gradually form the expected results of these stimuli in the iterative process, and eventually produce habitual behaviors in the direction that can obtain the most benefits. Compared with supervised learning and unsupervised learning, reinforcement learning directly abandons the need for Markov Decision Processes (MDP) [46]. It can find a balance between the unknown world and real-world

knowledge from the perspective of online planning to solve the optimal decision-making problem.

Q-learning algorithm belongs to time series differential learning, which can achieve model-free learning more efficiently than the Monte Carlo reinforcement learning algorithm [47]. This feature makes Q-learning more suitable for solving the optimization problem of wind speed time series [32]. Constructing a reasonable external environment, state space, action space, and reward function can ingeniously transform the above optimization problem into a reinforcement learning problem for solving. In this study, the Q-learning algorithm is utilized for models ensemble of GRU, BiLSTM, and DBN. The disadvantage of a single deep learning model is that its prediction performance is easily affected by data and the external environment, which leads to unstable prediction results. The ensemble of multiple deep learning models can effectively mitigate this negative impact. In the proposed model, the best weight combination is obtained by setting a reasonable reward and punishment strategy.

Step a: Initialize various parameters and states, including reward discounts γ, $0<\gamma<1$, learning rates $\alpha, 0<\alpha<1$, greedy parameters ε, Q tables, states $S=S_0$, and strategies π.

Among them, the Q agent is a value-based reinforcement learning agent. It maintains a Q table as critic to estimate the value function. State S and action A are used as the input of the critic, and a corresponding long-term return expectation is output after training. In addition, the state $S=\left[w_1, w_2, w_3\right]$ explains the composition of each specific state in the learning process, and w_1, w_2, w_3 are the weight coefficient.

$$\sum_{i=1}^{3} w_i = 1 \ , \ \exists w_i \cdot w_i \geq 0 \tag{6}$$

Step b: Perform actions $a=\pi^{\varepsilon}(s)$ according to the ε-greedy strategy π^{ε}. The mathematical calculation method of the strategy π^{ε} is provided as follows:

$$\pi^{\varepsilon}(x)=\begin{cases}\text{largest } Q \text{ value action} & \text{, probability } (1-\varepsilon) \\ \text{randomly action} & \text{, probability } \varepsilon\end{cases} \tag{7}$$

Where the ε is exploration probability, $\varepsilon \in (0,1)$.

Step c: Obtain instant rewards r_t based on the calculation method of rewards R.

According to the weight coefficients w_t, w_{t+1} corresponding to the old state S_t and the new state S_{t+1}, the loss function of the reinforcement learning stage can be solved separately, namely the prediction error $error_t$ and $error_{t+1}$. Rewards can be obtained through punishment and reward mechanisms for the agent. When $error_t < error_{t+1}$, the penalty $\Delta error - 1$ is imposed on the agent. Correspondingly, when $error_t \geq error_{t+1}$, the reward $\Delta error + 1$ is implemented for the agent. The mathematical expressions of the reward and punishment mechanism are presented as follows:

$$R=\begin{cases}\Delta error-1 & ,\ (error_t < error_{t+1})\\ \Delta error+1 & ,\ (error_t \geq error_{t+1})\end{cases}$$
$$\Delta error = error_t - error_{t+1} \tag{8}$$
$$r_t = R(S_t, a_t)$$

More specifically, the prediction error is represented by the Mean Square Error (MSE), and its expression is given as follows:

$$error = MSE = \frac{1}{N_Y}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2 \tag{9}$$

Where $\hat{Y}$ is the wind speed prediction results, Y is the real wind speed data, N_Y is the number of the real data.

Step d: Utilize the newly calculated evaluation function Q to update the Q table, and set the current state to $S_t = S_{t+1}$. The mathematical formula of the update method is provided as follows:

$$Q_{t+1}(S_t, a_t) = Q_t(S_t, a_t) + \alpha\left(r_t + \gamma \max_a Q_t(S_{t+1}, a_{t+1}) - Q_t(S_t, a_t)\right) \tag{10}$$

Where the learning rate is presented as α, $0<\alpha<1$, the discount factor is presented as γ, $0<\gamma<1$.

Step e: Repeat the above ***Steps b~d*** until the iteration termination condition is met. Finally, the EWT-Q-GRU-BiLSTM-DBN reinforcement learning ensemble model is obtained.

Figure 3 shows the mechanism of the deep reinforcement learning ensemble strategy.

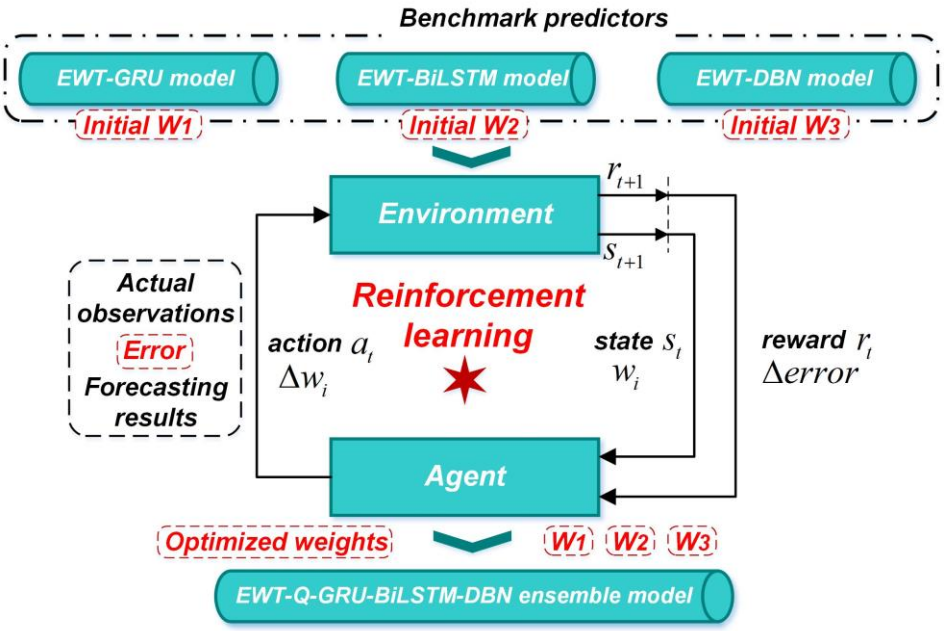


Figure 3 The mechanism of deep reinforcement learning ensemble strategy.

2.4 Module III: Improved error correction based on multiple decomposition-prediction

2.4.1 Wavelet packet decomposition

WPD is developed on the basis of wavelet decomposition and has the ability to process high and low frequency signals at the same time [48]. It is worth mentioning that WPD can adaptively match the corresponding frequency band suitable for the signal spectrum according to the characteristics and analysis requirements of the signal. Specifically, the mathematical expression of WPD is provided as follows:

$$\begin{cases} d_l^{j,2n} = \sum_k \mathbb{F}_{k-2l} d_k^{j-1,n} \\ d_l^{j,2n+1} = \sum_k g_{k-2l} d_k^{j-1,n} \end{cases} \tag{11}$$

Where $d_l^{j,2n}$ and $d_l^{j,2n+1}$ are wavelet packet coefficients, $j \in \{i, i-1, ..., 1\}$ is the scale parameter, l and k is the translation parameter, n is the frequency parameter, $\mathbb{F}_{k-2l}$ and g_{k-2l} are the low-pass filter bank and the high-pass filter bank in the wavelet packet decomposition, respectively.

2.4.2 Outlier-robust extreme learning machine

To better deal with outliers and non-Gaussian noise points in the dataset, ORELM is proposed to enhance the robustness of outliers [49]. The mathematical expression for minimizing the loss function and solving l_1-norm is provided as follows:

$$\begin{aligned} &loss = \mathbf{min} \left\| \mathbf{e}* \right\|_1 + \frac{1}{C} \left\| \Im \right\|_2^2, \ \text{s.t. } \mathbf{e}* = \mathbf{T} - \mathbf{H}\Im \\ &L_\mu(\mathbf{e}*, \Im, \lambda) = \left\| \mathbf{e}* \right\|_1 + \frac{1}{C} \left\| \Im \right\|_2^2 + \lambda^T (\mathbf{T} - \mathbf{H}\Im - \mathbf{e}*) + \frac{\mu}{2} \left\| \mathbf{T} - \mathbf{H}\Im - \mathbf{e}* \right\|_2^2 \end{aligned} \tag{12}$$

Where $\mathbf{e}*$ is the training error, C is the regularization parameter. $\mathbf{T}$ is the output of training data and $\mathbf{H}$ is the output results, the output weight is represented as $\Im$. L is the augmented Lagrangian function. λ is the Lagrange multiplier, μ is the penalty coefficient, and $\mu = \frac{2N}{\left\| \mathbf{T} \right\|_1}$.

2.4.3 Ljung-Box Q-test

As a subcategory of the Portmanteau statistical test, Ljung-Box Q-Test is often used to detect the autocorrelation of time series data [50]. Generally, if a series passes the LBQ-test, the autocorrelation of the residual sample will be reduced below the critical value determined by the significance level, and the autocorrelation will be eliminated basically. Specifically, the calculation method of LBQ-test is provided as follows [51]:

$$a_k = \sum_{t=k+1}^{n} \frac{\ell_t \ell_{t-k}}{\sum_{t=1}^{n} \ell_t^2}$$
$$Q(a) = n(n+2)\sum_{k=1}^{m} \frac{a_k^2}{n-k} \tag{13}$$

Where a_k is the autocorrelation estimate of the residual at lag *k*; *n* is the number of samples; ℓ is the mathematical modeling residual; *m* is the appropriate number of lags for the autocorrelation of the series samples.

In this study, an improved error correction technique is adopted to correct residual errors. If the residual errors in the prediction results of the first two stages can be exploited, the prediction system can better deal with the intermittent and randomness of the wind data [52]. The Multi-Predictor Deep Q Decomposition Ensemble Model (MPDQDEM) emphasizes the linear and non-linear components of the wind speed series, due to the built-in RL, GRU, BiLSTM, and DBN. The ORELM is selected as the correction model to grasp the remaining non-linear components, outliers, and non-Gaussian noise points. For the remaining predictable components in the residual error, its low-frequency interference signal and some unsubdivided high-frequency signals are difficult to be fully developed by ORELM. WPD is a time-frequency localization analysis method that can perform a multi-level division of frequency bands, and its time window and frequency window can be changed. WPD has the characteristic of adaptively selecting the frequency band to match with the signal spectrum, which makes the combination of WPD and ORELM can achieve better residual prediction results.

Step a: Pre-processing of residual series data.

Taking into account the sample balance of the divided dataset and the number of cross-validation, the validation data is divided into 6 parts to train different ORELM models, and the length of each part is 100. Different from directly training ORELM with all the validation data, using short data to train multiple ORELM models can avoid the data used for LBQ-test [50], and the data used for training the predictor are the same group. The forecasting accuracy of the ORELM in detected data will not be abnormally higher than the accuracy in testing data. The ORELM model keeps the same accuracy level both on the validation data and test data effectively ensuring that the method is effective. Therefore, the predictable state of the detected data can represent the state of the testing data. In addition, the WPD data processing method is incorporated to cooperate with the predictor to achieve a better correction effect. Considering the time-consuming problem of decomposition methods in multiple sub-ORELM modules, the number of decomposition cannot be set too large. Too few decomposition layers may lead to poor decomposition results. To achieve a balance between the two, we set the number of the decomposition to 3. In this part, subseries #1 of WPD is used as an example to show the method theory more conveniently.

Step b: Pre-defined parameter variables.

The prediction residual errors and prediction results of the *v*-th validation data part in the *c*-th correction are e_v^c and p_v^c. Correspondingly, the prediction residual errors and prediction results of the testing data in the *c*-th correction are e_T^c and p_T^c.

Step c: Test the predictability of residual series data.

Theoretically, the automatic correction function determines whether the residual series data can be predicted by ORELM. In ***Step c***, the LBQ-test method evaluates the predictability of the data according to the automatic correction function. If the prediction residual errors e_v^c are predictable after testing, then the series is used as the input of the ORELM model for training.

Step d: Correct the residual error.

The prediction residuals $\left[e_1^0, e_2^0, e_3^0, e_4^0, e_5^0, e_6^0\right]$ are obtained from the EWT-Q-GRU-BiLSTM-DBN model in the previous stage. During iteration #1, the sliding window starts to slide from the first part of the validation data. In the first sliding window of iteration #1, the e_1^0 is detected by the LBQ-test. If the detection result shows an unpredictable state, the error correction process is directly ended. If the detection result shows a predictable state, the data e_1^0 is input into the ORELM for training to get ORELM #1-1 in the first sliding window of iteration #1. As the sliding window moves, six sub-ORELMs #1-1~6 are obtained to form ORELM #1. And then, the prediction results $\left[p_1^1, p_2^1, p_3^1, p_4^1, p_5^1, p_6^1, p_T^1\right]$ can be obtained through the modeling prediction of ORELM #1. The p_T^1 is the prediction results of ORELM #1 in testing data. It is worth mentioning that slowing down the model fitting process can prevent overfitting to a certain extent. Here, we introduce an error correction shrinkage rate r to control this process, which is inspired by the learning rate in Gradient Boosting (GBoost) [53] machine. At the end of Iteration #1, a prediction result $\left\{p_v^1 \times r\right\}_{v=1,2,3,4,5,6}$ based on the error correction shrinkage rate r will be output. The prediction residuals input into iteration #2~6 can be calculated by the following expression:

$$\left\{e_v^c = e_v^{c-1} - p_v^c \times r\right\}_{v=1,2,3,4,5,6}^{c=1,2,3,4,5} \tag{14}$$

When the residual error in the prediction result is detected as an unpredictable state or when the number of iterations reaches the maximum, the process of multiple error corrections ends.

Superimposing the residual prediction results $\left\{\left[p_1^c, p_2^c, p_3^c, p_4^c, p_5^c, p_6^c, p_T^c\right] \times r\right\}_{c=1}^{6}$ and the prediction results of the EWT-Q-GRU-BiLSTM-DBN model can obtain the prediction results of the EWT-Q-GRU-BiLSTM-DBN-IEC model in the validation set and the test set. The specific calculation method is presented as follows:

$$results = \left\{\left[p_1^c, p_2^c, p_3^c, p_4^c, p_5^c, p_6^c, p_T^c\right] \times r\right\}_{c=1}^{6} + p_{EWT\text{-}Q\text{-}GRU\text{-}BiLSTM\text{-}DBN} \tag{15}$$

We traverse different correction shrinkage rates by cross validation to find the most suitable value. If the correction shrinkage rate is too small, the correction component will be too small and the correction will not be in place. If the correction shrinkage rate is too large, it will cause ORELM to correct errors excessively. The excessive correction has instead introduced new error components.

Figure 4 shows the mechanism of the improved decomposition-prediction multiple error correction method.

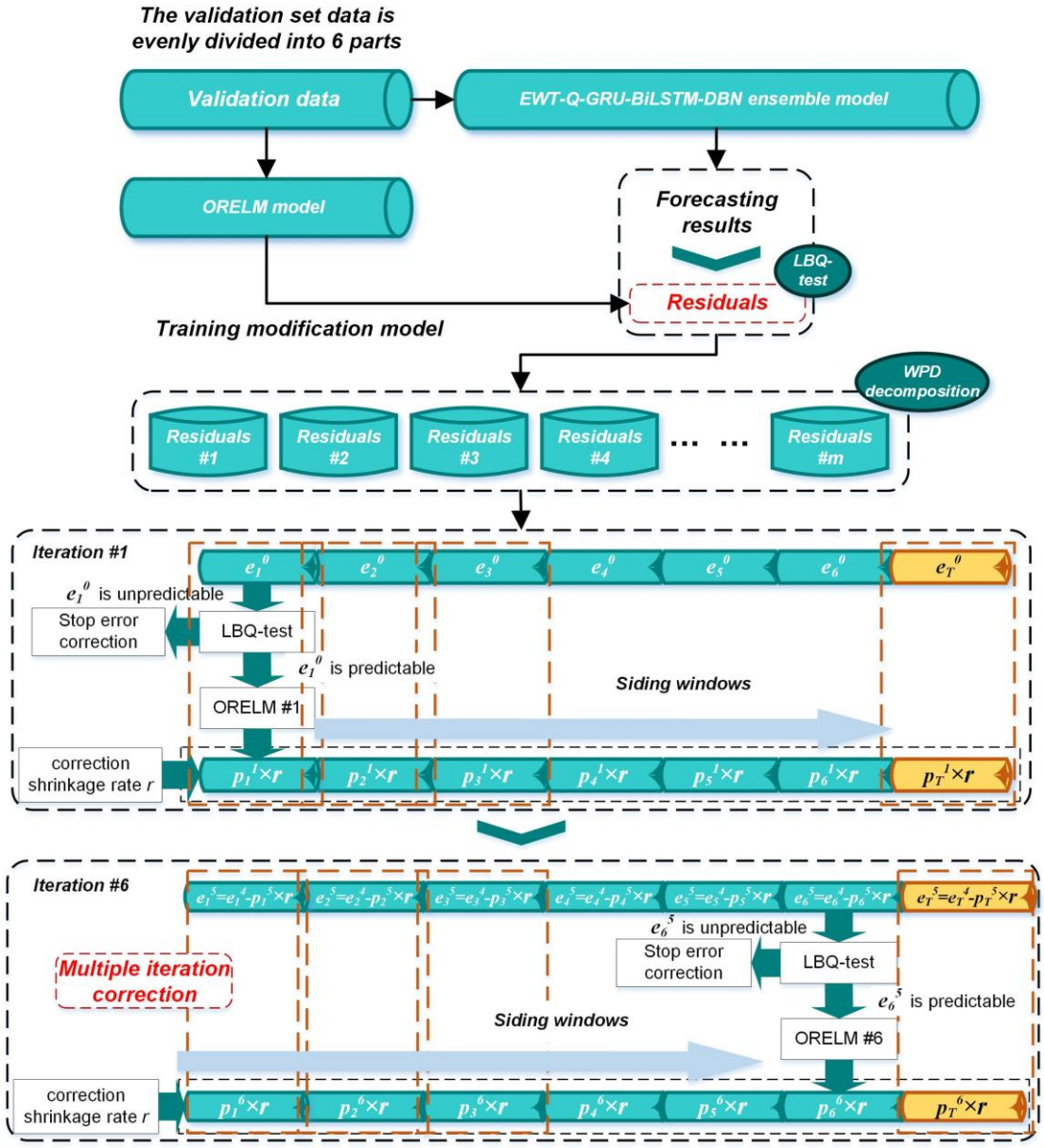


Figure 4 The mechanism of improved decomposition-prediction multiple error correction method.

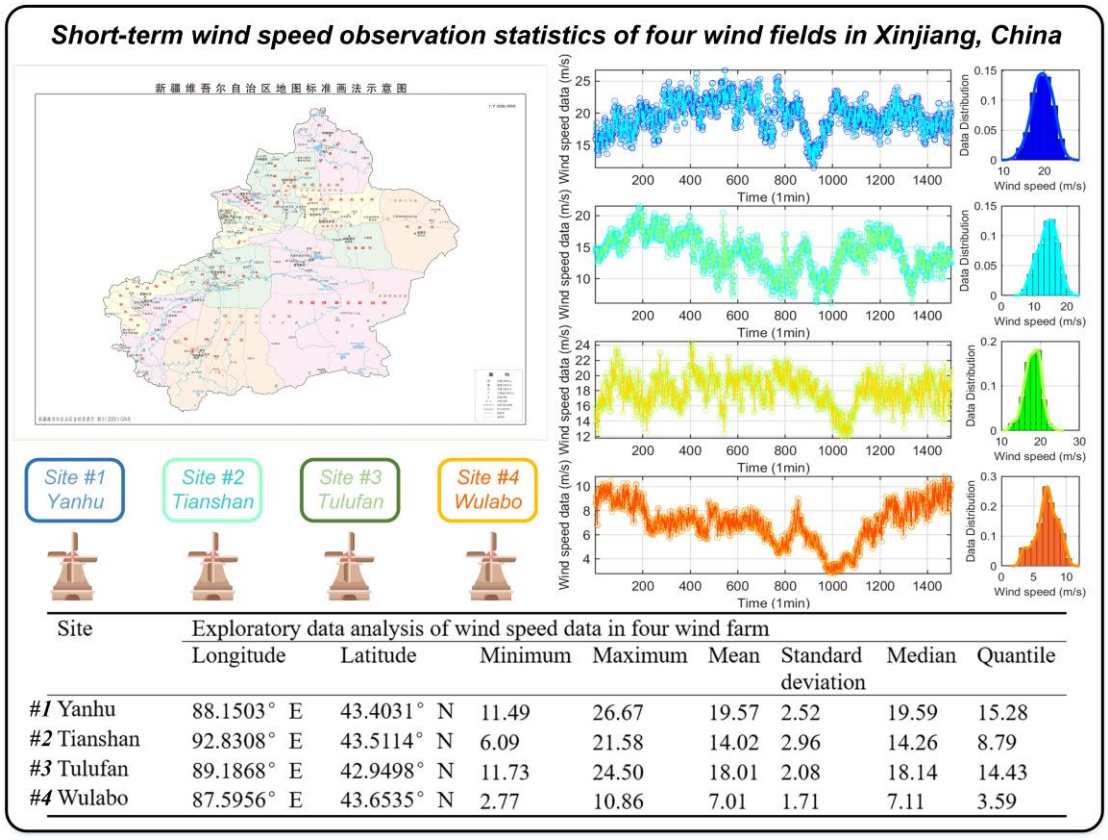
3 Results and discussion

3.1 Experimental dataset

To evaluate the prediction performance of the proposed model, four groups of original real wind speed series collected from Xinjiang, China are provided for the case studies. The data involved in the experiment is collected by high-sensitivity stationary wind measurement base station equipment. To realize the real-time collection of wind speed data, the front-end measuring sensor of this equipment adopts dual-channel redundancy mode. The sampling interval of wind data is 3 seconds to 10 minutes. The measurement range of wind data is 0m/s~70m/s, and the maximum resolution is 0.1m/s. Most of these wind speed measurement base stations are located in basins and

mountains, and the covered altitude ranges from 0m to 5000m. The ambient temperature of the equipment for wind measurement is -40℃~+55℃. The datasets used in this paper were collected in the summer of 2018. High-resolution data from nearly 30 wind measurement stations have been collected through actual measurements, with more than 1,500 wind data collected in each group. The time interval of the dataset utilized in this study is 1 min (the frequency of wind data recording).

It is worth mentioning that in order to unify the length of the data, the length of each data series is 1500. There is no 'thumb of rule' for dataset partitioning. In general, it is better to divide the training set and test set into a ratio of 0.7:0.3 or 0.8:0.2 for predictive modeling [54]. Since the proposed model requires training set to train the basic predictors, and validation set for ensemble optimization and multiple error corrections (equivalent to another training part), the length of the training data and the validation data are set to be the same. The proportion of training data, validation data, and testing data is 0.4:0.4:0.2. In general, the data is divided into three parts, including training set, validation set, and testing set. In the wind speed series, 1^{-st}~600^{-th} is training data, 601^{-st}~1200^{-th} is validation data, and 1201^{-st}~1500^{-th} is testing data. **Figure 5** shows the exploratory data analysis of wind speed data in four wind farms. The exploratory data analysis includes longitude, latitude, minimum, maximum, mean, standard deviation, median, and quantile.



Site	Exploratory data analysis of wind speed data in four wind farm							
	Longitude	Latitude	Minimum	Maximum	Mean	Standard deviation	Median	Quantile
#1 Yanhu	88.1503° E	43.4031° N	11.49	26.67	19.57	2.52	19.59	15.28
#2 Tianshan	92.8308° E	43.5114° N	6.09	21.58	14.02	2.96	14.26	8.79
#3 Tulufan	89.1868° E	42.9498° N	11.73	24.50	18.01	2.08	18.14	14.43
#4 Wulabo	87.5956° E	43.6535° N	2.77	10.86	7.01	1.71	7.11	3.59

Figure 5 The specific statistical information of the studied original datasets.

According to **Figure 5**, observations can be summarized as follows. The four wind speed data sets used in the experiment all have large wind speed amplitudes. The average wind speeds reached 19.57m/s, 14.02m/s, 18.01m/s, 7.01m/s, and the maximum wind speed is 26.67m/s, approximately equal to 96km/h, reaching the level

of wind scale 10 (whole gale). In addition, these four sets of wind speed data show greater volatility and instability. The fluctuation range of the difference between the maximum and minimum wind speed data is 8.09m/s~15.49m/s. Combining the longitude, latitude, and geographic location of the wind measurement station, it can be found that these data sets belong to the high wind environment data in high-altitude inland areas. Our research is conducted on the strong non-stationary wind in this environment, which contains abundant wind energy resources.

3.2 Evaluation indicators

Four mainstream evaluation indicators in the field of time series forecasting are used to test the forecasting errors, including Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Pearson Correlation Coefficient (PCC).

$$MAE = \left(\sum_{t=1}^{N_Y} \left| Y_t - \hat{Y}_t \right| \right) \Big/ N_Y \tag{16}$$

$$MAPE = \left(\sum_{t=1}^{N_Y} \left| (Y_t - \hat{Y}_t) / Y_t \right| \right) \Big/ N_Y \tag{17}$$

$$RMSE = \sqrt{\frac{1}{N_Y} \sum_{t=1}^{N_Y} \left(Y_t - \hat{Y}_t \right)^2} \tag{18}$$

$$PCC = \frac{\sum_{t=1}^{N_Y} \left(\left(Y_t - \overline{Y} \right) \left(\hat{Y}_t - \overline{\hat{Y}} \right) \right)}{\sqrt{\left(\sum_{t=1}^{N_Y} \left(Y_t - \overline{Y} \right)^2 \sum_{t=1}^{N_Y} \left(\hat{Y}_t - \overline{\hat{Y}} \right)^2 \right)}} \tag{19}$$

Where again $\hat{Y}$ is the prediction results, the Y is the real data, N_Y is the number of the real data, the $\overline{\hat{Y}}$ and $\overline{Y}$ is the average of the prediction results and real data, respectively.

3.3 Case study

Case I compares the performance of 8 benchmark models to screen out 3 deep learning models with the best performance in different datasets. ***Case II*** compares the impact of different ensemble strategies, including Q-learning, State Action Reward State Action (SARSA) [55], Non-dominated Sorting Genetic Algorithm II (NSGA-II), GWO, Particle Swarm Optimization and Gravitational Search Algorithm (PSOGSA), Whale Optimization Algorithm (WOA). ***Case III*** compares the impact of diverse decomposition algorithms, including EWT, Maximal Overlap Discrete Wavelet Packet Transform (MODWPT), ICEEMDAN, and Ensemble Empirical Mode Decomposition (EEMD). ***Case IV*** compares the performance of each stage of the proposed model, it is worth mentioning that the improved error correction technique is utilized in this subsection. ***Case V*** compares the performance of the proposed model and several state-of-the-art models. The abbreviations of the proposed model are provided as shown in **Table 2**.

Table 2 The abbreviations for each stage of the proposed model.

Experimental stage	Model	Abbreviation	Full name
Case II	Q-GRU-BiLSTM-DBN	MPDQEM	Multi-Predictor Deep Q Ensemble Model
Case III	EWT-Q-GRU-BiLSTM-DBN	MPDQDEM	Multi-Predictor Deep Q Decomposition Ensemble Model
Case IV	EWT-Q-GRU-BiLSTM-DBN-IEC	ICMPDQDEM **(Proposed)**	Improved Corrected Multi-Predictor Deep Q Decomposition Ensemble Model

3.3.1 Case I: Comparison with different benchmark forecasting models

Both traditional machine learning models and improved deep learning models can show good prediction accuracy in time series forecasting. To find the most suitable model from the alternative predictors, we modeled and compared Broyden Fletcher Goldfarb Shanno (BFGS), BPNN, Support Vector Machine (SVM), ENN, Nonlinear Auto Regressive (NAR), GRU, BiLSTM, and DBN to evaluate their performance. **Figure 6** shows the 1-step forecasting errors, the fitted curve, the frequency distribution graph, and the scatter graph of these benchmark models in site #1. **Table 3** presents the deterministic prediction errors of these benchmark models, where the model with the smallest error in each site is marked in bold.

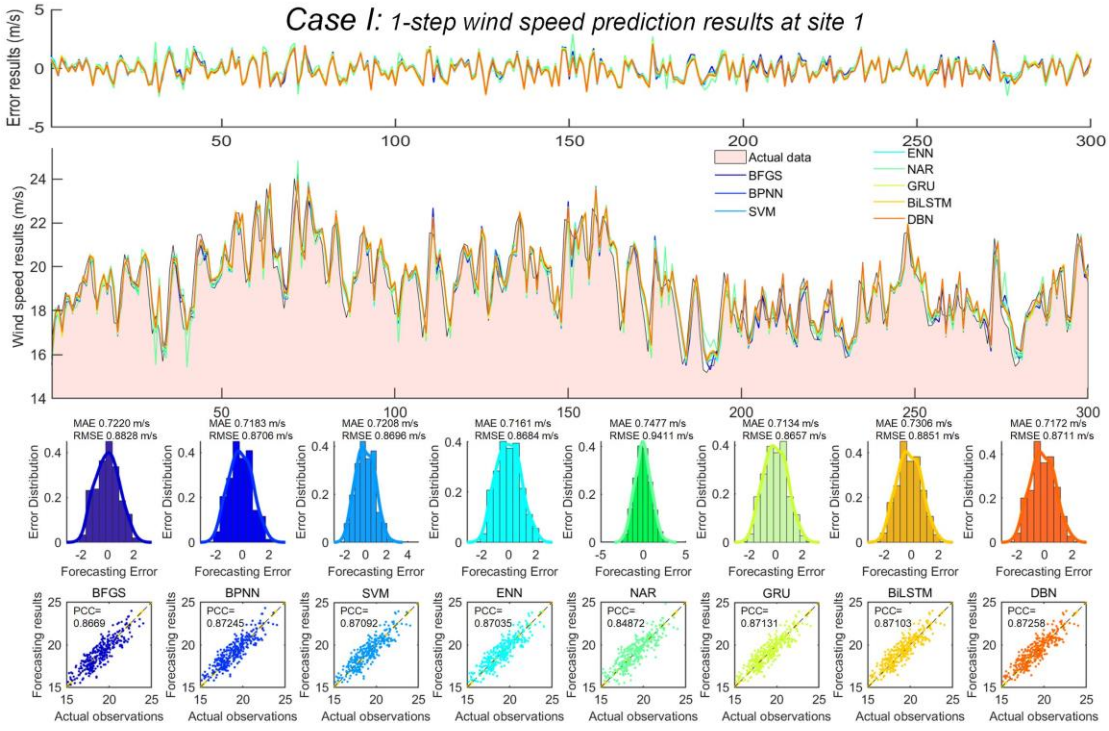


Figure 6 The 1-step prediction results of the benchmark models (Site #1).

Table 3 The deterministic forecasting results of the benchmark models.

Sites	Model	MAE (m/s)			MAPE (%)			RMSE (m/s)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
Site #1	BFGS	0.7220	1.2403	1.3654	3.7930	6.5023	7.1871	0.8828	1.5386	1.6951
	BPNN	0.7183	1.2300	1.3464	3.7928	6.4779	7.1553	0.8706	1.5071	1.6329
	SVM	0.7208	1.2214	1.3406	3.7888	6.4393	7.1128	0.8696	1.5115	1.6388
	ENN	0.7161	1.2205	1.3421	3.7730	6.4776	7.1030	0.8684	1.5064	1.6319
	NAR	0.7477	1.3160	1.4542	3.9523	6.9559	7.7233	0.9411	1.6302	1.7852
	GRU	**0.7134**	**1.2191**	**1.3349**	**3.7562**	**6.4202**	**7.0837**	**0.8657**	**1.5037**	**1.6289**
	BiLSTM	0.7306	1.2793	1.4475	3.8660	6.8300	7.8109	0.8851	1.5736	1.7608
	DBN	0.7172	1.2388	1.3745	3.7879	6.5895	7.3912	0.8711	1.5271	1.6771
Site #2	BFGS	0.5762	0.9887	1.0364	4.7039	8.0138	8.3814	0.7362	1.2556	1.3181
	BPNN	0.5608	0.9727	1.0313	4.5609	7.8936	8.3595	0.7226	1.2338	1.3215
	SVM	0.5766	0.9961	1.0666	4.6904	8.0397	8.5693	0.7448	1.2962	1.3907
	ENN	0.5654	0.9742	1.0508	4.5631	7.8571	8.5083	0.7205	1.2487	1.3411
	NAR	0.6569	1.1734	1.3042	5.3614	9.5587	10.6189	0.8567	1.6437	1.8120
	GRU	0.5616	0.9640	1.0225	4.5534	7.7943	8.2320	0.7184	1.2303	1.3176
	BiLSTM	**0.5599**	**0.9528**	**1.0102**	**4.5285**	**7.6771**	**8.1498**	**0.7145**	**1.2197**	**1.3048**
	DBN	0.5689	0.9964	1.0895	4.6253	8.2154	9.0921	0.7328	1.2894	1.4228
Site #3	BFGS	0.7682	1.3778	1.4868	4.3885	7.9130	8.6176	1.0026	1.7654	1.9376
	BPNN	0.7522	1.3252	1.4133	4.3014	7.6223	8.1921	0.9629	1.6475	1.7974
	SVM	0.7497	1.3180	1.4287	4.2816	7.5878	8.2822	0.9507	1.6468	1.8041
	ENN	0.7578	1.3325	1.4568	4.3412	7.6966	8.4671	0.9587	1.6637	1.8355
	NAR	0.8456	1.4204	1.4916	4.8214	8.2130	8.6605	1.1384	1.8229	1.9278
	GRU	0.7298	1.2799	1.3923	4.1725	7.3540	8.0383	0.9351	1.6044	1.7598
	BiLSTM	0.7654	1.3506	1.5186	4.3734	7.7807	8.7847	0.9692	1.7072	1.9187
	DBN	**0.7282**	**1.2675**	**1.3826**	**4.1489**	**7.1966**	**7.8579**	**0.9342**	**1.5983**	**1.7484**
Site #4	BFGS	0.6752	1.1723	1.2233	3.3407	5.8396	6.0996	0.8470	1.4409	1.5103
	BPNN	0.6873	1.1914	1.2504	3.4250	5.9991	6.3096	0.8464	1.4602	1.5497
	SVM	0.6615	1.1435	1.1797	3.2789	5.7082	5.8889	0.8232	1.4110	1.4720
	ENN	0.6609	1.1442	1.1821	3.2744	5.7041	5.9005	0.8264	1.4136	1.4897
	NAR	0.7921	1.3414	1.4004	3.9759	6.7246	7.0259	0.9899	1.6965	1.7749
	GRU	0.6603	1.1497	1.1928	3.2751	5.7455	5.9703	0.8256	1.4166	1.4976
	BiLSTM	**0.6587**	**1.1367**	**1.1774**	**3.2628**	**5.6633**	**5.8130**	**0.8145**	**1.3935**	**1.4547**
	DBN	0.8009	1.5899	1.9418	4.0374	8.1765	10.0129	1.0052	1.9512	2.3095

According to **Figure 6** and **Table 3**, observations can be summarized as follows:

To find the best benchmark model, we hope that the same alternative models perform best in multiple data sets, so that we can choose them without hesitation. However, due to the different characteristics of different models, the data characteristics that these predictors are good at processing are also different. So when there are many experimental comparison models, it may appear that the best performance in each group of experimental results is not always those few models [56]. Effective integration can often make the performance of the hybrid model higher than all benchmark models. The higher the upper limit of the benchmark model, the better the effect of the ensemble model [5]. Based on this theoretical basis, we adjust the criteria for selecting benchmark models to find the best performing model in each set of data sets, and then take the intersection of the models selected by multiple experiments to obtain candidate benchmark predictors. In the above four experiments, the best performance in each set of experiments is one of the three deep learning models: GRU, BiLSTM, and DBN. It can be found that if the deep learning model with the best performance in each set of experiments is used as the main predictor in the integrated model, and the remaining two deep learning models are used as auxiliary predictors, satisfactory phenomena may appear. The ensemble optimization algorithm obtains the ensemble ratio of the three benchmark predictors in the hybrid model through multiple iterations. Since each model is good at processing different data characteristics, it is possible that the auxiliary predictor will make up for some of the defects of the main predictor. This makes the integrated hybrid model have the characteristics of each sub-predictor and can adapt to more complex situations. It is worth mentioning that if all the benchmark predictors are integrated, the structure of the hybrid model will be very complicated. It seriously affects the computational efficiency of the model, and at the same time may produce the risk of overfitting.

a) The best performer on each dataset is one of the three deep learning models: GRU, BiLSTM, and DBN. Taking 1-step prediction results of site #1 as an example, the MAEs of BFGS, BPNN, SVM, ENN, NAR, GRU, BiLSTM, and DBN are 0.7220m/s, 0.7183m/s, 0.7208m/s, 0.7161m/s, 0.7477m/s, 0.7134m/s, 0.7306m/s, and 0.7172m/s, respectively. It can be seen that the model with the minimum error in site #1 is GRU. In the same comparison way, BiLSTM predictor performs better in site #2 and site #4, and DBN predictor performs better in site #3. Since the deep learning model has more hidden layers, it can more fully analyze the nonlinear characteristics of the wind speed time series. The better learning ability of deep networks makes them stand out, so the three deep learning models are selected as benchmark predictors for the next module.

Generally speaking, the stronger learning ability and adaptability of deep learning models enable them to obtain satisfactory prediction results. As for why GRU, BiLSTM, and DBN do not always perform best in these four sets of wind speed prediction experiments, some analyses are given as follows:

b) Compared with traditional prediction methods, deep learning models do not always perform better on all datasets. Taking 1-3 step prediction results of site #1 as an example, the MAEs of BPNN and BiLSTM are 0.7183m/s, 1.2300m/s, 1.3464m/s, 0.7306m/s, 1.2793m/s, and 1.4475m/s, respectively. It can be seen from the prediction error-index that the BPNN performs better than the BiLSTM at this time. From the fitting curve graph, error distribution graph, and

scatter plot, the traditional machine learning model does not lag far behind the deep learning model. The possible reason is that the deep learning model puts higher requirements on the user's prior knowledge in the hidden layer structure and parameter setting. Unreasonable initialization may lead to the risk of overfitting. Therefore, a more stable approach needs to be adopted.

c) The DBN deep learning model did not show the expected excellent performance in site #4. Taking the 1-3 step prediction results of site #4 as an example, the MAEs of DBN are 0.8009m/s, 1.5899m/s, and 1.9418m/s. The reason for the poor performance of deep learning models may be that there are too little learnable data, resulting in inadequate model training. The deep learning model does not exert all its energy, which may make its prediction performance inferior to traditional models. Therefore, a more robust method needs to be utilized in subsequent sections.

3.3.2 Case II: Comparison with models employing different ensemble strategies

In this subsection, different ensemble strategies are employed to compare the optimization performance. The optimization algorithms involved in this sub-section include Q-learning, SARSA, NSGA-II, GWO, PSOGSA, and WOA. The single objective, multi-objective optimization algorithms, and reinforcement learning algorithms all iteratively obtain the best weight combination results by setting optimization goals. Appropriate ensemble weights can make benchmark predictors complement each other and play a stronger role. **Figure 7** shows the 1-step forecasting errors, the fitted curve, the frequency distribution graph, and the scatter graph of these ensemble models in site #1. **Table 4** presents the deterministic prediction errors of these ensemble models, where the model with the smallest error in each site is marked in bold. **Table 5** lists the performance improvement percentages of benchmark models by the Q learning ensemble strategy.

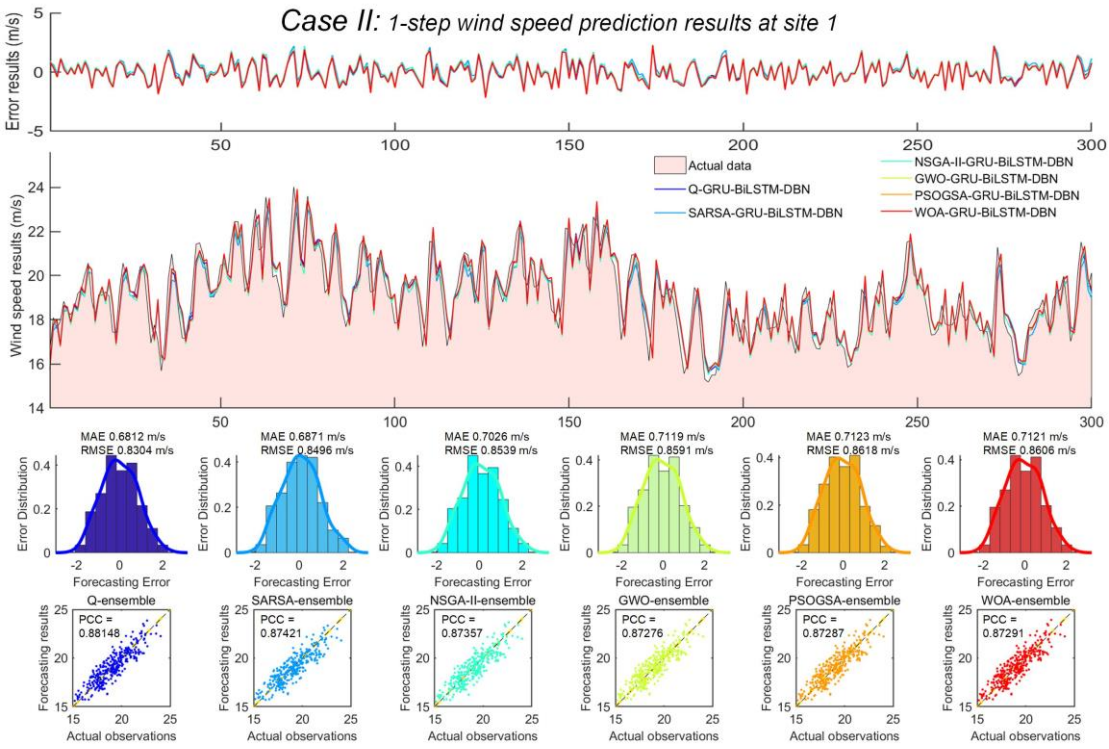


Figure 7 The 1-step prediction results of the different ensemble strategies (Site #1).

Table 4 The deterministic forecasting results of the ensemble models.

Sites	Model	MAE (m/s)			MAPE (%)			RMSE (m/s)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
Site #1	**Q-GRU-BiLSTM-DBN**	**0.6812**	**1.1008**	**1.2478**	**3.5813**	**5.7570**	**6.5429**	**0.8304**	**1.3669**	**1.5406**
	SARSA-GRU-BiLSTM-DBN	0.6871	1.1883	1.2827	3.6025	6.3071	6.8114	0.8496	1.4601	1.5741
	NSGA-II-GRU-BiLSTM-DBN	0.7026	1.2000	1.3028	3.6826	6.2414	6.8095	0.8539	1.4926	1.6040
	GWO-GRU-BiLSTM-DBN	0.7119	1.2165	1.3289	3.7522	6.4016	7.0780	0.8591	1.4966	1.6257
	PSOGSA-GRU-BiLSTM-DBN	0.7123	1.2183	1.3330	3.7536	6.4109	7.0819	0.8618	1.5004	1.6281
	WOA-GRU-BiLSTM-DBN	0.7121	1.2171	1.3302	3.7528	6.4062	7.0805	0.8606	1.4982	1.6275
Site #2	**Q-GRU-BiLSTM-DBN**	**0.5327**	**0.8356**	**0.9396**	**4.3050**	**6.6826**	**7.5260**	**0.6827**	**1.0530**	**1.1800**
	SARSA-GRU-BiLSTM-DBN	0.5379	0.8683	0.9604	4.3390	6.9740	7.7180	0.6879	1.0990	1.2145
	NSGA-II-GRU-BiLSTM-DBN	0.5488	0.9307	0.9849	4.4476	7.5553	7.9072	0.7030	1.1895	1.2637
	GWO-GRU-BiLSTM-DBN	0.5585	0.9520	1.0100	4.5154	7.6500	8.1476	0.7132	1.2148	1.3038
	PSO-GRU-BiLSTM-DBN	0.5583	0.9518	1.0060	4.5143	7.6441	8.1467	0.7130	1.2139	1.3027
	WOA-GRU-BiLSTM-DBN	0.5581	0.9513	1.0050	4.5123	7.6371	8.1453	0.7128	1.2123	1.3001
Site #3	**Q-GRU-BiLSTM-DBN**	**0.6967**	**1.0495**	**1.3275**	**3.9737**	**5.9699**	**7.5166**	**0.8883**	**1.3172**	**1.6603**
	SARSA-GRU-BiLSTM-DBN	0.7080	1.1108	1.3729	4.0372	6.3142	7.6210	0.9063	1.4056	1.7124
	NSGA-II-GRU-BiLSTM-DBN	0.7181	1.2168	1.3741	4.1095	7.1205	7.7708	0.9246	1.5768	1.7214
	GWO-GRU-BiLSTM-DBN	0.7270	1.2667	1.3708	4.1298	7.1725	7.8351	0.9318	1.5946	1.7478
	PSO-GRU-BiLSTM-DBN	0.7273	1.2671	1.3807	4.1356	7.1797	7.8455	0.9319	1.5973	1.7469
	WOA-GRU-BiLSTM-DBN	0.7266	1.2661	1.3701	4.1282	7.1607	7.8282	0.9301	1.5924	1.7441
Site #4	**Q-GRU-BiLSTM-DBN**	**0.6473**	**0.9788**	**1.0864**	**3.1896**	**4.7974**	**5.3300**	**0.7999**	**1.2117**	**1.3499**
	SARSA-GRU-BiLSTM-DBN	0.6507	1.0191	1.1108	3.2081	4.9704	5.4066	0.8073	1.2782	1.3950
	NSGA-II-GRU-BiLSTM-DBN	0.6566	1.1254	1.1610	3.2362	5.6063	5.7126	0.8101	1.3537	1.4294
	GWO-GRU-BiLSTM-DBN	0.6581	1.1357	1.1762	3.2515	5.6479	5.8075	0.8124	1.3815	1.4378
	PSO-GRU-BiLSTM-DBN	0.6582	1.1360	1.1765	3.2532	5.6582	5.8099	0.8139	1.3893	1.4447
	WOA-GRU-BiLSTM-DBN	0.6578	1.1359	1.1754	3.2510	5.6433	5.8030	0.8127	1.3845	1.4341

Table 5 The performance improvement percentages of benchmark models by the Q learning ensemble strategy.

Model	Sites	P_{MAE} (%)			P_{MAPE} (%)			P_{RMSE} (%)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
Q-GRU-BiLSTM-DBN vs. GRU	Site #1	4.51	9.71	6.53	4.66	10.33	7.63	4.08	9.09	5.42
	Site #2	5.15	13.32	8.11	5.46	14.26	8.58	4.97	14.41	10.44
	Site #3	4.53	18.00	4.65	4.76	18.82	6.49	5.01	17.90	5.66
	Site #4	1.96	14.86	8.93	2.61	16.50	10.72	3.12	14.46	9.86
Q-GRU-BiLSTM-DBN vs. BiLSTM	Site #1	6.75	13.95	13.80	7.36	15.71	16.23	6.18	13.13	12.50
	Site #2	4.85	12.30	6.99	4.93	12.95	7.65	4.45	13.66	9.56
	Site #3	8.97	22.30	12.58	9.14	23.27	14.44	8.35	22.85	13.47
	Site #4	1.73	13.89	7.73	2.24	15.29	8.31	1.79	13.04	7.20
Q-GRU-BiLSTM-DBN vs. DBN	Site #1	5.01	11.14	9.22	5.45	12.63	11.48	4.67	10.49	8.14
	Site #2	6.37	16.13	13.76	6.93	18.66	17.22	6.84	18.33	17.06
	Site #3	4.33	17.20	3.98	4.22	17.05	4.34	4.92	17.59	5.04
	Site #4	19.18	38.44	44.05	21.00	41.33	46.77	20.42	37.90	41.55

According to **Figure 7** and **Tables 4~5**, observations can be summarized as follows:

a) The ensemble strategy can make the benchmark model complement each other. Taking 1-step prediction results of site #1 as an example, the P_{MAEs} (%) improvement of Q-GRU-BiLSTM-DBN vs. GRU, vs. BiLSTM, and vs. DBN are 4.51%, 6.75%, and 5.01%, respectively. In addition, when a single predictor is unexpected, the ensemble method can compensate for the shortcomings and improve the robustness. Taking 1-3 step prediction results of site #4 as an example, the P_{MAE} (%) of Q-GRU-BiLSTM-DBN vs. DBN are 19.18%, 38.44%, and 44.05%. The ensemble strategy gives each benchmark predictor an optimal combining weight, so that these models can be integrated organically.

b) Among the weighted ensemble methods compared, Q-learning has the best ensemble effect. Taking 1-step prediction results of site #1 as an example, the MAEs of Q/SARSA/NSGA-II/GWO/PSOGSA/WOA-GRU-BiLSTM-DBN are 0.6812m/s, 0.6871m/s, 0.7026m/s, 0.7119m/s, 0.7123m/s, and 0.7121m/s, respectively. Reinforcement learning shows excellent talents in weight optimization due to the reward and punishment mechanism. Different from the meta-heuristic optimization algorithm, the Q-learning algorithm belongs to time difference learning, which has better optimization performance. Comparing with the Monte Carlo reinforcement learning algorithm, Q-learning can achieve more efficient model-free learning. Comparing the multi-objective optimization algorithm and the single objective optimization algorithm, it can be found that the Pareto solution set of the multi-objective optimization problem contains more effective information. However, the single objective weighted summation can only approximate the convex Pareto surface, so NSGA-II is superior in performance to single objective algorithms such as GWO, PSOGSA, and WOA. The single objective optimization algorithms compared are all based on similar bionic theories, so the small differences in their performance are difficult to explain theoretically.

3.3.3 Case III: Comparison with models utilizing diverse decomposition algorithms

The decomposition method can significantly improve the prediction accuracy of the model by reducing the instability of the wind speed series. To select the most suitable decomposition method, we compare EWT, MODWPT, ICEEMDAN, and EEMD based on ***Case II***. **Figure 8** shows the 1-step forecasting errors, the fitted curve, the frequency distribution graph, and the scatter graph of these decomposition-ensemble models in site #1. **Table 6** presents the deterministic prediction errors of these decomposition-ensemble models, where the model with the smallest error in each site is marked in bold. **Table 7** lists the performance improvement percentages of the ensemble models by the different decomposition methods.

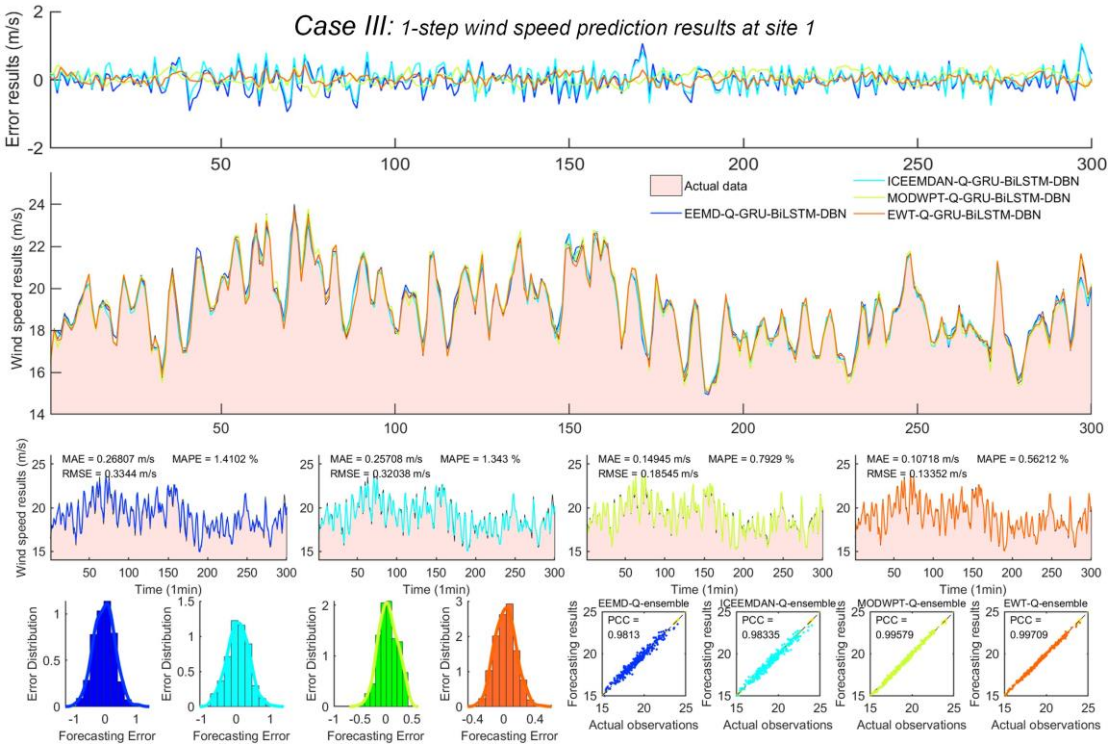


Figure 8 The 1-step prediction results of the diverse decomposition algorithms (Site #1).

Table 6 The deterministic forecasting performance of different data preprocessing methods.

Sites	Model	MAE (m/s)			MAPE (%)			RMSE (m/s)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
Site #1	**EWT-Q-GRU-BiLSTM-DBN**	**0.1072**	**0.1848**	**0.2854**	**0.5621**	**0.9727**	**1.5040**	**0.1335**	**0.2356**	**0.3614**
	MODWPT-Q-GRU-BiLSTM-DBN	0.1494	0.2233	0.3751	0.7929	1.1837	1.9949	0.1854	0.2750	0.4697
	ICEEMDAN-Q-GRU-BiLSTM-DBN	0.2571	0.3356	0.5619	1.3430	1.7617	2.9216	0.3204	0.4159	0.7162
	EEMD-Q-GRU-BiLSTM-DBN	0.2681	0.4781	0.8969	1.4102	2.4540	4.6126	0.3344	0.6024	1.0815
Site #2	**EWT-Q-GRU-BiLSTM-DBN**	**0.1430**	**0.1542**	**0.2020**	**1.1386**	**1.2583**	**1.5964**	**0.1783**	**0.1945**	**0.2517**
	MODWPT-Q-GRU-BiLSTM-DBN	0.1481	0.2344	0.3382	1.2298	1.9485	2.7900	0.1909	0.3039	0.4326
	ICEEMDAN-Q-GRU-BiLSTM-DBN	0.2443	0.3028	0.4356	2.0092	2.4497	3.5017	0.3136	0.3853	0.5680
	EEMD-Q-GRU-BiLSTM-DBN	0.6528	0.8819	1.0758	5.2756	7.1889	8.8094	0.8135	1.0910	1.3220
Site #3	**EWT-Q-GRU-BiLSTM-DBN**	**0.0931**	**0.2018**	**0.2935**	**0.5280**	**1.1645**	**1.7053**	**0.1171**	**0.2444**	**0.3530**
	MODWPT-Q-GRU-BiLSTM-DBN	0.1589	0.2125	0.3482	0.9085	1.2013	1.9856	0.2002	0.2640	0.4340
	ICEEMDAN-Q-GRU-BiLSTM-DBN	0.2993	0.3800	0.6908	1.7159	2.1871	3.9828	0.4128	0.5029	0.8642
	EEMD-Q-GRU-BiLSTM-DBN	0.3125	0.5234	0.7432	1.7990	2.9436	4.1779	0.4455	0.7131	0.9853
Site #4	**EWT-Q-GRU-BiLSTM-DBN**	**0.1109**	**0.1645**	**0.2212**	**0.5475**	**0.8049**	**1.0686**	**0.1391**	**0.2071**	**0.2873**
	MODWPT-Q-GRU-BiLSTM-DBN	0.1334	0.2176	0.2308	0.6571	1.0579	1.1425	0.1650	0.2600	0.2849
	ICEEMDAN-Q-GRU-BiLSTM-DBN	0.3340	0.4203	0.7086	1.6780	2.1088	3.5566	0.4272	0.5021	0.8873
	EEMD-Q-GRU-BiLSTM-DBN	0.4007	0.7720	0.9062	2.0219	3.9755	4.6820	0.4902	0.9699	1.0966

Table 7 The performance improvement percentages of the ensemble models by the different decomposition methods.

Model	Sites	P_{MAE} (%)			P_{MAPE} (%)			P_{RMSE} (%)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
EWT-Q-GRU-BiLSTM-DBN vs. Q-GRU-BiLSTM-DBN	Site #1	84.27	83.21	77.13	84.30	83.10	77.01	83.92	82.77	76.54
	Site #2	73.15	81.54	78.50	73.55	81.17	78.79	73.88	81.53	78.67
	Site #3	86.64	80.77	77.89	86.71	80.49	77.31	86.81	81.44	78.74
	Site #4	82.87	83.20	79.64	82.83	83.22	79.95	82.61	82.91	78.71
MODWPT-Q-GRU-BiLSTM-DBN vs. Q-GRU-BiLSTM-DBN	Site #1	78.06	79.71	69.94	77.86	79.44	69.51	77.67	79.88	69.51
	Site #2	72.19	71.95	64.01	71.43	70.84	62.93	72.03	71.14	63.34
	Site #3	77.20	79.76	73.77	77.14	79.88	73.58	77.46	79.96	73.86
	Site #4	79.39	77.77	78.76	79.40	77.95	78.57	79.37	78.55	78.89
ICEEMDAN-Q-GRU-BiLSTM-DBN vs. Q-GRU-BiLSTM-DBN	Site #1	62.26	69.51	54.97	62.50	69.40	55.35	61.42	69.58	53.51
	Site #2	54.14	63.77	53.64	53.33	63.34	53.47	54.06	63.41	51.86
	Site #3	57.04	63.79	47.96	56.82	63.36	47.01	53.53	61.82	47.95
	Site #4	48.40	57.06	34.78	47.39	56.04	33.27	46.59	58.57	34.27
EEMD-Q-GRU-BiLSTM-DBN vs. Q-GRU-BiLSTM-DBN	Site #1	60.65	56.57	28.12	60.62	57.37	29.50	59.73	55.93	29.80
	Site #2	22.54	5.54	14.50	22.54	7.58	17.05	19.17	3.60	12.03
	Site #3	55.15	50.13	44.02	54.73	50.69	44.42	49.85	45.86	40.66
	Site #4	38.09	21.13	16.59	36.61	17.13	12.16	38.71	19.96	18.77

According to **Figure 8** and **Tables 6~7**, observations can be summarized as follows:

a) The EWT, MODWPT, ICEEMDAN, and EEMD decomposition methods can effectively improve the prediction accuracy of the model. Taking 1-step prediction results of site #1 as an example, the P_{MAEs} (%) improvement of the hybrid model EWT/MODWPT/ICEEMDAN/EEMD-Q-GRU-BiLSTM-DBN vs. Q-GRU-BiLSTM-DBN are 84.27%, 78.06%, 62.26%, and 60.65%, respectively. The original wind speed series has obvious chaos and volatility, which poses a great challenge to modeling. Fortunately, the decomposition method overcomes these disadvantages by decomposing the original series into several more stable subseries. In short, the decomposition method can make the wind speed series easier to predict by reducing the instability.

b) Among the above-mentioned comparative decomposition methods, the model pre-processed by EWT decomposition has the best performance. Taking 1-step prediction results of site #1 as an example, the MAEs of the hybrid model EWT/MODWPT/ICEEMDAN/EEMD-Q-GRU-BiLSTM-DBN are 0.1072m/s, 0.1494m/s, 0.2571m/s, and 0.2681m/s, respectively. The possible reason is that EWT can adaptively determine the number of decomposition layers, which to a certain extent avoids human error. In addition, because MODWPT overcomes the problem of phase distortion, it also has a good decomposition effect. Comparing with the EEMD, the ICEEMDAN reduces modal effects by adding adaptive noise to the time series. Better convergence makes the decomposition of ICEEMDAN more efficient. The frequent modal aliasing in EMD-like algorithms exposes the shortcomings of the above two EMD-based variant algorithms, so their decomposition effect is not as good as EWT and MODWPT.

3.3.4 Case IV: Comparison with models using an improved error correction technique

The prediction results without post-processing often still have some potential predictable components. If these predictable components are developed, the prediction performance of the model can be further improved. In this subsection, an improved error correction technique is adopted to correct residual errors. **Figure 9** shows the 1-step forecasting errors, the fitted curve, the frequency distribution graph, and the scatter graph of GRU, BiLSTM, DBN, MPDQEM, MPDQDEM, and ICMPDQDEM in site #1. **Table 9** presents the deterministic prediction errors of these models, where the model with the smallest error in each site is marked in bold.

To embody the best error correction shrinkage rates in the module of error correction, the last block cross-validation is adopted [57]. After normalization, the range of error correction shrinkage rates is set to 0~1. Too small step length of candidate correction shrinkage rates will greatly increase the time-consuming process of error correction. An excessively large candidate correction shrinkage rates step length may result in missed the optimal error correction rate. To balance the two, the correction step length is set to 0.1. The main purpose of introducing error correction shrinkage rate is to slow down the correction process and reduce the risk of overfitting. Too small error correction shrinkage rate may lead to incomplete correction, and there are still predictable components remaining. A too large error correction shrinkage rate may lead to over-correction, which may introduce new errors and reduce the accuracy of the model. **Table 8** presents the best correction shrinkage rate when the MAEs are the smallest at site #1~#4.

Table 8 The optimal correction shrinkage rate under different sites.

Model	Sites			
	#1	#2	#3	#4
Proposed model (ICMPDQDEM)	0.5	0.7	0.3	0.4

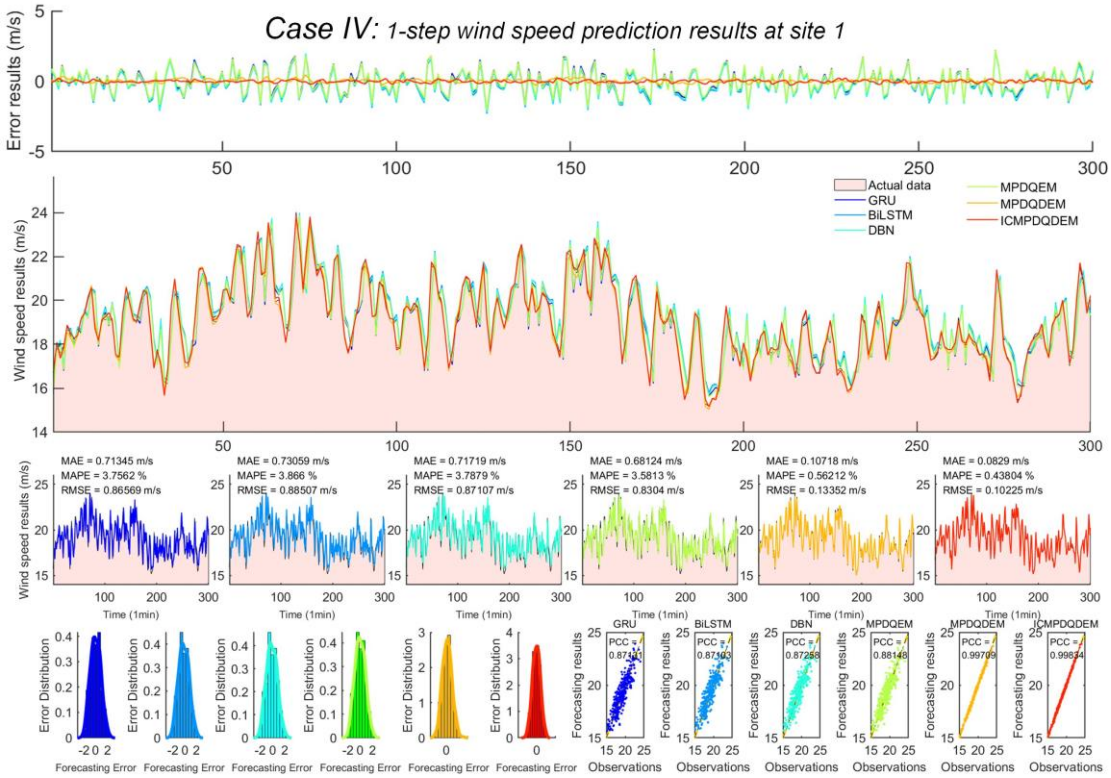


Figure 9 The 1-step prediction results for each stage of the proposed model (Site #1).

Table 9 Performance comparison for each stage of the proposed model.

Sites	Model	MAE (m/s)			MAPE (%)			RMSE (m/s)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
Site #1	GRU	0.7134	1.2191	1.3349	3.7562	6.4202	7.0837	0.8657	1.5037	1.6289
	BiLSTM	0.7306	1.2793	1.4475	3.8660	6.8300	7.8109	0.8851	1.5736	1.7608
	DBN	0.7172	1.2388	1.3745	3.7879	6.5895	7.3912	0.8711	1.5271	1.6771
	MPDQEM	0.6812	1.1008	1.2478	3.5813	5.7570	6.5429	0.8304	1.3669	1.5406
	MPDQDEM	0.1072	0.1848	0.2854	0.5621	0.9727	1.5040	0.1335	0.2356	0.3614
	ICMPDQDEM	**0.0829**	**0.1028**	**0.1425**	**0.4380**	**0.5418**	**0.7507**	**0.1023**	**0.1300**	**0.1797**
Site #2	GRU	0.5616	0.9640	1.0225	4.5534	7.7943	8.2320	0.7184	1.2303	1.3176
	BiLSTM	0.5599	0.9528	1.0102	4.5285	7.6771	8.1498	0.7145	1.2197	1.3048
	DBN	0.5689	0.9964	1.0895	4.6253	8.2154	9.0921	0.7328	1.2894	1.4228
	MPDQEM	0.5327	0.8356	0.9396	4.3050	6.6826	7.5260	0.6827	1.0530	1.1800
	MPDQDEM	0.1430	0.1542	0.2020	1.1386	1.2583	1.5964	0.1783	0.1945	0.2517
	ICMPDQDEM	**0.0661**	**0.0890**	**0.1153**	**0.5188**	**0.7013**	**0.9073**	**0.0826**	**0.1094**	**0.1412**
Site #3	GRU	0.7298	1.2799	1.3923	4.1725	7.3540	8.0383	0.9351	1.6044	1.7598
	BiLSTM	0.7654	1.3506	1.5186	4.3734	7.7807	8.7847	0.9692	1.7072	1.9187
	DBN	0.7282	1.2675	1.3826	4.1489	7.1966	7.8579	0.9342	1.5983	1.7484
	MPDQEM	0.6967	1.0495	1.3275	3.9737	5.9699	7.5166	0.8883	1.3172	1.6603
	MPDQDEM	0.0931	0.2018	0.2935	0.5280	1.1645	1.7053	0.1171	0.2444	0.3530
	ICMPDQDEM	**0.0906**	**0.1235**	**0.1643**	**0.5130**	**0.6987**	**0.9304**	**0.1138**	**0.1540**	**0.2051**
Site #4	GRU	0.6603	1.1497	1.1928	3.2751	5.7455	5.9703	0.8256	1.4166	1.4976
	BiLSTM	0.6587	1.1367	1.1774	3.2628	5.6633	5.8130	0.8145	1.3935	1.4547
	DBN	0.8009	1.5899	1.9418	4.0374	8.1765	10.0129	1.0052	1.9512	2.3095
	MPDQEM	0.6473	0.9788	1.0864	3.1896	4.7974	5.3300	0.7999	1.2117	1.3499
	MPDQDEM	0.1109	0.1645	0.2212	0.5475	0.8049	1.0686	0.1391	0.2071	0.2873
	ICMPDQDEM	**0.0803**	**0.0985**	**0.1211**	**0.3957**	**0.4840**	**0.5932**	**0.0982**	**0.1205**	**0.1534**

According to **Figure 9** and **Table 9**, it can be summarized as follows:

a) The improved error correction technique can effectively correct the residual errors. Taking 1-3 step prediction results of site #1 as an example, the MAEs of ICMPDQDEM are 0.0829m/s, 0.1028m/s, and 0.1425m/s, respectively. The possible reason is that block cross-validation selects the best correction shrinkage rate. An appropriate correction rate can ensure that the correction model can mine predictable components as much as possible while avoiding the mixing of redundant errors. In addition, the WPD can finely decompose high-frequency and low-frequency data. The ORELM has strong robustness. These two advantages make its prediction accuracy satisfactory.

b) The proposed error correction method has a more significant improvement effect when the prediction steps are higher. Taking 1-3 step prediction results of site #1 as an example, the ICMPDQDEM reduces the P_{MAEs} (%) of the MPDQDEM with 22.67%, 44.37%, and 50.07%, respectively. It can be seen that the model performance is improved to a greater extent in the 2-step and 3-step predictions. Generally, as the forecasting steps increase, the error will become larger. These residuals contain more predictable components. These predictable components are detected by the LBQ-test, and multiple iterations are performed to correct the errors. Then, the autocorrelation component of the correction results almost disappears. This shows that the predictable components in the prediction residuals are basically eliminated.

c) Each module of the proposed model has a positive effect on the improvement of prediction accuracy. The ICMPDQDEM performed best in the comparison experiment. Taking 1-step prediction results of site #1 as an example, the MAEs of GRU, BiLSTM, DBN, MPDQEM, MPDQDEM, and ICMPDQDEM are 0.7134m/s, 0.7306m/s, 0.7172m/s, 0.6812m/s, 0.1072m/s, and 0.0829m/s, respectively. The deep reinforcement learning ensemble in the MPDQEM makes the benchmark predictors complementary to each other through the optimal weight combination. It avoids the performance limitations of a single predictor and the prediction bias caused by accidental errors. By decomposing the original wind speed series into several more stable subseries, the decomposition algorithm in the MPDQDEM makes the chaotic series orderly. The ICMPDQDEM corrects the residual errors based on the previous two-stage model, so that the performance of the hybrid model reaches the best. The complementarity of each module makes the proposed model satisfactory.

3.3.5 Case V: Comparison with state-of-the-art models

To objectively verify the superiority of the proposed model, we selected four state-of-the-art models published in 2018~2021 for experimental comparison. **Appendix. Table B** shows the structure of the four state-of-the-art models.

Liu et al. [5] constructed a deep learning framework including data prediction, multi-learner ensemble, and adaptive multiple error correction processing. Compared with Liu's method, we have made further improvements to each module. In terms of predictors, more deep learning models are chosen. As for the decomposition method, the EWT which adaptively determines the number of decomposition layers is selected. For ensemble optimization, a novel reinforcement learning algorithm is used. In the correction strategy, a decomposition idea is added.

Deng et al. [58] proposed a decomposition-correction hybrid model. In error correction module, the VMD-ARIMA method based on quasi-real-time is utilized to repair the error series. Unfortunately, one-time error repair often fails to develop all predictable components. The remaining predictable components can actually give the model more room for improvement. We designed a multiple error correction model based on LBQ-test and decomposition-prediction. The appropriate error correction shrinkage rate ensures that the model does not introduce additional residuals while developing predictable components to the greatest extent.

Niu et al. [30] used MOGOA to perform a weighted ensemble of BPNN, GRNN, ARIMA, ENN, and ELM. Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) is used for data denoising preprocessing. The innovative aspect of the proposed model in our research is to use Q-learning for iterative search. Reinforcement learning breaks the stereotyped thinking of traditional learning from the perspectives of strategies, value functions, and models. The advantages in strategy search and decision making make the reinforcement learning model perform well in weight optimization.

Song et al. [29] employed the GWO to complete the optimization fusion of the benchmark models. However, the defect of the ICEEMDAN in determining the number of decomposition layers limits its performance improvement. We use an adaptive decomposition tool EWT to reduce the generation of human error, and the optimal number of decomposition layers helps the model achieve satisfactory results.

In the process of model reproduction, we continued the parameters used in the original author's article as much as possible. **Appendix. Tables C~F** list the parameter settings of the state-of-the-art models. **Figure 10** shows the 1-step forecasting errors, the fitted curve, the frequency distribution graph, and the scatter graph of these comparison models in site #1. **Figure 11** lists the error indicators of each prediction site. **Table 10** presents the deterministic prediction errors of the proposed model and above state-of-the-art models, where the model with the smallest error in each site is marked in bold.

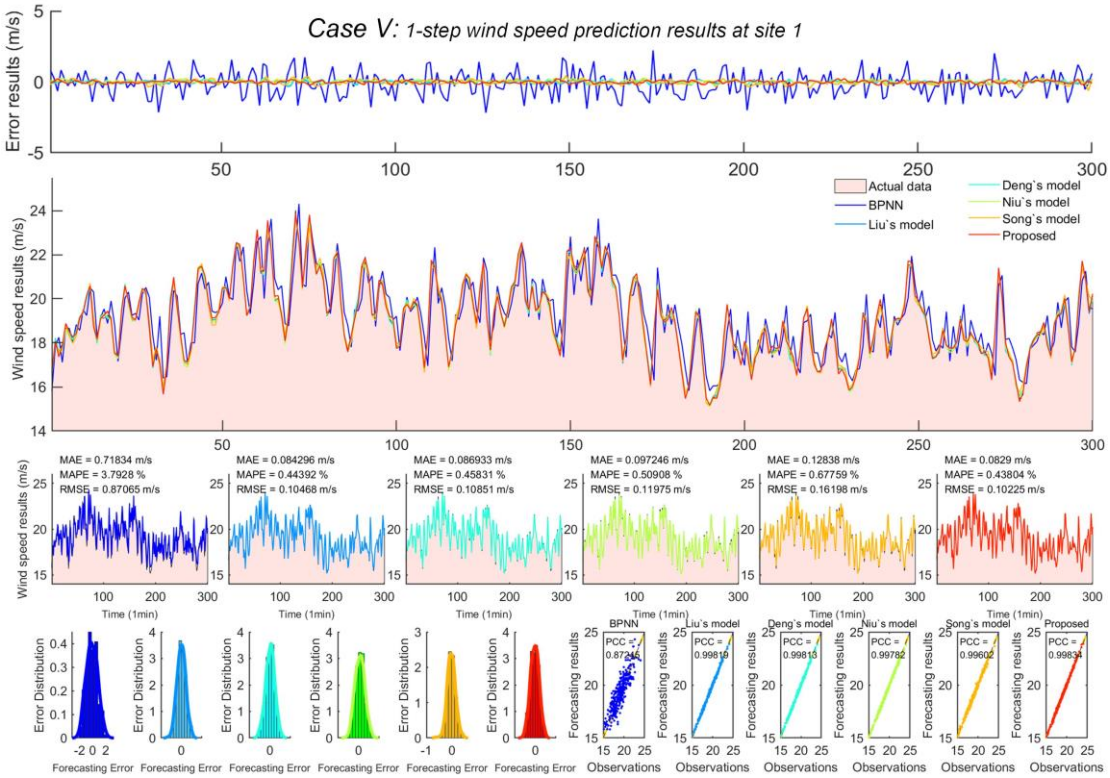


Figure 10 The 1-step prediction results of the proposed model and several state-of-the-art models (Site #1).

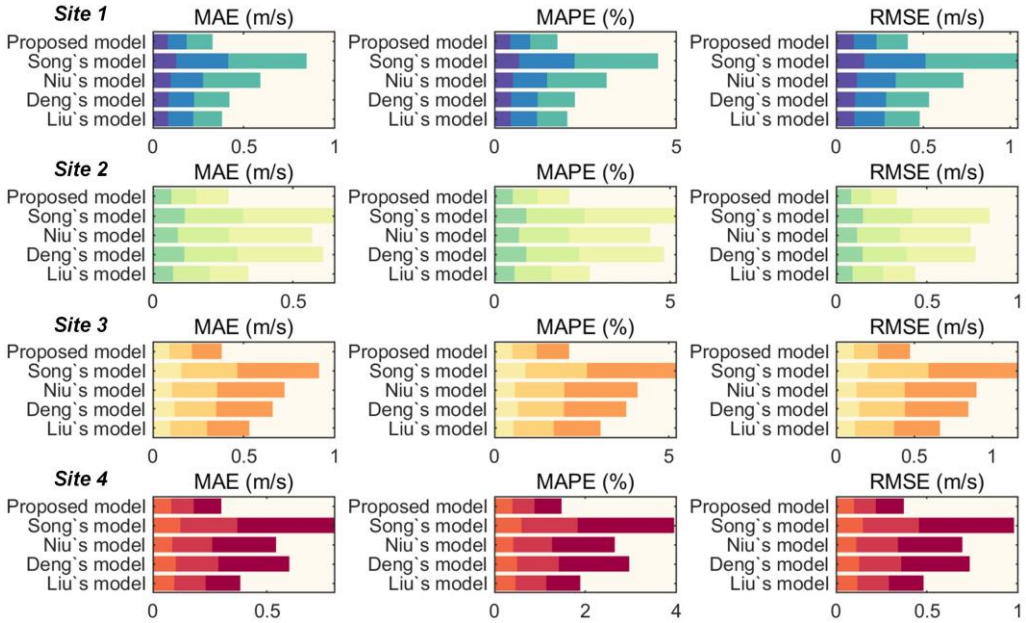


Figure 11 Stacked graph of error indicators for each comparison model.

Table 10 The deterministic forecasting performance of the proposed model and existing models.

Sites	Model	MAE (m/s)			MAPE (%)			RMSE (m/s)		
		1-step	2-step	3-step	1-step	2-step	3-step	1-step	2-step	3-step
Site #1	BPNN	0.7183	1.2300	1.3464	3.7928	6.4779	7.1553	0.8706	1.5071	1.6329
	Liu's model	0.0843	0.1380	0.1573	0.4439	0.7260	0.8322	0.1047	0.1737	0.2022
	Deng's model	0.0869	0.1395	0.1940	0.4583	0.7308	1.0199	0.1085	0.1788	0.2466
	Niu's model	0.0972	0.1788	0.3142	0.5091	0.9380	1.6341	0.1198	0.2221	0.3898
	Song's model	0.1284	0.2859	0.4303	0.6776	1.5229	2.2923	0.1620	0.3515	0.5319
	Proposed	**0.0829**	**0.1028**	**0.1425**	**0.4380**	**0.5418**	**0.7507**	**0.1023**	**0.1300**	**0.1797**
Site #2	BPNN	0.5608	0.9727	1.0313	4.5609	7.8936	8.3595	0.7226	1.2338	1.3215
	Liu's model	0.0722	0.1303	0.1383	0.5699	1.0504	1.0886	0.0905	0.1673	0.1775
	Deng's model	0.1123	0.1903	0.3049	0.9025	1.5049	2.4103	0.1456	0.2424	0.3776
	Niu's model	0.0890	0.1831	0.2965	0.6956	1.4246	2.3121	0.1152	0.2360	0.3875
	Song's model	0.1137	0.2085	0.3279	0.9037	1.6542	2.6179	0.1478	0.2700	0.4253
	Proposed	**0.0661**	**0.0890**	**0.1153**	**0.5188**	**0.7013**	**0.9073**	**0.0826**	**0.1094**	**0.1412**
Site #3	BPNN	0.7522	1.3252	1.4133	4.3014	7.6223	8.1921	0.9629	1.6475	1.7974
	Liu's model	0.0961	0.2014	0.2322	0.5431	1.1597	1.3406	0.1203	0.2508	0.2931
	Deng's model	0.1195	0.2280	0.3104	0.6749	1.3196	1.7871	0.1475	0.2936	0.4051
	Niu's model	0.1046	0.2481	0.3707	0.5883	1.4137	2.1038	0.1307	0.3087	0.4592
	Song's model	0.1561	0.3077	0.4490	0.8906	1.7623	2.5725	0.2028	0.3884	0.5734
	Proposed	**0.0906**	**0.1235**	**0.1643**	**0.5130**	**0.6987**	**0.9304**	**0.1138**	**0.1540**	**0.2051**
Site #4	BPNN	0.6873	1.1914	1.2504	3.4250	5.9991	6.3096	0.8464	1.4602	1.5497
	Liu's model	0.0937	0.1376	0.1520	0.4607	0.6775	0.7466	0.1176	0.1730	0.1898
	Deng's model	0.1008	0.1863	0.3109	0.4960	0.9178	1.5460	0.1267	0.2310	0.3762
	Niu's model	0.0849	0.1751	0.2800	0.4140	0.8504	1.3787	0.1113	0.2278	0.3544
	Song's model	0.1203	0.2499	0.4277	0.5942	1.2323	2.1217	0.1479	0.3072	0.5224
	Proposed	**0.0803**	**0.0985**	**0.1211**	**0.3957**	**0.4840**	**0.5932**	**0.0982**	**0.1205**	**0.1534**

According to **Figures 10~11** and **Table 10**, observations can be summarized as follows:

a) Comparing the experimental results of ***Case V***, the state-of-the-art models have excellent prediction performance. Taking 1-step prediction results of site #1 as an example, the MAEs of BPNN, Liu's model, Deng's model, Niu's model, Song's model and proposed model are 0.7183m/s, 0.0843m/s, 0.0869m/s, 0.0972m/s, 0.1284m/s, and 0.0829m/s, respectively. After years of hard work by researchers, the hybrid model has greatly improved in terms of data pre-processing, post-processing, and mechanism upgrades. It can be seen from the results, the classic BPNN model is obviously insufficient when compared with these state-of-the-art models. This proves that the decomposition strategies, the ensemble methods, and the error correction algorithms are all very effective in wind speed prediction.

b) With the support of deep learning benchmark predictors, EWT decomposition, Q-learning weight optimization, multiple error correction strategies based on LBQ-test, and optimal error correction shrinkage rate, the proposed ICMPDQDEM model shows satisfactory robustness and prediction accuracy. Taking 1-3 step prediction results of site #1 as an example, the MAEs, MAPEs, and RMSEs of the proposed model are 0.0829 m/s, 0.1028m/s, 0.1425m/s, 0.4380%, 0.5418%, 0.7507%, 0.1023m/s, 0.1300m/s, and 0.1797m/s, respectively. Comparing the prediction error results of classic BPNN and four state-of-the-art models, the proposed model has the best prediction performance. The improvement and combination of multiple modules help the proposed model to capture the changes of wind speed from a deeper level. The proposed hybrid model has a more reasonable structure.

4 Application analysis

4.1 Real-time property

The proposed ICMPDQDEM model is based on deep learning and reinforcement learning. These two parts consume a lot of time, so experiments are needed to prove the feasibility of the model in terms of time consumption. All experiments in this article are performed on a laptop computer with Windows 10, 2.30 GHz AMD Ryzen 7 3750H with Radeon Vega Mobile Gfx CPU and GTX 1050. **Table 11** shows the computation times of the proposed deep reinforcement learning model. The content of the table shows that the training time with four different wind speed data sets is in the range of [187.49s, 247.62s]. It can be seen that the longest training time is 247.62s (about 4.127 minutes). Compared with the 300 sample points (that is, 300 minutes) output from modeling, this time consumption is feasible. The update of the model will not affect forecasting continuity. In addition, the time taken by the trained model to predict a wind speed series is in the range of [1.76s, 2.11s]. It can be seen that the longest forecasting time is 2.11s, which is much smaller than the time interval of the model (1minute). This shows that the update of the model will not cause conflicts between the prediction process and the prediction results, and then lead to the loss of prediction sample points. Due to the complex structure of the proposed model, the computation time is longer than the single methods, but in general, the proposed deep reinforcement learning model can meet the feasibility of time consumption.

Table 11 The computation times of the proposed deep reinforcement learning model.

Time	Sites			
	Site #1	Site #2	Site #3	Site #4
Training time	213.55s	247.62s	226.88s	187.49s
Forecasting time	2.11s	1.98s	1.76s	1.85s
Computation time	215.66s	249.60s	228.64s	189.34s

It is worth mentioning that if the proposed model is embedded in MapReduce [59], Apache Spark [60], Apache Hadoop [61], and other big data platforms, its computing speed can be greatly improved. Faster modeling can better adapt to engineering needs.

4.2 Application potentials

The proposed model uses wind speed data with a time interval of 1 min for modeling. If the research results are applied to engineering practice, the requirement for wind speed time history is that the data points need to be sampled continuously at equal intervals. Different from the 10-minute average wind speed standard in China, the 1-minute wind data used in the proposed model belongs to high-resolution data. For high-resolution data, although the acquisition process may be troublesome, the application is very flexible. Commonly, high-resolution data can be converted into 10-min, 30-min, or even 60-min low-resolution data through the classic averaging method [62]. In addition, some feature extraction methods can also convert high-resolution data into low-resolution data while preserving the key information of wind speed data to the greatest extent [44]. As more and more power electronic equipment and facilities are applied to traditional wind power systems, the construction of high-resolution wind forecasting systems has become particularly important. More specifically, the application potentials of high-resolution data are summarized as follows:

- High-resolution wind speed prediction is helpful for the timely scheduling and dispatching of the wind power system. The minute-level short-term wind speed prediction provides a guarantee for the daily safe operation of large-scale wind farms, including avoiding tripping and power failure caused by load information errors in the microgrid [63], and reducing the impact of instantaneous voltage fluctuations on system equipment [64].
- High-resolution wind speed prediction provides more detailed results, which can save equipment costs for enterprises Precise wind speed prediction provides more guidance for the safety margin setting of the wind power system network. A reasonable load safety margin reservation adjustment can avoid waste of electricity, thereby reducing enterprise operating expenses.

4.3 Application scenarios

Wind conditions at different sites are rather complex and random. ***Section 3.1*** discusses the characteristics of the studied wind field, and the proposed model is suitable for wind speed prediction in high-altitude windy environments of inland areas. Different from the wind in inland areas, the sea-land breeze in coastal areas is also a typical common wind. The application of inland breeze and sea-land breeze is very important, and their respective characteristics are summarized as follows:

Xinjiang areas (inland breeze): Xinjiang is located in the northern temperate zone, deep inland and far from the sea, with scarce precipitation and arid climate. Xinjiang's topography is dominated by plains, basins, mountains, and Gobi deserts. The minimal ground shelter is very conducive to the generation of high-speed winds. Affected by mountains, high-altitude convective wind disturbances, and wind vortices can also cause near-surface winds. Different from coastal areas, the pressure fluctuations caused by the significant daily temperature difference in Xinjiang will also promote the formation of wind. Combining these geographical conditions, the wind speed in Xinjiang can usually reach the level of wind scale 10 (whole gale).

Coastal areas (sea-land breeze): Sea-land breeze occurs in offshore and coastal areas. The sea-land breeze is usually caused by the temperature difference between the sea and the land during the day-night cycle. The temperature difference causes the density and pressure difference of the atmosphere near the surface in coastal areas. The pressure gradient forces the airflow to move from high-pressure areas to low-pressure areas. Among them, the airflow blowing from the ocean to the land during the day is called the sea breeze, and the airflow blowing from the land to the ocean at night is called the land breeze. Taking the sea-land breeze in the tropics as an example, the wind speed of the sea breeze is about 7m/s, and the wind speed of the land breeze is about 1m/s~2m/s.

The differences in generation mechanism, location, and wind speed amplitude make the wind conditions of the above two types of winds vary greatly. For different wind conditions, it usually requires in-depth and specific experimental research to obtain satisfactory modeling results. A model compatible with wind speed prediction tasks in inland and coastal areas is expected to be studied. It is worth mentioning that due to data-driven characteristics, machine learning models are sensitive to the size of training samples [44]. Under normal circumstances, the prediction performance of a machine learning model will increase as the number of training samples increases. When the training sample is too small, it is difficult for the model to be trained, and the performance cannot be exerted fully.

5 Conclusions and future works

In this research, a deep reinforcement learning hybrid ensemble model is proposed for wind speed forecasting. After repeated experimental verification and case studies, the main findings are listed as follows: (a) The adopted adaptive data decomposition method can avoid the human error caused by a lack of prior knowledge. Choosing a more suitable decomposition layer can improve decomposition efficiency. (b) To determine the best model fusion weights, reinforcement learning integration strategies are developed for applications in the field of wind speed prediction. The model after the best weight combination has better robustness. (c) The WPD-ORELM multiple error correction technology combined with LBQ-test and the best correction shrinkage rate can effectively mine the predictable components. (d) The proposed model is better than the four state-of-the-art models in the comparative experiment. In general, the proposed model can contribute to the dispatch and management of wind power systems.

In future research work, more reinforcement learning mechanisms can be applied to wind speed prediction modeling, including: (a) the influence of reinforcement learning for the parameters optimization of machine learning model; (b) the effect of reinforcement learning for the features selection of wind speed time series; (c) the

performance of reinforcement learning in dynamic ensemble coefficient optimization. In addition, it is worth discussing more modeling application scenarios in future works. Finding a compatible model that is suitable for both inland large-scale wind fields and coastal sea wind fields will provide more help for wind energy development.

6 Acknowledgements

The study is fully supported by the National Natural Science Foundation of China (Grant No. 61873283), the Changsha Science & Technology Project (Grant No. KQ1707017), the innovation driven project of the Central South University (2019CX005), the Postgraduate Scientific Research Innovation Project of Hunan Province [Grant No. CX20200106], and the Fundamental Research Funds for the Central Universities of Central South University [Grant No. 2020zzts528].

7 Conflict of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Appendix: Supplementary materials

Table A The parameter settings of the proposed reinforcement learning model.

Parameter	Value
EWT	
Detection method	scalespace
Degree for the polynomial interpolation	6
Maximum number of bands	25
Sampling rate	1
Filter width	10
DBN	
MaxEpochs	100
Batch size	1
Activation function	sigm
Momentum factor	0.0
Learning rate	0.01
GRU	
MaxEpochs	100
MiniBatchSize	16
LearnRateSchedule	piecewise
L2Regularization	0.001
Learning rate	0.01
BiLSTM	
MaxEpochs	100
GradientThreshold	1
L2Regularization	0
BiLSTM-Layer	3
Learning rate	0.01
WPD	
Level of decomposition	3
Mother wavelet	db10
ORELM	
Number of the hidden neurons	Selected by cross validation
Q-learning	
Maximum iteration	100
Learning rate	0.3
Discount factor	0.95

Table B The brief analysis and summary of the state-of-the-art models.

Author	Published year	Model structure
Liu's model [5]	2021	MEC-MMAdapGA-MODWPT-ELM/ORELM/DBN
Deng's model [58]	2020	EWT-ENN-VMD-ARIMA
Niu's model [30]	2019	MOGOA-CEEMDAN-BPNN/GRNN/ARIMA/ENN/ELM
Song's model [29]	2018	GWO-ICEEMDAN- BPNN/ENN/WNN/GRNN

Table C The parameter settings of Liu's model.

Model	Algorithm	Parameter	Value
Liu's model [5]	MODWPT	Level of decomposition	3
		Mother wavelet	Daubechies wavelet
	DBN	Size of input units	10
		Size of hidden units	15
		Size of output units	1
		Momentum factor	0.0
		Learning rate	0.01
	ELM	Number of the hidden neurons	Selected by cross validation
	ORELM	Number of the hidden neurons	Selected by cross validation
	MMAdapGA	Size of population	50
		Maximum iteration	100
		Independent Discrete Precision	0.1
	ARMA	Auto-regressive degree	Selected by Bayesian information criterion
		Moving average degree	Selected by Bayesian information criterion

Table D The parameter settings of Deng's model.

Model	Algorithm	Parameter	Value
Deng's model [58]	EWT	Degree for the polynomial interpolation	6
		Detection method	scalespace
		Maximum number of bands	25
		Sampling rate	1
		Filter width	10
	ENN	Number of hidden neurons	Selected by cross validation
	VMD	Number of modes	3
		Moderate bandwidth constraint	1000
	ARIMA	Auto-Regressive/Integrated/Moving Average	Selected by Bayesian information criterion

Table E The parameter settings of Niu's model.

Model	Algorithm	Parameter	Value
Niu's model [30]	CEEMDAN	Ensemble size	500
		Noise level	0.2
	MOGOA	The maximum number of iterations	100
		Population size	50
		The number of grasshoppers	100
		The number of dims	5
		The distance between the two individuals	[1,4]
		f and l	0.5,1.5
		$\mu_{\min}$ and $\mu_{\max}$	0.00004,1
	BPNN	Number of hidden neurons	Selected by cross validation
	GRNN	Number of pattern neurons	Selected by cross validation
		Number of summation neurons	Selected by cross validation
	ARIMA	Auto-Regressive/Integrated/Moving Average	Selected by Bayesian information criterion
	ENN	Number of hidden neurons	Selected by cross validation
	ELM	Number of hidden neurons	Selected by cross validation

Table F The parameter settings of Song's model.

Model	Algorithm	Parameter	Value
Song's model [29]	ICEEMDAN	Ensemble size	500
		Noise level	0.2
	GRNN	Number of pattern neurons	Selected by cross validation
		Number of summation neurons	Selected by cross validation
	BPNN	Number of hidden neurons	Selected by cross validation
	ENN	Number of hidden neurons	Selected by cross validation
	WNN	Number of hidden neurons	Selected by cross validation
	GWO	Population size	50
		Maximum iterations	100

References

[1] Kudelin A, Kutcherov V. Wind ENERGY in Russia: The current state and development trends. Energy Strategy Reviews. 2021;34:100627.

[2] Gao Y, Ma S, Wang T, Wang T, Gong Y, Peng F, et al. Assessing the wind energy potential of China in considering its variability/intermittency. Energy Conversion and Management. 2020;226:113580.

[3] Georgilakis PS. Technical challenges associated with the integration of wind power into power systems. Renewable and Sustainable Energy Reviews. 2008;12:852-63.

[4] GWEC GWEC. Global Wind: 2019Report. 2019.

[5] Liu H, Yang R, Wang T, Zhang L. A hybrid neural network model for short-term wind speed forecasting based on decomposition, multi-learner ensemble, and adaptive multiple error corrections. Renewable Energy. 2021;165:573-94.

[6] Wang J, Li Y. Multi-step ahead wind speed prediction based on optimal feature extraction, long short term memory neural network and error correction strategy. Applied Energy. 2018;230:429-43.

[7] Lei M, Shiyan L, Chuanwen J, Hongling L, Yan Z. A review on the forecasting of wind speed and generated power. Renewable and Sustainable Energy Reviews. 2009;13:915-20.

[8] Lorenc AC. Analysis methods for numerical weather prediction. Quarterly Journal of the Royal Meteorological Society. 1986;112:1177-94.

[9] Erdem E, Shi J. ARMA based approaches for forecasting the tuple of wind speed and direction. Applied Energy. 2011;88:1405-14.

[10] Palomares-Salas J, De La Rosa J, Ramiro J, Melgar J, Aguera A, Moreno A. ARIMA vs. Neural networks for wind speed forecasting. 2009 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications: IEEE; 2009. p. 129-33.

[11] Bludszuweit H, Domínguez-Navarro JA, Llombart A. Statistical analysis of wind power forecast error. IEEE Transactions on Power Systems. 2008;23:983-91.

[12] Tang Z, De Almeida C, Fishwick PA. Time series forecasting using neural networks vs. Box-Jenkins methodology. Simulation. 1991;57:303-10.

[13] Zhang W, Qu Z, Zhang K, Mao W, Ma Y, Fan X. A combined model based on CEEMDAN and modified flower pollination algorithm for wind speed forecasting. Energy conversion and management. 2017;136:439-51.

[14] Khosravi A, Machado L, Nunes RO. Time-series prediction of wind speed using machine learning algorithms: A case study Osorio wind farm, Brazil. Applied Energy. 2018;224:550-66.

[15] Li G, Shi J. On comparing three artificial neural networks for wind speed forecasting. Applied Energy. 2010;87:2313-20.

[16] Ren C, An N, Wang J, Li L, Hu B, Shang D. Optimal parameters selection for BP neural network based on particle swarm optimization: A case study of wind speed forecasting. Knowledge-based systems. 2014;56:226-39.

[17] Liu H, Mi X, Li Y. An experimental investigation of three new hybrid wind speed forecasting models using multi-decomposing strategy and ELM algorithm. Renewable Energy. 2018;123:694-705.

[18] Wu Y-X, Wu Q-B, Zhu J-Q. Data-driven wind speed forecasting using deep feature extraction and LSTM. IET Renewable Power Generation. 2019;13:2062-9.

[19] Santhosh M, Venkaiah C, Kumar DV. Short-term wind speed forecasting approach using ensemble empirical mode decomposition and deep Boltzmann machine. Sustainable Energy, Grids and Networks. 2019;19:100242.

[20] Yan R, Liao J, Yang J, Sun W, Nong M, Li F. Multi-Hour and Multi-Site Air Quality Index Forecasting in Beijing Using CNN, LSTM, CNN-LSTM, and Spatiotemporal Clustering. Expert Systems with Applications. 2020:114513.

[21] Jiang P, Liu Z, Niu X, Zhang L. A combined forecasting system based on statistical method, artificial neural networks, and deep learning methods for short-term wind speed forecasting. Energy. 2021;217:119361.

[22] Liu H, Yang R, Duan Z, Wu H. A hybrid neural network model for marine dissolved oxygen concentrations time-series forecasting based on multi-factor analysis and a multi-model ensemble. Engineering. 2021.

[23] Sun N, Zhou J, Liu G, He Z. A hybrid wind speed forecasting model based on a decomposition method and an improved regularized extreme learning machine. Energy Procedia. 2019;158:217-22.

[24] Moreno SR, Mariani VC, Coelho LdS. Hybrid multi-stage decomposition with parametric model applied to wind speed forecasting in Brazilian Northeast. Renewable Energy. 2021;164:1508-26.

[25] Yan X, Liu Y, Xu Y, Jia M. Multistep forecasting for diurnal wind speed based on hybrid deep learning model with improved singular spectrum decomposition. Energy Conversion and Management. 2020;225:113456.

[26] Xiang L, Li J, Hu A, Zhang Y. Deterministic and probabilistic multi-step forecasting for short-term wind speed based on secondary decomposition and a deep learning method. Energy Conversion and Management. 2020;220:113098.

[27] Zhang J, Wei Y, Tan Z. An adaptive hybrid model for short term wind speed forecasting. Energy. 2020;190:115615.

[28] Peng T, Zhang C, Zhou J, Nazir MS. Negative correlation learning-based RELM ensemble model integrated with OVMD for multi-step ahead wind speed forecasting. Renewable Energy. 2020;156:804-19.

[29] Song J, Wang J, Lu H. A novel combined model based on advanced optimization algorithm for short-term wind speed forecasting. Applied Energy. 2018;215:643-58.

[30] Niu X, Wang J. A combined model based on data preprocessing strategy and multi-objective optimization algorithm for short-term wind speed forecasting. Applied Energy. 2019;241:519-39.

[31] Liu Z, Jiang P, Wang J, Zhang L. Ensemble forecasting system for short-term wind speed forecasting based on optimal sub-model selection and multi-objective version of mayfly optimization algorithm. Expert Systems with Applications. 2021;177:114974.

[32] Liu H, Yu C, Wu H, Duan Z, Yan G. A new hybrid ensemble deep reinforcement learning model for wind speed short term forecasting. Energy. 2020;202:117794.

[33] Xu W, Liu P, Cheng L, Zhou Y, Xia Q, Gong Y, et al. Multi-step wind speed prediction by combining a WRF simulation and an error correction strategy. Renewable Energy. 2021;163:772-82.

[34] Ding M, Zhou H, Xie H, Wu M, Nakanishi Y, Yokoyama R. A gated recurrent unit neural networks based wind speed error correction model for short-term wind power forecasting. Neurocomputing. 2019;365:54-61.

[35] Duan J, Zuo H, Bai Y, Duan J, Chang M, Chen B. Short-term wind speed forecasting using recurrent neural networks with error correction. Energy. 2021;217:119397.

[36] Liu H, Chen C. Multi-objective data-ensemble wind speed forecasting model with stacked sparse autoencoder and adaptive decomposition-based error correction. Applied Energy. 2019;254:113686.

[37] Zhang Y, Han J, Pan G, Xu Y, Wang F. A multi-stage predicting methodology based on data decomposition and error correction for ultra-short-term wind energy prediction. Journal of Cleaner Production. 2021;292:125981.

[38] Gilles J. Empirical wavelet transform. IEEE transactions on signal processing. 2013;61:3999-4010.

[39] Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:14123555. 2014.

[40] Peng T, Zhang C, Zhou J, Nazir MS. An integrated framework of Bi-directional long-short term memory (BiLSTM) based on sine cosine algorithm for hourly solar radiation forecasting. Energy. 2021;221:119887.

[41] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997;9:1735-80.

[42] Liu H, Wu H, Li Y. Smart wind speed forecasting using EWT decomposition, GWO evolutionary optimization, RELM learning and IEWT reconstruction. Energy Conversion and Management. 2018;161:266-83.

[43] Zhang G, Liu D. Causal convolutional gated recurrent unit network with multiple decomposition methods for short-term wind speed forecasting. Energy Conversion and Management. 2020;226:113500.

[44] Liu H, Yang R, Duan Z. Wind speed forecasting using a new multi-factor fusion and multi-resolution ensemble model with real-time decomposition and adaptive error correction. Energy Conversion and Management. 2020;217:112995.

[45] Sutton RS, Barto AG. Reinforcement learning: An introduction: MIT press; 2018.

[46] Van Otterlo M, Wiering M. Reinforcement learning and markov decision processes. Reinforcement Learning: Springer; 2012. p. 3-42.

[47] Watkins CJ, Dayan P. Q-learning. Machine learning. 1992;8:279-92.

[48] Gokhale M, Khanduja DK. Time domain signal analysis using wavelet packet decomposition approach. Int'l J of Communications, Network and System Sciences. 2010;3:321.

[49] Zhang K, Luo M. Outlier-robust extreme learning machine for regression problems. Neurocomputing. 2015;151:1519-27.

[50] Rasool G, Bouaynaya N, Iqbal K, White G. Surface myoelectric signal classification using the AR-GARCH model. Biomedical Signal Processing and Control. 2014;13:327-36.

[51] Hassani H, Yeganegi MR. Selecting optimal lag order in Ljung–Box test. Physica A: Statistical Mechanics and its Applications. 2020;541:123700.

[52] Ma Z, Chen H, Wang J, Yang X, Yan R, Jia J, et al. Application of hybrid model based on double decomposition, error correction and deep learning in short-term wind speed prediction. Energy Conversion and Management. 2020;205:112345.

[53] Friedman JH. Greedy function approximation: a gradient boosting machine. Annals of statistics. 2001:1189-232.

[54] Emeksiz C, Tan M. Multi-step wind speed forecasting and Hurst analysis using novel hybrid secondary decomposition approach. Energy. 2022;238:121764.

[55] Zhao D, Wang H, Shao K, Zhu Y. Deep reinforcement learning with experience replay based on SARSA. 2016 IEEE Symposium Series on Computational Intelligence (SSCI): IEEE; 2016. p. 1-6.

[56] Zhang L, Wang J, Niu X, Liu Z. Ensemble wind speed forecasting with multi-objective Archimedes optimization algorithm and sub-model selection. Applied Energy. 2021;301:117449.

[57] Bergmeir C, Benítez JM. On the use of cross-validation for time series predictor evaluation. Information Sciences. 2012;191:192-213.

[58] Deng Y, Wang B, Lu Z. A hybrid model based on data preprocessing strategy and error correction system for wind speed forecasting. Energy Conversion and Management. 2020;212:112779.

[59] Bendre M, Manthalkar R. Time series decomposition and predictive analytics using MapReduce framework. Expert Systems with Applications. 2019;116:108-20.

[60] Harnie D, Saey M, Vapirev AE, Wegner JK, Gedich A, Steijaert M, et al. Scaling machine learning for target prediction in drug discovery using apache spark. Future Generation Computer Systems. 2017;67:409-17.

[61] Nandimath J, Banerjee E, Patil A, Kakade P, Vaidya S, Chaturvedi D. Big data analysis using Apache Hadoop. 2013 IEEE 14th International Conference on Information Reuse & Integration (IRI): IEEE; 2013. p. 700-3.

[62] Demolli H, Dokuz AS, Ecemis A, Gokcek M. Wind power forecasting based on daily wind speed data using machine learning algorithms. Energy Conversion and Management. 2019;198:111823.

[63] Hu R, Hu W, Gökmen N, Li P, Huang Q, Chen Z. High resolution wind speed forecasting based on wavelet decomposed phase space reconstruction and self-organizing map. Renewable Energy. 2019;140:17-31.

[64] Wang Y, Wang H, Srinivasan D, Hu Q. Robust functional regression for wind speed forecasting based on Sparse Bayesian learning. Renewable Energy. 2019;132:43-60.

AUTHOR DECLARATION TEMPLATE

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He/she is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the Corresponding Author and which has been configured to accept email from csuliuhui@csu.edu.cn

All of authors,
17.06.2021

FLOWCHART OF THE PROPOSED REINFORCEMENT LEARNING MODEL

1 DATA CONVERSION AND PREPROCESSING

An adaptive data decomposition method is applied to reduce the instability of the original wind speed time series

2 DEEP LEARNING BENCHMARK PREDICTORS

Gated Recurrent Unit network
Deep Belief Network
Bi-directional Long Short-Term Memory

3 REINFORCEMENT LEARNING ENSEMBLE

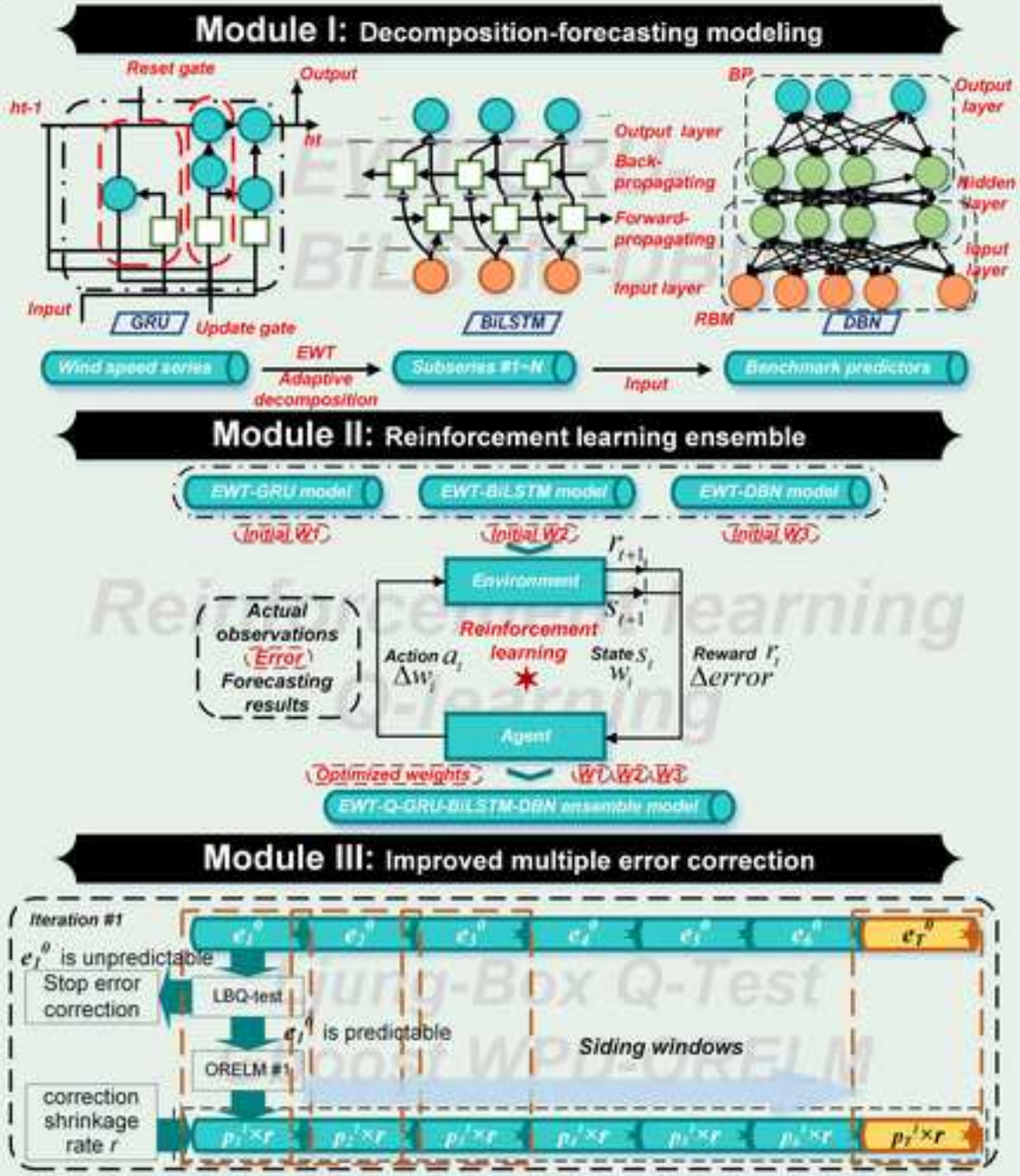
A reinforcement learning ensemble strategy is developed to determine the best model fusion weight.

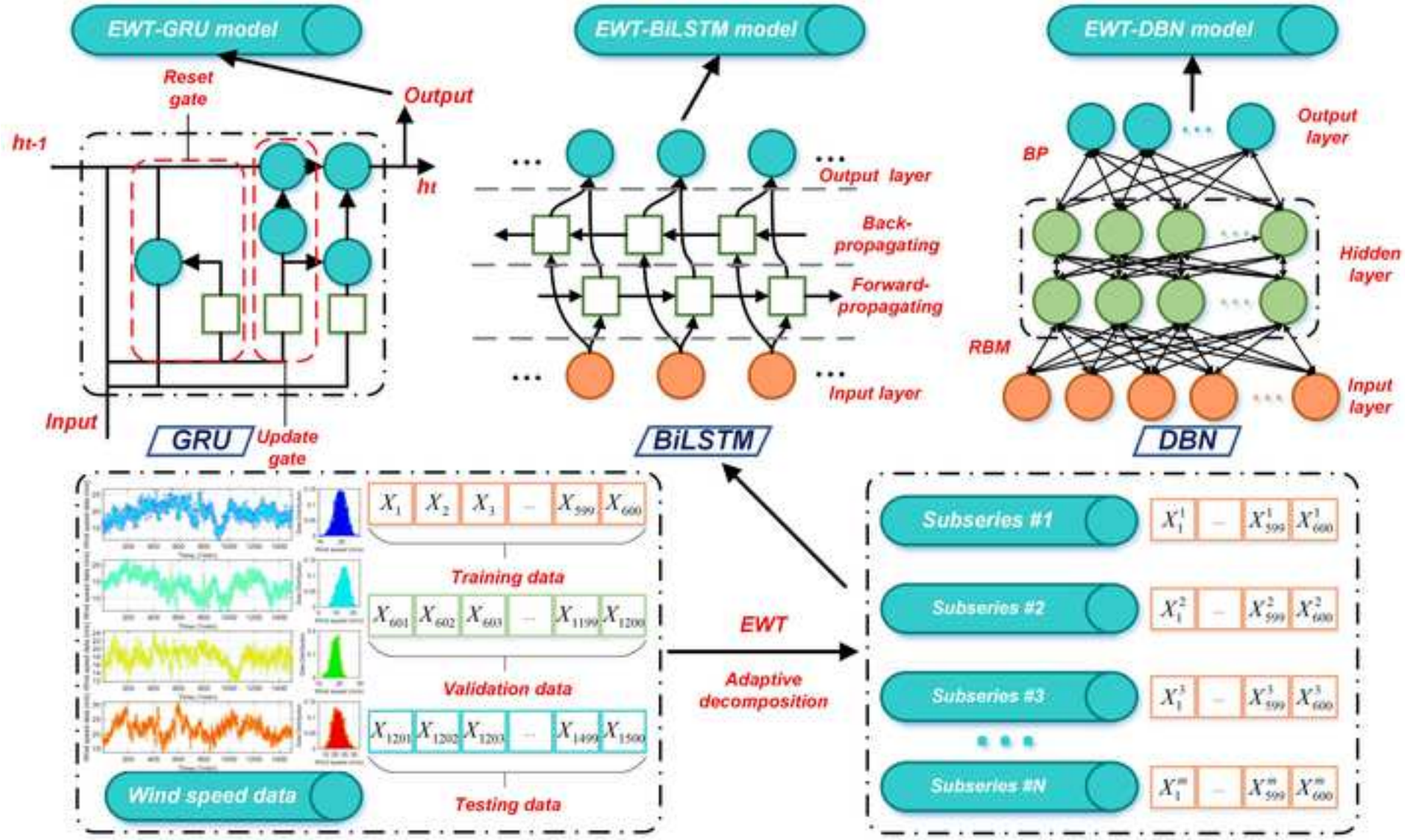
4 IMPROVED MULTIPLE ERROR CORRECTION

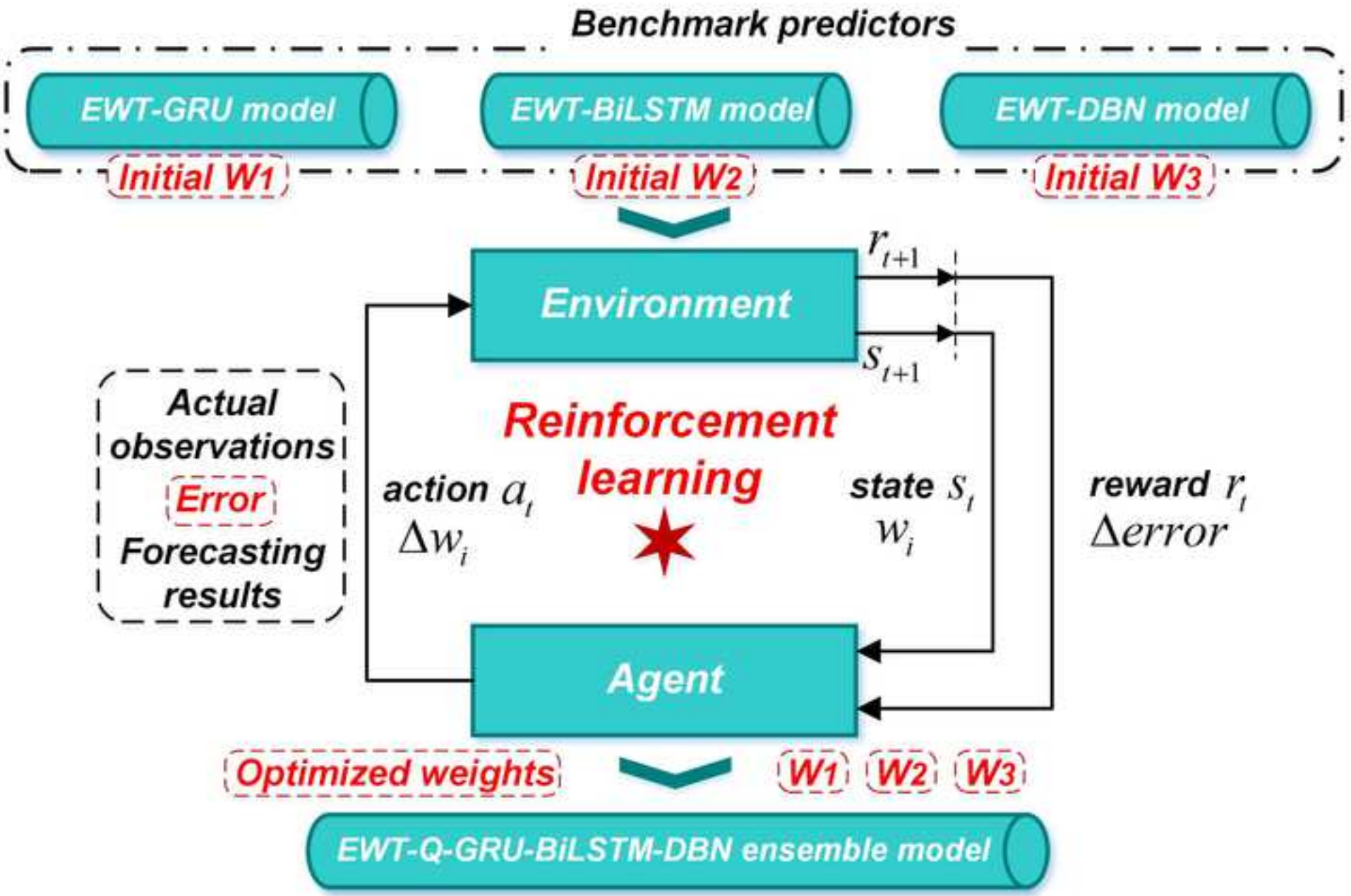
An improved multiple error correction technique based on decomposition-prediction is proposed

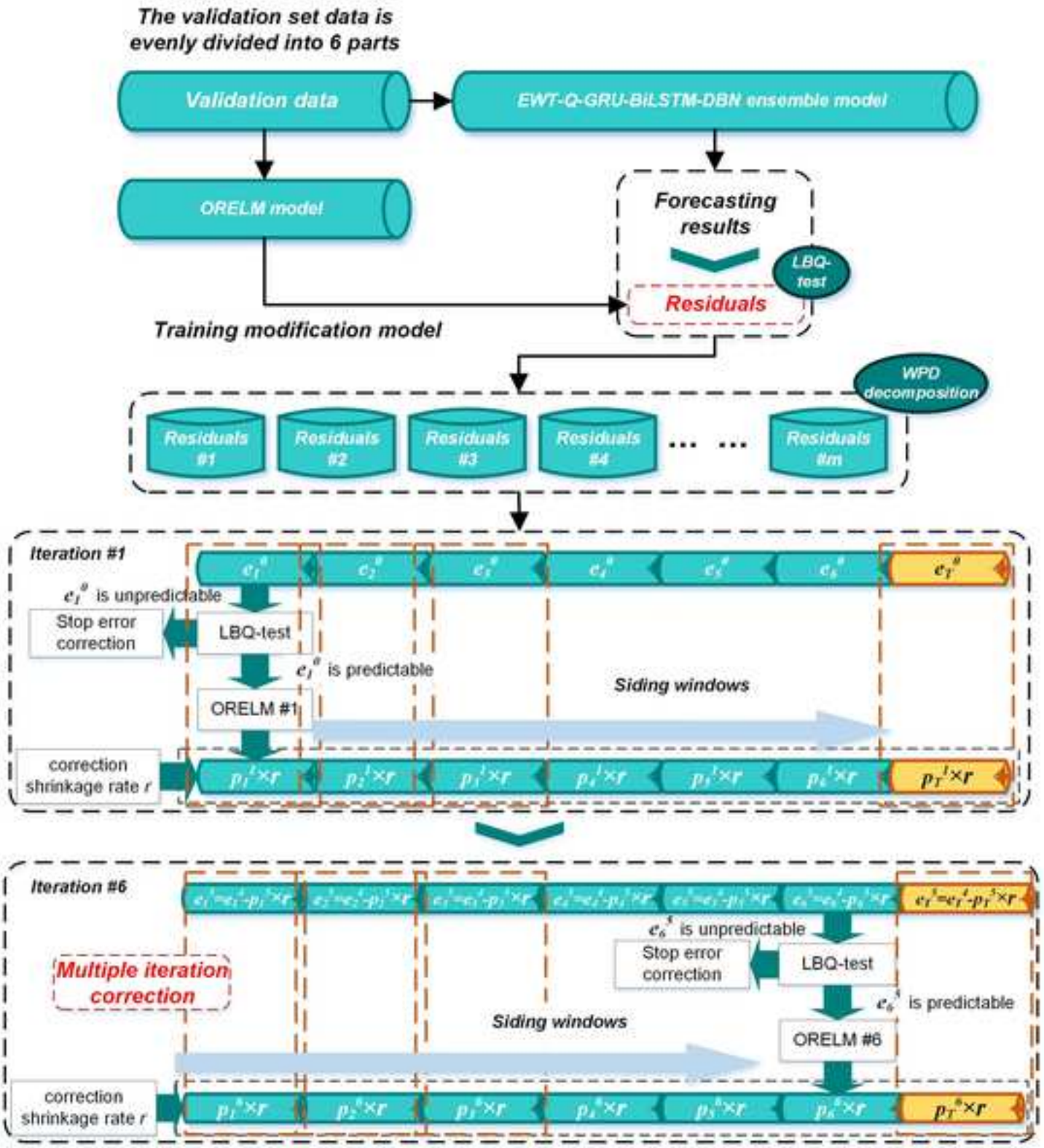
5 FORECASTING RESULTS

Proposed ICMPDQDEM model

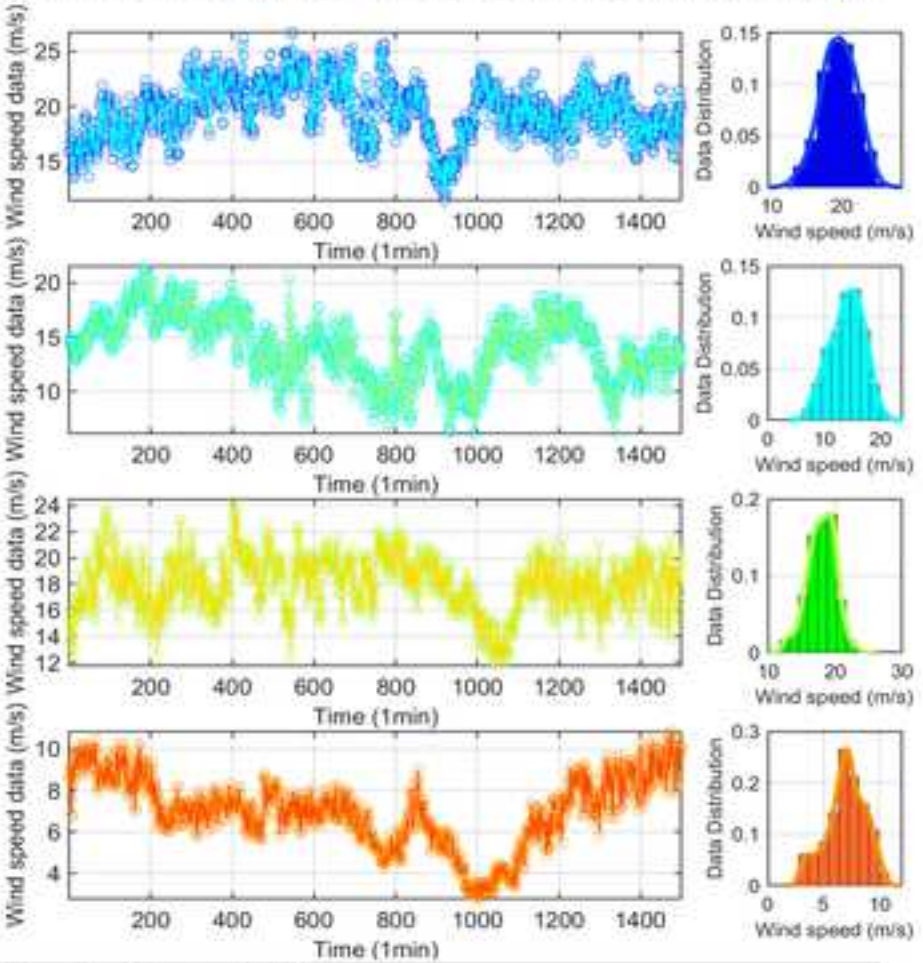








Short-term wind speed observation statistics of four wind fields in Xinjiang, China



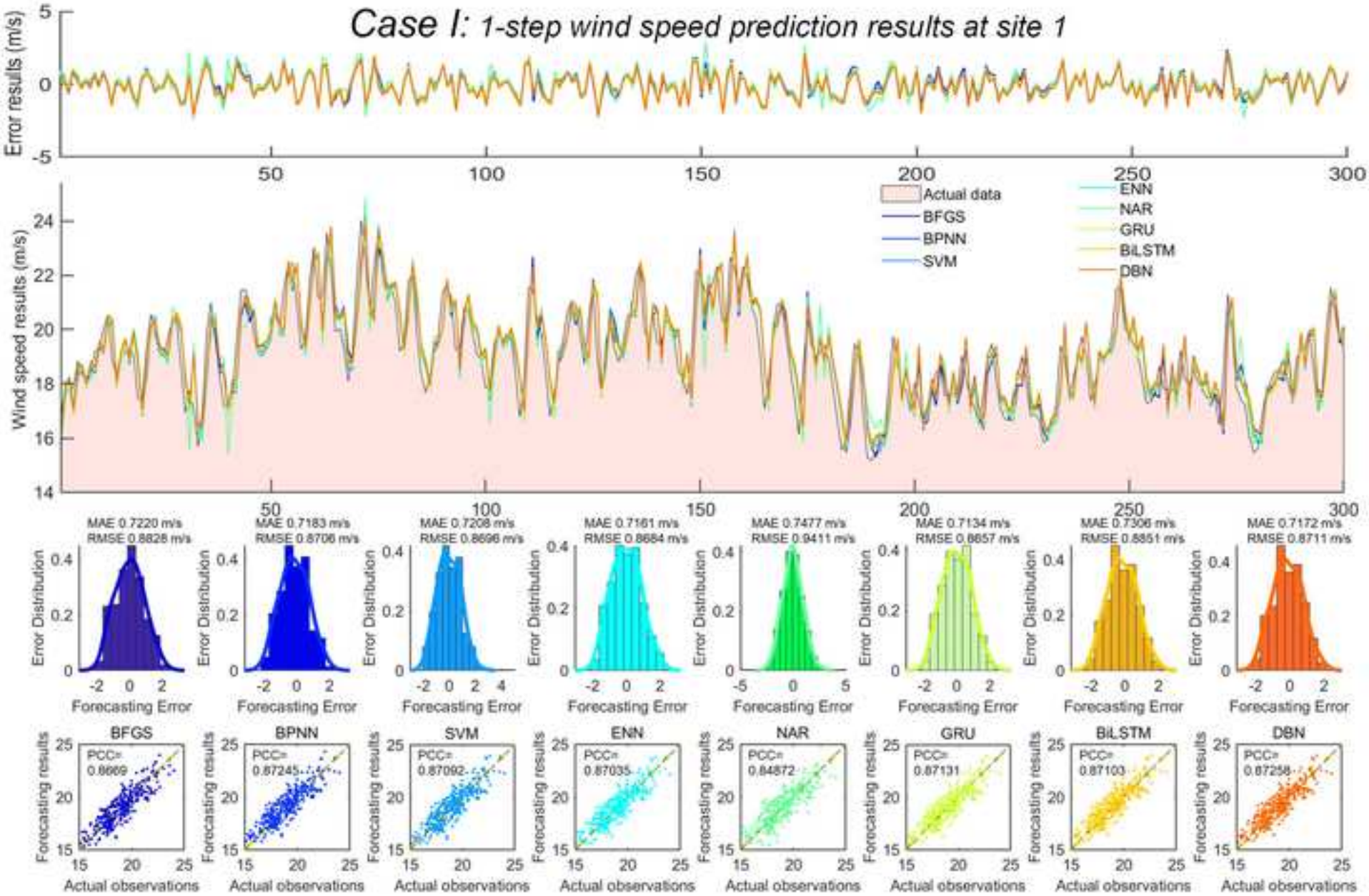
Site #1 Yanhu

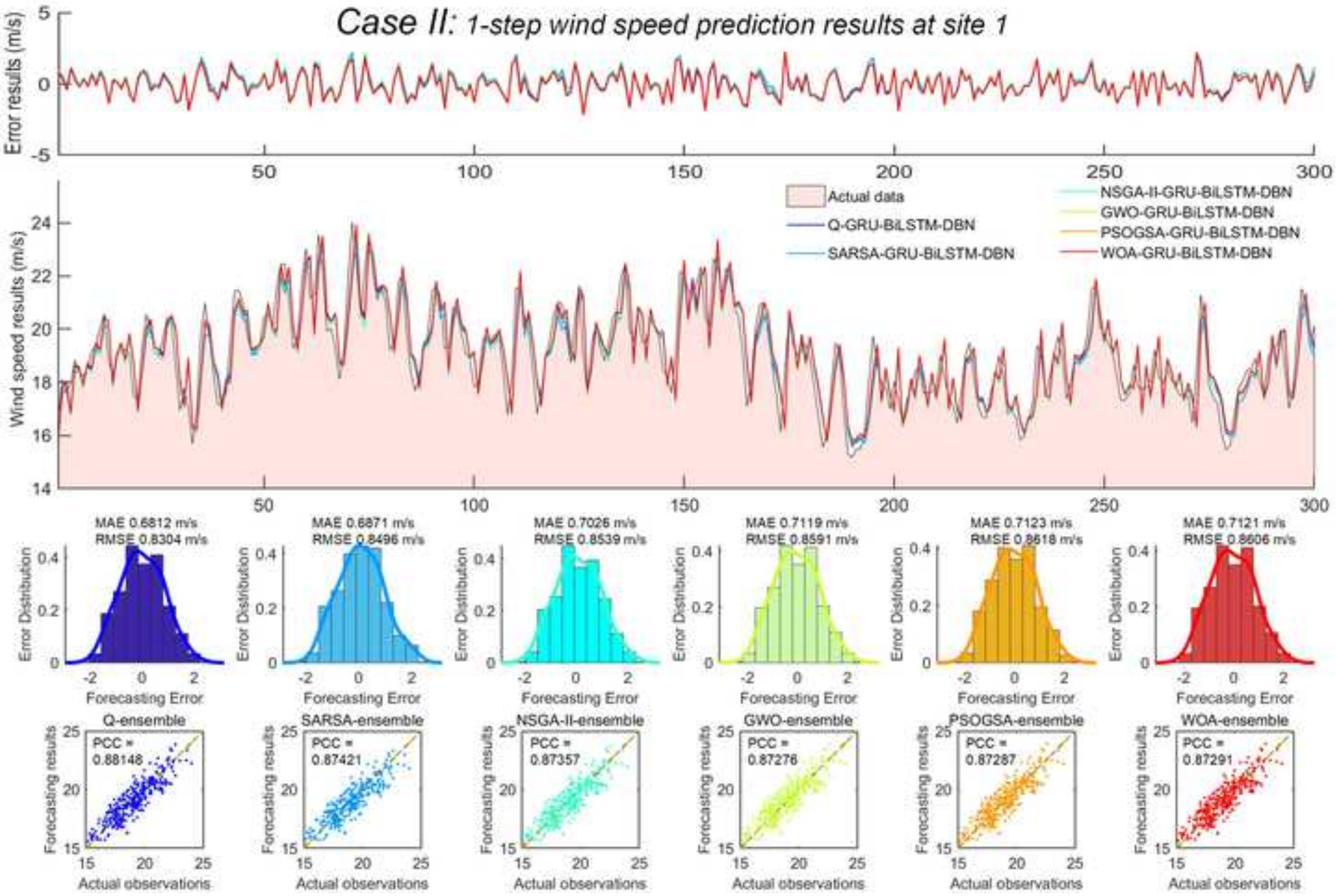
Site #2 Tianshan

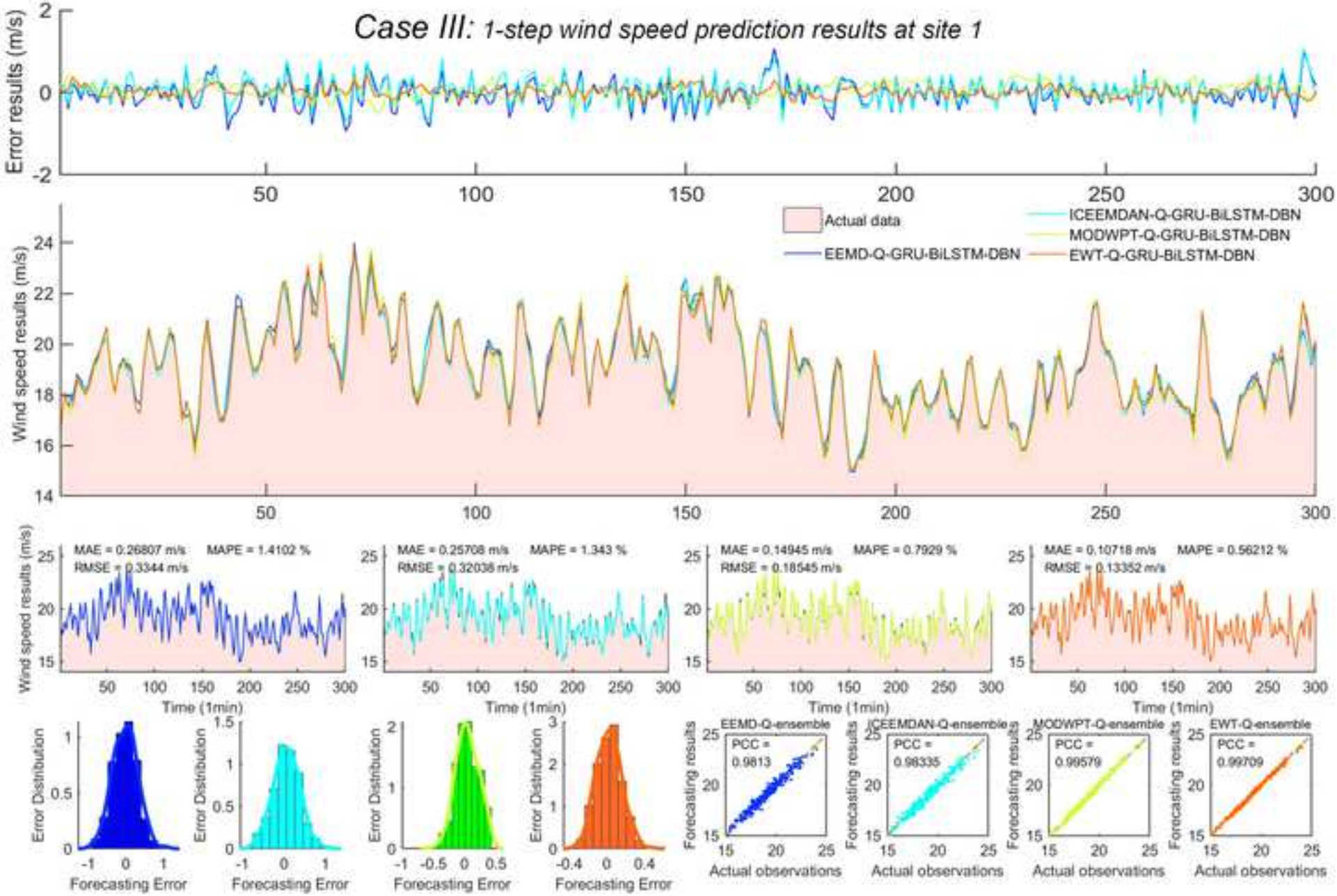
Site #3 Tulufan

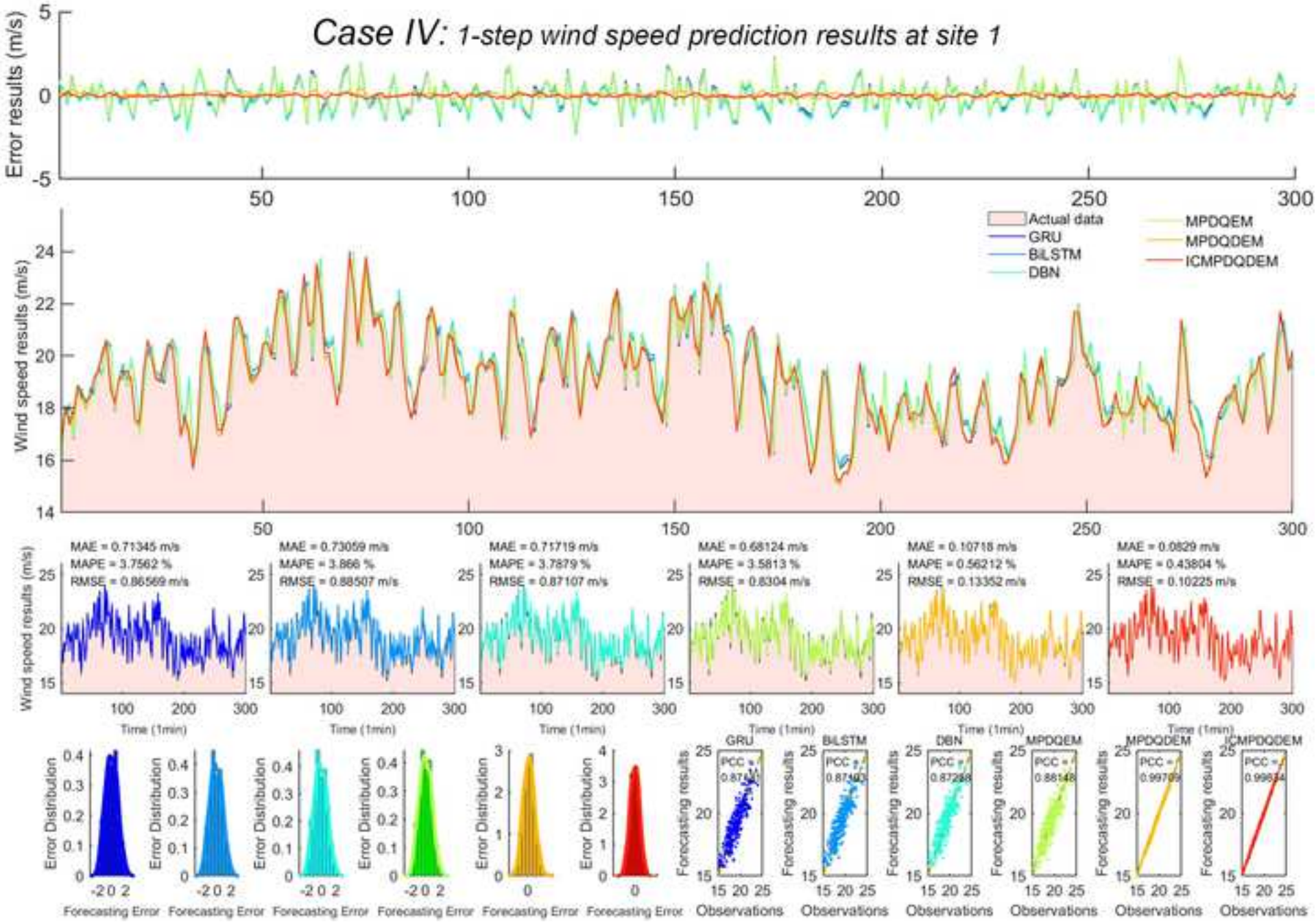
Site #4 Wulabo

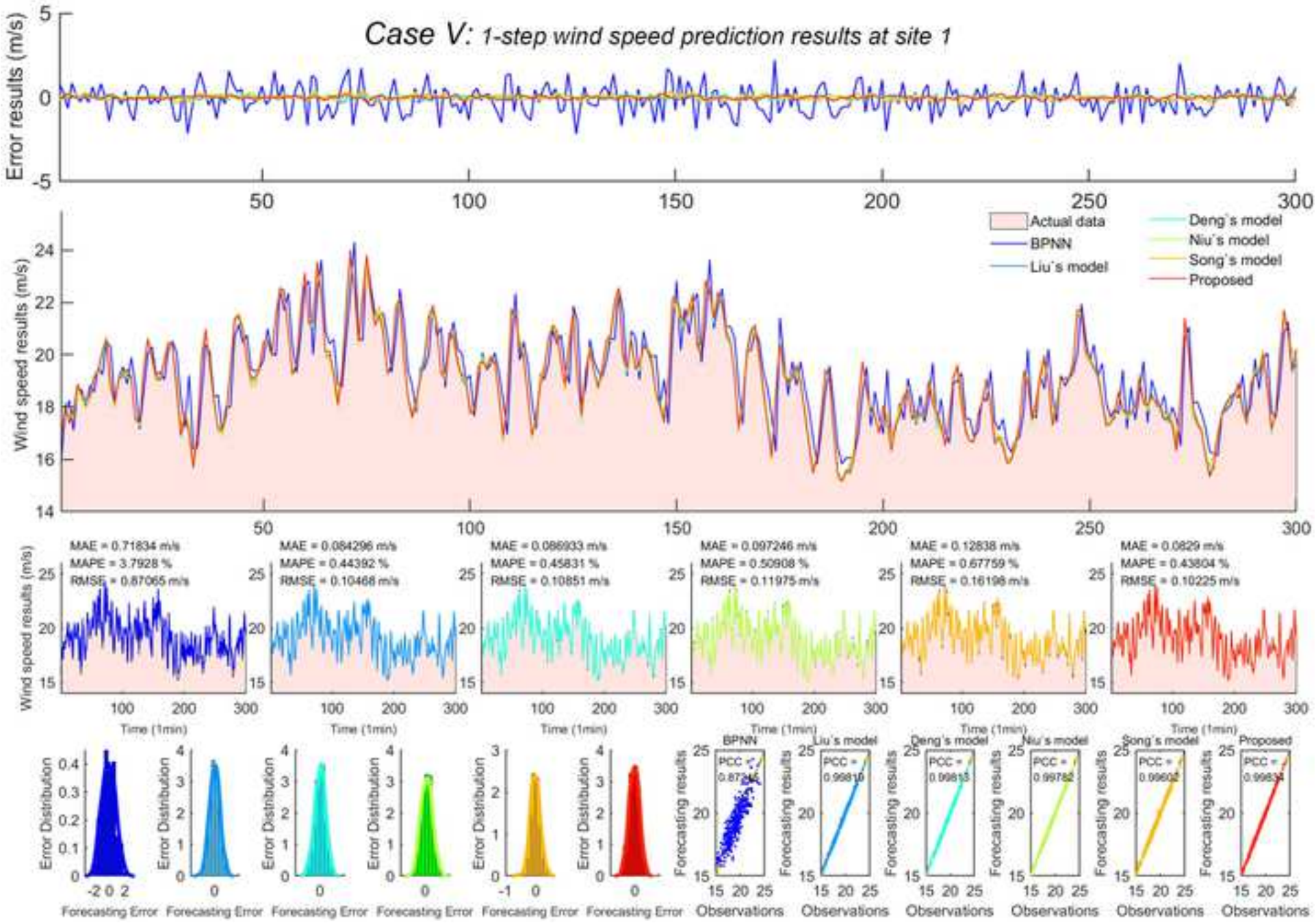
Site	Exploratory data analysis of wind speed data in four wind farm							
	Longitude	Latitude	Minimum	Maximum	Mean	Standard deviation	Median	Quantile
#1 Yanhu	88.1503° E	43.4031° N	11.49	26.67	19.57	2.52	19.59	15.28
#2 Tianshan	92.8308° E	43.5114° N	6.09	21.58	14.02	2.96	14.26	8.79
#3 Tulufan	89.1868° E	42.9498° N	11.73	24.50	18.01	2.08	18.14	14.43
#4 Wulabo	87.5956° E	43.6535° N	2.77	10.86	7.01	1.71	7.11	3.59

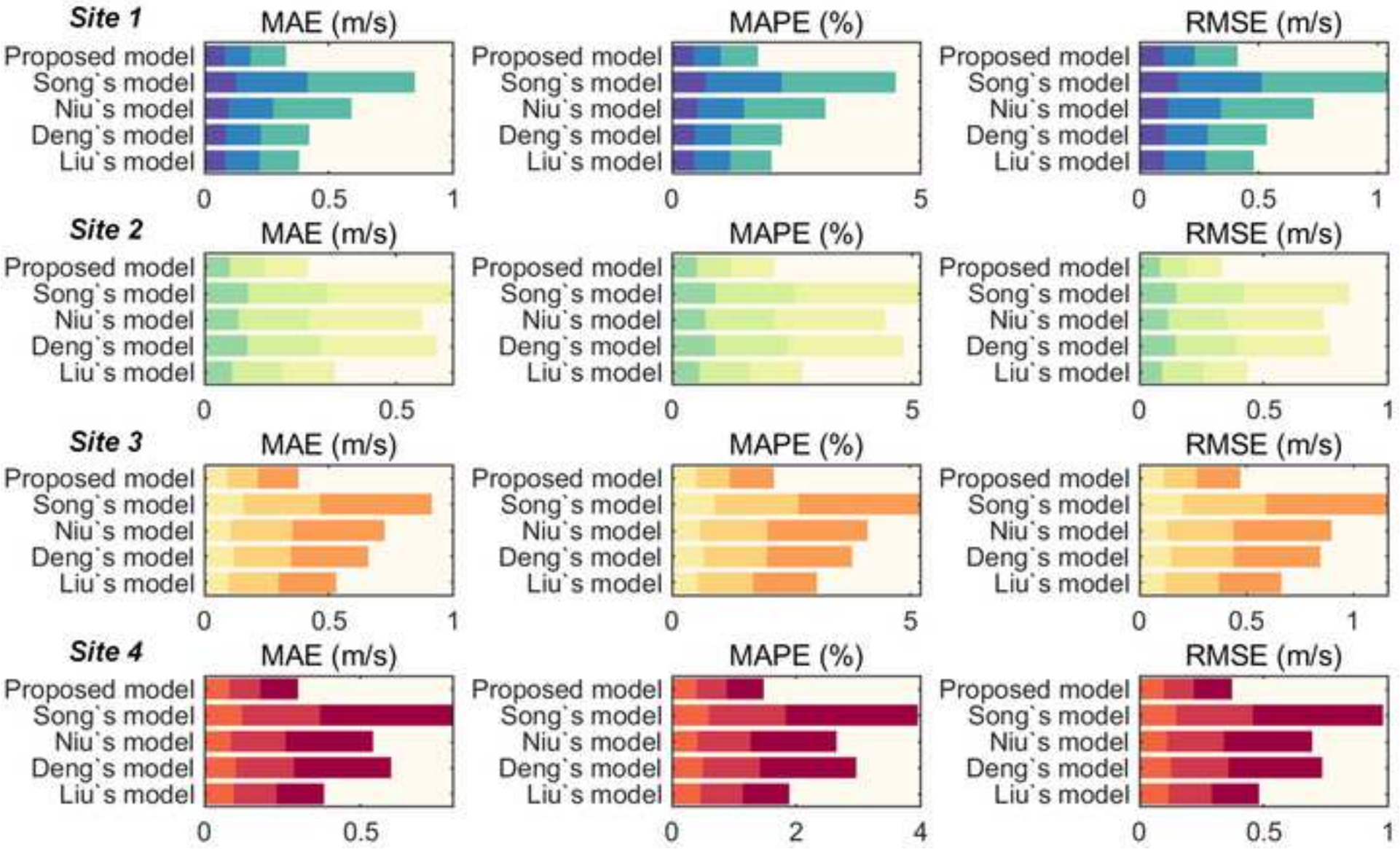












Credit author statement:

Rui Yang:
Algorithm derivation, Programming, Validation, Writing – original draft preparation.
Hui Liu:
Conceptualization, Methodology, Validation, Writing – reviewing and editing.
Nikolaos Nikitas:
Methodology, Writing – reviewing and editing
Zhu Duan:
Validation, Writing – reviewing and editing
Yanfei Li:
Algorithm derivation, Programming, Validation
Ye Li:
Programming, Validation