Wind Power Forecasting Using A Heterogeneous Ensemble of Decomposition-based NNRW Techniques

Pakarat Musikawan¹, Khamron Sunat², and Yanika Kongsorot³,

ABSTRACT
Accurate and reliable wind power forecasting plays a vital role in the operation and management of power systems. Hence, it has become necessary to research and develop a high-accuracy wind power forecasting model. However, owing to highly nonlinear and non-stationary patterns of wind power time-series, creating a wind forecasting model capable of predicting such series accurately is both complicated and challenging. Aiming at this challenge, this paper introduces a new decomposition-based hybrid model based on multiple decomposition techniques, neural network with random weights (NNRW), and a linear combiner. In our approach, the original time-series is decomposed into a collection of sub-series by different decomposition techniques. Each sub-series is modeled and predicted separately using NNRW. The predicted signals of each decomposition model are then reconstructed independently. Finally, all of the reconstructed results are integrated by the combiner using a linear combination method. The predictive performance of the proposed method was compared with other state-of-the-art techniques in over 12 wind power time-series. The experimental results show that the predictive performance of the proposed method was compared with other state-of-the-art techniques in over 12 wind power time-series. The experimental results show that the predictive performance of the proposed method remarkably outperforms the other competitors, proving the developed model to be effective, efficient, and practical.

Keywords: Wind power forecasting, Time-series, Neural network with random weights, Decomposition technique, Hybrid model, Ensemble system

1. INTRODUCTION
Wind energy is becoming more and more important as a worldwide energy supply. According to the report of the Global Wind Energy Council (GWEC), the global cumulative installed electricity generation capacity from wind power in 2018 was 51.3 gigawatts (GW) and the GWEC forecasts that the capacity of wind power generation will reach higher than 840 GW by the end of 2022 [1]. Moreover, the report released by the GWEC shows that the cumulative installed wind power capacity could reach 2000 GW by 2030. This illustrates that wind energy has gained greater distinction and has attracted global attention. Wind power is the conversion of energy from the wind into electricity, which is generated by the passing of airflow through wind turbines. The power generated is therefore dependent upon the wind speed. Accurate wind power forecasting is necessary for power system operations, such as planning, dispatching, and maintenance schedules. However, as wind speed is intermittent and fluctuating, it is not easy to model and predict accurately [2, 3].

In applications of wind power forecasting, decomposition-based hybrid approaches have been proposed based on the combination of decomposition techniques and forecasting models. Different decomposition techniques have been widely applied in the hybrid methods for preprocessing because they can effectively reduce the non-stationary characteristics of the wind power time series [4], including empirical mode decomposition (EMD) [5], variational mode decomposition (VMD) [6], discrete wavelet transform (DWT) [7], wavelet packet decomposition (WPD) [8], and singular spectrum analysis (SSA) [9]. These decomposition methods are used in the data preprocessing stage to decompose the time-series of wind power into several components. Then, a forecasting model is built for each decomposed component. Generally, conventional machine learning algorithms are usually utilized to perform as forecasting models [10]. They have been become the most dominant techniques in decomposition-based hybrid approaches, due to their forecasting ability. However, these algorithms require a lot of training time to iteratively find their optimal parameters. Hence, it is necessary to balance forecasting accuracy with the required computational time.

Non-iterative learning approaches have also been proposed to avoid some of the difficulties faced by iterative learning algorithms [11]. A neural network with random weights (NNRW) is a class of non-iterative learning algorithm for training NN with a fixed hidden layer size [12, 13, 14]. The weights and biases within the hidden layer of NNRW are randomly assigned, while the output layer parameters are de-
determined by finding the least square solution. Due to their ability to generate a forecasting model with extremely fast learning speed and satisfactory performance, they have attracted the attention of numerous research studies [15, 16, 17], especially in the area of wind energy [18, 19, 20]. According to previous literature [21], it is clear that NNRW can dramatically accelerate the computational speed of the decomposition-based hybrid approach.

Based on the aforementioned research, there have been a lot of successful applications of the individual decomposition technique integrated with NNRW in time-series forecasting. However, the single decomposition-based hybrid approach often cannot accurately capture the complex relationships existing in the highly nonlinear and non-stationary time-series. By borrowing the idea of ensemble learning that incorporates the advantages of different individual algorithms, this paper proposes a new decomposition-based hybrid approach, named *Heterogeneous Ensemble of Decomposition-based NNRW* (EDNNRW). In our approach, the original time-series is decomposed into a finite number of components through different decomposition techniques. Five prominent decomposition techniques (EMD, VMD, SSA, DWT, and WPD) have been applied in the decomposition process in the preprocessing of our system because they are extensively applied signal decomposition techniques that have been proven to be effective, rapid, and practicable data preprocessing tools in time-series forecasting [22, 23, 24, 25, 26]. To inherit the merits of fast learning, computational simplicity, and good generalization capabilities, four types of NNRW models are utilized to perform as predictors for each decomposed component in the forecasting process. The final forecasting results of each decomposition technique can be reconstructed by adding up all the predicted results. Finally, all of the reconstructed signals are integrated as the ultimate result via a linear combiner, due to its architectural simplicity, fast modeling, and functional approximation capabilities. Simulations on wind power forecasting have demonstrated that the developed framework integrating multiple decomposition techniques, NNRW, and a linear combiner. This method has not been found in previous studies to their ability to generate a forecasting model with extremely fast learning speed and satisfactory performance, they have attracted the attention of numerous research studies [15, 16, 17], especially in the area of wind energy [18, 19, 20]. According to previous literature [21], it is clear that NNRW can dramatically accelerate the computational speed of the decomposition-based hybrid approach.

The three main scientific contributions and novelties of this research are given in the following list:

1. We propose a new decomposition-based hybrid framework integrating multiple decomposition techniques, NNRW, and a linear combiner. This method has not been found in previous studies to the best knowledge of the authors.

2. EMD, VMD, SSA, DWT, and WPD were integrated into the developed framework to decompose the original signals to reduce the non-stationary characteristics as much as possible. This technique has also not been previously published.

3. Four types of NNRW methods were utilized as predictors of the developed decomposition-based hybrid approach. The impacts of various NNRW methods in the developed model were investigated and documented.

The remainder of this paper is organized as follows: the literature review is presented in Section 2; our proposed method is described in Section 3; our experimental results and performance evaluations are presented in Section 4; and lastly, the conclusions are illuminated in Section 5.

2. LITERATURE REVIEW

Many previously published studies have proposed different methods for wind power forecasting. These can be divided into four broad categories [27]: (a) physical methods, (b) statistical methods, (c) intelligent methods, and (d) hybrid methods. Each method, however, is not without its limitations. Physical methods build forecasting methods through physical or meteorological information, such as temperature, pressure, altitude, and so on. Their drawback is that they are very time-consuming [28]. Statistical methods model the predictors through the use of historical data including autoregression (AR), moving average (MA), the combination of AR and MA (ARMA), and AR integrated MA (ARIMA) [29]. Since these models are linear approaches, they are incapable of accurately predicting highly nonlinear or non-stationary time series. Intelligent methods primarily employ machine learning techniques to find the relationship between the input variables and the corresponding output data. Some of these approaches are support vector machine (SVM) [30], artificial neural network (ANN) [31], and ensemble systems [32]. Hybrid methods aggregate various methodologies together. Generally, hybrid approaches combine decomposition-based methods and predictors. They generally have better prediction performance than the previously mentioned approaches. The hybrid approaches provide effective forecasting performance as they combine the advantages of different methodologies, and have thus received increasing attention [33].

The decomposition technique is a powerful tool for reducing the forecast difficulty by converting the original non-stationary time series into several relatively more stationary sub-series. EMD is a self-adaptive analysis technique for the time-domain processing of a nonlinear and non-stationary signal [22]. The EMD decomposes a signal \( x = [x(1), \ldots, x(T)] \) into a finite collection of \( K - 1 \) intrinsic mode functions (IMFs) and one residue [22]. The group of IMFs \([u_1, \ldots, u_{K-1}]\) and the residue \( r \) can be mathematically expressed as \( x = r + \sum_{k=1}^{K-1} u_k \). VMD [23] is an adaptive and non-recursive signal decomposition algorithm which is appropriate for analyzing non-stationary signals. The VMD decomposes a signal into \( K \) components with limited bandwidth in the spectral domain. Both the bandwidth and
center frequency of each component are determined by iteratively searching for the optimal solution of a variational problem. DWT [25] is a mathematical technique and powerful tool for analyzing the time-frequency domain. It is well suited for non-stationary signals. The DWT decomposes the signal into a set of approximation and detail coefficients. The approximation and detail coefficients represent the low and high frequency components, respectively. DWT decomposes only the approximation coefficient at each level. The WPD [26] is a generalized version of DWT which decomposes both the approximation and detail coefficients at each level. SSA [24] is a non-parametric technique which is widely employed in time series analysis. The core purpose of this approach is decomposing an original time-series of data into a sum of sub-series in which each sub-series can be identified as either a trend, quasi-periodic component, or noise.

Jiang et al., 2012 [5] proposed a combination of the EMD, the largest Lyapunov exponent (LLE) prediction method, and the grey forecasting model. The EMD was employed as the data preprocessing approach to decompose the time-series of wind power into various IMF components and one residual component. Then, the LLE method was performed to predict each IMF. Finally, the grey forecasting model was employed to predict the residual component. Zhang et al., 2018 [6] proposed a hybrid prediction model with the VMD and a long short-term memory network (LSTM), called VMD-LSTM. In the first step, the wind power time-series is decomposed into various sub-series using the VMD. In the VMD-LSTM, the LSTM network is exploited to find each sub-series of wind power. Wang et al., 2020 [9] presented a hybrid of SSA and the Laguerre neural network (LNN) optimized by the opposition transition state transition algorithm (OTSTA). The time-series of wind power was decomposed into various sub-series using SSA in the first step. An optimized LNN was built for each sub-series. Catalão et al., 2011 [7] proposed a combination of the DWT and multilayer perceptron (MLP) trained by the Levenberge-Marquardt (LM) algorithm for wind power forecasting in Portugal. In the first stage, the DWT was used to decompose the wind power series into a set of sub-series. Then, the future values of these sub-series were predicted using the LM network. Laouafi et al., 2017 [8] presented a hybrid of the WPD and adaptive neuro-fuzzy inference system (ANFIS) for the prediction of wind power generation in France. As mentioned before, the literature shows that EMD, VMD, SSA, DWT, and WPD can improve the predictive performance of the hybrid approaches in applications of wind power forecasting. The iterative intelligent methods were adopted to perform as predictors in the hybrid methods. These algorithms are very time-consuming because they employ iterative learning methods for tuning their parameters.

NNRW was originally described by Schmidt et al. [12], and they called it a Schmidt neural network (SNN). It is a fast learning approach for training a single hidden layer feedforward network (SLFN) with a fixed hidden layer size. The hidden layer parameters of an SLFN trained by NNRW are randomly generated, whereas the output weights and output biases are analytically determined by finding a least-square solution. Pao et al. [13] proposed a variant version of NNRW named random vector function-link network (RVFL). The RVFL was developed for training a functional-link network (FLN) [34]. In the RVFL, direct connections from the input nodes to the output nodes were allowed. Another version of the RVFL [35] considered the output bias term, which herein we name RVFL#. Huang et al. [14] proposed a learning algorithm, referred to as extreme learning machine (ELM), for training SLFN. Unlike the original SNN, the bias term within the output layer of the ELM is not considered [11].

In the area of wind energy forecasting, several hybrids of decomposition techniques and NNRW have been proposed by researchers [18, 19, 20]. Abdoos [36] proposed a combination of the Gram-Schmidt orthogonalization (GSO), VMD, and ELM, called VMD-GSO-ELM. The VMD was utilized to decompose the wind power signal into several sub-series, and each decomposed sub-series was utilized to create the training patterns. Then, the GSO was employed as a feature selection method to eliminate irrelevant input features from each training dataset for ELM. Finally, the ELM was used as a forecasting model for each dataset with selected features. The experimental results showed that the VMD-GSO-ELM had faster learning speed than other iterative learning algorithms. Naik et al., 2018 [37] proposed a hybrid EMD and non-iterative learning approach for both wind speed and wind power predictions. Several non-iterative learning methods were selected and compared in this work, including kernel ridge regression (KRR), RVFL, and ELM. The experimental results showed that both a hybrid EMD and KRR (EMD-KRR), as well as a combination of EMD and RVFL (EMD-RVFL), could achieve promising results in applications of both wind speed and wind power forecasting. Their experimental results have also shown that although the predictive performance of EMD-RVFL was slightly lower than that of EMD-KRR, the training speed of EMD-RVFL was much faster than EMD-KRR. Moreover, there are a large number of parameters which must be set in the EMD-KRR algorithm and the EMD-KRR was very sensitive to the values of those parameters. This illustrates that these aforementioned hybrid techniques inherit the advantages of fast learning speed, and good generalization performance from NNRW. Moreover, they have shown excellent results working with time-series for wind power forecasting.
3. PROPOSED METHOD

Let \{f_1, \ldots, f_M\} denote a set of \(M\) base models, where the \(m\)th base model \(f_m\) is separately trained on \(\{X, Y\} = \{(x_i, y_i)\}_{i=1}^N\). Here, \(x_i = [x(1), \ldots, x(n)] \in \mathbb{R}^n\) is the \(i\)th input sample and \(y_i \in \mathbb{R}\) is the corresponding desired output. Suppose that the \((x_i, y_i)\) for the base model \(f_m\) can be decomposed into \(K\) components using the \(m\)th decomposition technique \(D_m\), that is \(\{\hat{x}_{i,k}^m, \hat{y}_{i,k}^m\}_{k=1}^K\}. \hat{x}_{i,k}^m\) and \(\hat{y}_{i,k}^m\) denote the \(k\)th decomposed components of \(x_i\) and \(y_i\) using the \(m\)th decomposition technique, respectively, where \(\hat{x}_{i,k}^m = [\hat{x}_{i,k}^m(1), \ldots, \hat{x}_{i,k}^m(n)] \in \mathbb{R}^n\), \(\hat{y}_{i,k}^m \in \mathbb{R}\), \(x_i = \sum_{k=1}^K \hat{x}_{i,k}^m\) and \(y_i = \sum_{k=1}^K \hat{y}_{i,k}^m\). Here, \(D_m \in \{\text{EMD}, \text{VMD}, \text{SSA}, \text{WPD}, \text{DWT}\}\}, and \(m = 1, \ldots, M\). Therefore, the base model \(f_m\) for a sample \(x_i\) with \(K\) components can be expressed as shown in Eq. (1).

\[
f_m(x_i) = \sum_{k=1}^K f_k^m(\hat{x}_{i,k}^m) \tag{1}
\]

\(f_k^m\) is the \(k\)th predictor within \(f_m\). In considering the influence of the type of network structure, \(f_k^m\) is given by Eq. (2).

\[
f_k^m(\hat{x}_{i,k}^m) = \sum_{i=1}^L \beta^m_{i,k} \sigma(\hat{x}_{i,k}^m; w_{i,k}^m, b_{i,k}^m) + \varphi(\hat{x}_{i,k}^m, \mu) \tag{2}
\]

\(\hat{x}_{i,k}^m\) represents the \(k\)th component decomposed from \(x_i\) using the \(m\)th decomposition technique. Here, \(w_{i,k}^m \in \mathbb{R}^n\) denotes the input weights that connect the input layer and the \(i\)th hidden node, and \(b_{i,k}^m\) is the bias of the \(i\)th hidden node. Both the weights and biases within the hidden layer are randomly generated based on a uniform distribution. \(\beta^m_{i,k}\) is the weight connecting the \(i\)th hidden node and the output layer of \(f_k^m\) within \(f_m\). \(\varphi\) denotes the structural function, which is adopted to define the type of network structure. The \(\varphi\) for an input \(\hat{x}_{i,k}^m\) is formulated as shown in Eq. (3).

\[
\varphi(\hat{x}_{i,k}^m, \mu) = \begin{cases} 
0, & \text{if } \mu = 0 \text{ (ELM)} \\
\beta^0_{0,k} + \sum_{l=L+1}^{L+n} \hat{x}_{j,k}^m(l-L)\beta^m_{l,k}, & \text{if } \mu = 1 \text{ (SNN)} \\
\beta^0_{0,k} + \sum_{l=L+1}^{L+n} \hat{x}_{j,k}^m(l-L)\beta^m_{l,k}, & \text{if } \mu = 2 \text{ (RVFL)} \\
\beta^0_{0,k} + \sum_{l=L+1}^{L+n} \hat{x}_{j,k}^m(l-L)\beta^m_{l,k}, & \text{if } \mu = 3 \text{ (RVFL)*}
\end{cases} \tag{3}
\]

\(\beta^0_{0,k}\) and \(\beta^m_{0,k}\) represent the output bias and the weight, respectively, that connect the \(l\)th input node and the output layer of the \(f_m\) for the \(k\)th component. Here, the SLFN structure is adopted if \(\mu \in \{0, 1\}\), while the FLN structure is employed if \(\mu \in \{2, 3\}\). The output weights within the \(m\)th base model for estimating the \(k\)th component are determined by the minimization in Eq. (4).

\[
\arg\min_{\{\hat{y}_{i,k}^m\}} \left\{ \sum_{j=1}^N \left[ y_j - f_m(\hat{x}_{j,k}^m) \right]^2 \right\} \tag{4}
\]

To combine the predicted results of all the base models, all the predicted results of the \(M\) base models are integrated through a linear combination method. Therefore, the ensemble output function can be written as shown in Eq. (5).

\[
F(x_j) = \omega_0 + \sum_{m=1}^M \omega_m f_m(x_j) \tag{5}
\]

\(\omega_m\) denotes the coefficient connecting the \(m\)th base model and the combination layer.

To obtain the optimal \(\{\omega_0, \ldots, \omega_M\}\), the objective function for minimizing the training error can be formulated as shown in Eq. (6).

\[
\arg\min_{\{\omega_m\}} \left\{ \sum_{j=1}^N \left[ y_j - \left( \omega_0 + \sum_{m=1}^M \omega_m f_m(x_j) \right) \right]^2 \right\} \tag{6}
\]

By using Eq. (6), the objective function can be rewritten in the matrix form as shown in Eqs. (7) to (9).

\[
\mathcal{L} = (\Phi \omega - Y)^\top (\Phi \omega - Y) = Y^\top Y + \omega^\top \Phi^\top \Phi \omega - \omega^\top \Phi^\top Y \tag{7}
\]

\[
= Y^\top Y + \omega^\top \Phi^\top \omega - 2 \omega^\top \Phi^\top Y
\]

\[
\Phi = \begin{bmatrix} f_1(x_1) & \ldots & f_M(x_1) \\ \vdots & \vdots & \vdots \\ 1 & \ldots & 1 \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \tag{8}
\]

\[
\omega = \begin{bmatrix} \omega_0 \\ \vdots \\ \omega_M \end{bmatrix}, \quad \Phi^\top \omega = \Phi^\top Y \tag{9}
\]

\(\Phi \in \mathbb{R}^{N \times (M+1)}\) is the output matrix of the base models. \(\omega \in \mathbb{R}^{(M+1) \times 1}\) and \(Y \in \mathbb{R}^{N \times 1}\) denote the ensemble weight and the desired output vectors, respectively.

Taking the derivative of \(\mathcal{L}\) with respect to \(\omega\) and setting the derivative equal to zero, we obtain Eq. (10).

\[
\frac{\partial \mathcal{L}}{\partial \omega} = 0 \rightarrow 2 \Phi^\top \Phi \omega - 2 \Phi^\top Y = 0 \tag{10}
\]

\[
\rightarrow \Phi^\top \Phi \omega = \Phi^\top Y
\]

Assuming that \(\Phi^\top \Phi\) is an invertible matrix, the optimal least square solution of Eq. (7) is given by Eq. (11).

\[
\omega = \Phi^\dagger Y \tag{11}
\]
4. All of the reconstructed results are integrated by
3. The predicted signals of each decomposition tech-
2. The NNRW predictor is built to complete the fore-
1. Each decomposition technique is used to decom-

\[ \Phi^\dagger = (\Phi^\top \Phi)^{-1} \Phi^\top \] is the generalized pseudoinverse of \( \Phi \). However, \( \Phi^\top \Phi \) can be a non-invertible matrix. To avoid the non-invertible problem, the SVD is commonly employed to compute the generalized pseudoinverse in all cases [14]. Therefore, the SVD was adopted to compute \( \Phi^\dagger \) in this study.

**Theorem 1** ([38, 39]) Given \( \Phi \in \mathbb{R}^{n \times m} \) such that \( \Phi \Phi^\top \) is the minimum norm least-square solution of \( \Phi x = b \), where \( \Phi \in \mathbb{R}^{n \times m} \), and \( b \in \mathbb{R}^m \). It is necessary and sufficient that \( \Phi = \Phi^\dagger \), which is the generalized inverse of \( \Phi \).

**Remarks 1:** According to Theorem 1, the proposed ensemble model has following important properties:

- \( x^* = \Phi^\dagger b \) is the least-square solution of \( \Phi x = b \)

\[
\| \Phi x^* - b \| = \| \Phi \Phi^\dagger b - b \| = \operatorname{argmin}_x \| \Phi x - b \| \tag{12}
\]

- \( x^* = \Phi^\dagger b \) has the minimum norm among all the other solutions of \( \Phi x = b \)

\[
\| x^* \| = \| \Phi^\dagger b \| \leq \| x \|, \quad \forall x \in \{ x : \| \Phi x - y \| \leq \| \Phi z - y \| , \forall z \in \mathbb{R}^n \} \tag{13}
\]

- \( x^* = \Phi^\dagger b \) is the minimum norm least-squares solution of \( \Phi x = b \), which is always unique.

The learning process of the proposed decomposition-based hybrid approach is summarized as follows:

1. Each decomposition technique is used to decompose the wind power series data. The time-series data of each decomposition method is decomposed into \( K \) decomposed components. In this step, five single decomposition techniques are adopted separately: EMD, VMD, SSA, DWT, and WPD.

2. The NNRW predictor is built to complete the forecasting computation for each decomposed component of each decomposition technique. In this step, four types of NNRW models are presented: ELM, SNN, RVFL, and RVFL×.

3. The predicted signals of each decomposition technique are directly summed to build the reconstructed time-series of wind power through Eq. (1).

4. All of the reconstructed results are integrated by a linear combination method using Eq. (5). The weighted coefficients of this combiner can be obtained via Eq. (11).

4. CASE STUDY AND RESULTS DISCUSSION

4.1 Datasets specification and preparation

Twelve actual wind power datasets were retrieved from the 50Hertz Transmission GmbH website. These are available at https://www.50hertz.com/. These datasets were collected over 12 months from January 1, 2018, to December 31, 2018, in Germany. These data series were recorded at an interval of 15 minutes. The series of the wind power datasets were continuously recorded with the exception of March 25, 2018, from 2:00 to 2:45, and November 26, 2018, at 12:00 and 12:15, when data was not collected. The specification and statistical information including mean, maximum (Max.), minimum (Min.), standard deviation (SD), skewness (Skew.), and kurtosis (Kurt.) values of each dataset are detailed in Table 1.

<p>| Table 1: Statistical information for the wind power datasets. |</p>
<table>
<thead>
<tr>
<th>Dataset #</th>
<th>Sample Mean</th>
<th>Max.</th>
<th>Min.</th>
<th>SD</th>
<th>Skew.</th>
<th>Kurt.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>2976</td>
<td>6092.09</td>
<td>14354.70</td>
<td>90.48</td>
<td>4265.99</td>
<td>0.30</td>
</tr>
<tr>
<td>Feb</td>
<td>2688</td>
<td>2771.60</td>
<td>10329.50</td>
<td>11.67</td>
<td>2283.86</td>
<td>0.91</td>
</tr>
<tr>
<td>Mar</td>
<td>2972</td>
<td>4458.47</td>
<td>13773.59</td>
<td>85.37</td>
<td>3522.37</td>
<td>0.81</td>
</tr>
<tr>
<td>Apr</td>
<td>2880</td>
<td>3866.14</td>
<td>12935.60</td>
<td>25.05</td>
<td>2865.42</td>
<td>0.49</td>
</tr>
<tr>
<td>May</td>
<td>2976</td>
<td>2888.28</td>
<td>11406.72</td>
<td>89.84</td>
<td>2097.93</td>
<td>1.05</td>
</tr>
<tr>
<td>Jun</td>
<td>2880</td>
<td>2545.68</td>
<td>12204.81</td>
<td>59.60</td>
<td>2071.62</td>
<td>1.79</td>
</tr>
<tr>
<td>Jul</td>
<td>2976</td>
<td>1911.56</td>
<td>6085.28</td>
<td>15.07</td>
<td>1440.31</td>
<td>0.90</td>
</tr>
<tr>
<td>Aug</td>
<td>2976</td>
<td>2567.85</td>
<td>6987.07</td>
<td>90.10</td>
<td>1963.07</td>
<td>1.11</td>
</tr>
<tr>
<td>Sep</td>
<td>2976</td>
<td>3300.70</td>
<td>13057.33</td>
<td>127.15</td>
<td>2841.73</td>
<td>1.24</td>
</tr>
<tr>
<td>Oct</td>
<td>2978</td>
<td>4884.56</td>
<td>15382.83</td>
<td>11.03</td>
<td>4060.25</td>
<td>0.78</td>
</tr>
<tr>
<td>Nov</td>
<td>2878</td>
<td>3837.43</td>
<td>12577.33</td>
<td>90.10</td>
<td>2943.53</td>
<td>0.83</td>
</tr>
<tr>
<td>Dec</td>
<td>2976</td>
<td>6388.35</td>
<td>15727.40</td>
<td>275.67</td>
<td>4022.16</td>
<td>0.47</td>
</tr>
</tbody>
</table>

All of the experiments were performed using 30 independent runs. In each run, the dataset was divided into training and test sets, in which the first 80% was designated for training, and the last 20% was assigned for testing. In our experiment, the min-max normalization method was adopted to scale the time-series data to values in the range of 0 to 1.

4.2 Evaluation metrics

To evaluate the predictive performance of different comparative algorithms, three well-known error measurement indices were considered. These metrics include the root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). The details of these error measurements are given in Table 2.

<p>| Table 2: Evaluation metrics for measuring predictive accuracy. |</p>
<table>
<thead>
<tr>
<th>Criteria</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root mean square error (RMSE)</td>
<td>[ \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2} ]</td>
</tr>
<tr>
<td>Mean absolute error (MAE)</td>
<td>[ \frac{1}{N} \sum_{i=1}^{N}</td>
</tr>
<tr>
<td>Mean absolute percentage error (MAPE)</td>
<td>[ \frac{1}{N} \sum_{i=1}^{N} \frac{</td>
</tr>
</tbody>
</table>

To further evaluate the enhancement of model \( A \) over model \( B \), the improvement percentage of each criterion was exploited to illustrate the promotion degree, which can be expressed as shown in Eq. (14).

\[ I_p = \left| \frac{E_A - E_B}{E_B} \right| \times 100 \tag{14} \]
\( \nu \) is the RMSE, MAE, or MAPE. Here, \( E_A \) and \( E_B \) are the evaluated values of model \( A \) and \( B \), respectively, using measure \( \nu \).

4.3 Comparative algorithms and parameter settings

To verify the effectiveness of the proposed EDNNRW, ten comparative algorithms were selected for comparison with the proposed method. The selected algorithms were ELM [14], SNN [12], RVFL [13], RVFL\* [35], VMD-WRELM [18], EMDDRVFL [37], CVAELM [19], WPD-EMD-ELM [40], CEEMDAN-ANN [41], and VMD-GSO-ELM [36].

Following previous studies [42, 28, 43], three-level decomposition of WPD was applied in this study. Since the three-level WPD provides eight frequency bands, the maximum number of decomposed components for each decomposition method was eight. For all algorithms, an additive sigmoid function was applied as the nonlinear mapping activation function for the hidden layer. The 15-minute historical data values of the wind power series in the past day (24 hours) were considered as the input for prediction of the desired value. The maximum number of lag orders (features) was empirically set to 24 \( \times \) 4 = 96. The other parameters of each competing algorithm were set to the same as those used in the corresponding published research.

4.4 Comparison of statistical error measures

The predictive performance comparisons of the different algorithms in one, three, and five step ahead forecasting for the wind power predictions are tabulated in Tables 3 to 8. As shown in these tables, we find that the proposed EDNNRWRVFL and EDNNWRVRFL\* produce a relatively better forecasting accuracy than the other comparative algorithms in most cases. The average RMSE, MAE, and MAPE of the proposed EDNNRWEELM and EDNNRWSSN are generally lower than those of the ELM, SNN, VMD-WRELM, CVAELM, WPD-EMD-ELM, VMD-GSO-ELM, EMD-RVFL, and CEEMDAN-ANN. From these tables, it can be observed that RVFL, RVFL\* and the decomposition-based RVFL methods (EMD-RVFL, EDNNRWRVFL, and EDNNWRVRFL\*) have good forecasting abilities. This indicates that the direct connections between the input layer and the output layer can significantly improve the predictive performance of these models. Interestingly, we observed that the proposed EDNNRWRVFL and EDNNWRVRFL\* approaches do not need a large number of hidden nodes to attain good predictive performance. In Tables 6 to 8, we observed that the predictive performance of all the competitors decreases as the number of n-step ahead increases. This indicates that it gets harder to accurately capture the complex relationships existing in the multi-step ahead forecasting as \( n \) increases.

For multiple-comparison tests, the Friedman statistical test was employed to perform multiple-comparison tests for multiple-problem analysis, as suggested in [44]. Under the null hypothesis, the performance of all \( k \) competitors are equivalent, so their average ranks \( R_j \) over all \( N \) benchmarks should be equal. The Friedman statistic \( (\chi^2_F) \) can be calculated as shown in Eq. (15).

\[
\chi^2_F = \frac{12N}{k(k+1)} \sum_{j=1}^{k} R_j^2 - \frac{k(k-1)^2}{4}
\]

The \( F_F \) is distributed according to the chi-square distribution with \( k-1 \) degree of freedom whenever the values of \( N \) and \( k \) are sufficiently large. As a rule of a thumb, \( N > 10 \) and \( k > 5 \) [44].

Iman and Davenport [45] showed that the \( \chi^2_F \) is undesirably conservative, and presented an improved version of the \( \chi^2_F \), called the Iman-Davenport test \( (F_F) \) which is computed with Eq. (16).

\[
F_F = \frac{(N-1) \chi^2_F}{N(k-1) - \chi^2_F}
\]

The \( F_F \) is distributed according to the \( F \)-distribution with \( k-1 \) and \( (k-1)(N-1) \) degrees of freedom.

In this experiment, the number of competitors is 13. There are twelve time-series datasets for this experiment. For each dataset, one, three, and five step ahead forecasting were considered. Three different sizes of hidden layers with 25, 50, and 100 nodes were tested. Three evaluation metrics were utilized. Therefore, \( N = 12 \times 3^2 = 324 \), and \( k = 14 \).

In our case, the \( \chi^2_F \) value for this experiment is equal to 3213.41, and thus the \( F_F \) value is 1039.40. The critical value for the \( F \)-distribution with \( 14-1 = 13 \) and \( (14-1)(324-1) = 4199 \) degrees of freedom at a 0.05 significance level is 1.72. Because the value of \( F_F \) is greater than the critical value of the \( F \)-distribution,
## Table 3: Comparison of the RMSE for single-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>#Node</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELM</td>
<td>25</td>
<td>107.899</td>
<td>58.170</td>
<td>52.996</td>
<td>793.069</td>
<td>336.040</td>
<td>358.193</td>
<td>412.041</td>
<td>517.072</td>
<td>638.924</td>
<td>661.249</td>
<td>634.932</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>360.899</td>
<td>336.264</td>
<td>331.493</td>
<td>139.040</td>
<td>139.293</td>
<td>139.493</td>
<td>139.693</td>
<td>139.893</td>
<td>139.993</td>
<td>139.993</td>
<td>139.993</td>
<td></td>
</tr>
<tr>
<td>SNN</td>
<td>25</td>
<td>945.192</td>
<td>246.644</td>
<td>217.227</td>
<td>239.192</td>
<td>60.912</td>
<td>60.812</td>
<td>60.712</td>
<td>60.612</td>
<td>60.512</td>
<td>60.412</td>
<td>60.312</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>1643.440</td>
<td>492.611</td>
<td>524.031</td>
<td>776.627</td>
<td>380.368</td>
<td>380.468</td>
<td>380.568</td>
<td>380.668</td>
<td>380.768</td>
<td>380.868</td>
<td>380.968</td>
<td></td>
</tr>
<tr>
<td>RFVL</td>
<td>25</td>
<td>121.671</td>
<td>107.397</td>
<td>93.828</td>
<td>83.286</td>
<td>82.164</td>
<td>77.876</td>
<td>82.021</td>
<td>91.084</td>
<td>95.094</td>
<td>83.234</td>
<td>55.873</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>127.645</td>
<td>106.945</td>
<td>95.876</td>
<td>85.425</td>
<td>84.312</td>
<td>80.054</td>
<td>84.201</td>
<td>93.254</td>
<td>97.264</td>
<td>85.404</td>
<td>57.973</td>
<td></td>
</tr>
<tr>
<td>EDNN</td>
<td>25</td>
<td>134.328</td>
<td>117.714</td>
<td>88.266</td>
<td>50.405</td>
<td>85.537</td>
<td>85.777</td>
<td>85.954</td>
<td>101.349</td>
<td>113.085</td>
<td>152.142</td>
<td>120.694</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>125.832</td>
<td>108.374</td>
<td>90.326</td>
<td>50.405</td>
<td>85.537</td>
<td>85.777</td>
<td>85.954</td>
<td>101.349</td>
<td>113.085</td>
<td>152.142</td>
<td>120.694</td>
<td></td>
</tr>
<tr>
<td>RW</td>
<td>25</td>
<td>127.368</td>
<td>108.945</td>
<td>84.869</td>
<td>83.947</td>
<td>83.814</td>
<td>80.543</td>
<td>84.381</td>
<td>93.204</td>
<td>97.114</td>
<td>85.374</td>
<td>57.921</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>134.328</td>
<td>117.714</td>
<td>88.266</td>
<td>50.405</td>
<td>85.537</td>
<td>85.777</td>
<td>85.954</td>
<td>101.349</td>
<td>113.085</td>
<td>152.142</td>
<td>120.694</td>
<td></td>
</tr>
</tbody>
</table>

## Table 4: Comparison of the MAE for single-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>#Node</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
<td>370.968</td>
<td>386.977</td>
<td>380.368</td>
<td>597.738</td>
<td>275.451</td>
<td>299.329</td>
<td>365.125</td>
<td>430.517</td>
<td>503.417</td>
<td>685.363</td>
<td>437.841</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>370.968</td>
<td>386.977</td>
<td>380.368</td>
<td>597.738</td>
<td>275.451</td>
<td>299.329</td>
<td>365.125</td>
<td>430.517</td>
<td>503.417</td>
<td>685.363</td>
<td>437.841</td>
<td></td>
</tr>
</tbody>
</table>

## Table 5: Comparison of the MAPE for single-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>#Node</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
<td>370.968</td>
<td>386.977</td>
<td>380.368</td>
<td>597.738</td>
<td>275.451</td>
<td>299.329</td>
<td>365.125</td>
<td>430.517</td>
<td>503.417</td>
<td>685.363</td>
<td>437.841</td>
<td></td>
</tr>
</tbody>
</table>

### Notes
- The tables show the comparison of the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) for single-step ahead forecasting on wind power datasets using different algorithms.
- The tables compare the performance of ELM, SNN, RVFL, and EDNN algorithms.
- The data is organized by algorithm, #Node, and month (Jan, Feb, Mar, ..., Nov, Dec).
- Each row shows the error metrics for a specific algorithm and #Node combination.
- The error metrics are presented for each month, with values indicating the forecast error.
- The tables are structured to easily compare the performance across different algorithms and nodes.

---

Table 5: Comparison of the MAPE for single-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Node</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sep</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNN</td>
<td>10</td>
<td>3.8868</td>
<td>15.2657</td>
<td>8.8867</td>
<td>7.8598</td>
<td>6.2778</td>
<td>5.3055</td>
<td>4.8465</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNN</td>
<td>50</td>
<td>9.9242</td>
<td>3.0710</td>
<td>6.0146</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
</tr>
<tr>
<td>RVFL*</td>
<td>10</td>
<td>3.6608</td>
<td>2.5630</td>
<td>4.0854</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
</tr>
<tr>
<td>VMD-GSO-ELM</td>
<td>10</td>
<td>6.1948</td>
<td>4.4566</td>
<td>2.9857</td>
<td>2.1544</td>
<td>1.8349</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CEEMDAN-ANN</td>
<td>40</td>
<td>10.6918</td>
<td>8.2632</td>
<td>6.7875</td>
<td>5.0343</td>
<td>3.5192</td>
<td>2.3288</td>
<td>1.6746</td>
<td>0.4393</td>
<td>0.5287</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNN</td>
<td>50</td>
<td>9.9242</td>
<td>3.0710</td>
<td>6.0146</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
<td>2.6837</td>
</tr>
<tr>
<td>RVFL*</td>
<td>10</td>
<td>3.6608</td>
<td>2.5630</td>
<td>4.0854</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
<td>2.6661</td>
</tr>
</tbody>
</table>

we reject the null hypothesis that the predictive performance of all competitors are statistically equivalent.

Based on this null hypothesis rejection, the Ne- menyi post-hoc test was conducted to determine whether the predictive performance of two among k competitors are significantly different. The value of $q_{\alpha}$ for a 0.05 significance level is $q_{0.05} = 3.354$, which can be confirmed in standard statistical textbooks. Thus, the value of $C_D$ is equal to $3.354 \sqrt{\frac{14}{14+1}} \approx 1.102$. The statistical results of the post-hoc analyses for the wind forecasting are presented using a critical difference diagram, as shown in Figure 1. In this figure, the algorithms with higher ranks (lower numbers) are preferable to those with lower ranks (higher numbers). Statistically equivalent algorithms are grouped into a clique, represented by a red horizontal bar.

In Figure 1, the overall performance of RVFL, RVFL*, EDNNRW_{RVFL}, and EDNNRW_{RVFL*} were comparable. This figure shows that the overall predictive performance of the proposed EDNNRW_{RVFL} and EDNNRW_{RVFL*} were significantly superior to the other competitors. The overall predictive performance of VMD-GSO-ELM and EMD-RVFL were significantly better than those of VMD-WRELM, CVAELM, WPD-EMD-ELM, and CEEMDAN-ANN. Interestingly, we observed that the FLN family approaches (RVFL and RVFL*) have good average ranks when compared with ELM, SNN, VMD-WRELM, CVAELM, WPD-EMD-ELM, VMD-GSO-ELM, and CEEMDAN-ANN. It is noteworthy that the proposed EDNNRW_{RVFL} and EDNNRW_{RVFL*} had statistically significantly better average ranks than the other comparative algorithms, and their clique was located far from other cliques with large gaps.

The forecasting results and the corresponding residual errors of the different competitors in five step ahead forecasting for the wind power forecasts are depicted in Figure 2. From this figure, it can be observed that the proposed EDNNRW_{RVFL} has good forecasting abilities and its residual errors are closer to zero than those of the other comparative algorithms in most cases.

4.5 Comparison of improvement percentages

To further exhibit the effectiveness of the proposed EDNNRW_{RVFL}, the improvement percentages in terms of RMSE, MAE, and MAPE were used for analysis. The improvement percentages of the proposed EDNNRW_{RVFL} over ELM, SNN, RVFL, RVFL*, VMD-WRELM, EMD-RVFL, CVAELM, WPD-EMD-ELM, VMD-GSO-ELM, and CEEMDAN-ANN in terms of RMSE, MAE, and MAPE for the wind power forecast in one, three, and five step ahead forecasting are presented in Table 6.
Table 6: Comparison of the RMSE for multiple-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ELM</td>
<td>25</td>
<td>25</td>
<td>1196.1308</td>
<td>1377.1179</td>
<td>648.1125</td>
<td>778.2142</td>
<td>645.0175</td>
<td>788.0661</td>
<td>1038.7537</td>
<td>1108.3298</td>
<td>482.7572</td>
<td>577.1378</td>
<td>490.0680</td>
<td>626.0798</td>
<td>...</td>
<td>152.4094</td>
<td>193.2458</td>
<td>221.1502</td>
<td>265.0290</td>
<td>266.0228</td>
<td>294.7846</td>
<td>353.6811</td>
<td>392.6975</td>
<td>341.1524</td>
<td>393.5398</td>
<td>325.6738</td>
<td>374.5463</td>
<td></td>
</tr>
<tr>
<td>EDNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RVFL</td>
<td>25</td>
<td>25</td>
<td>42.2337</td>
<td>...</td>
<td>75.2109</td>
<td>27.3967</td>
<td>133.3630</td>
<td>22.2062</td>
<td>68.6445</td>
<td>22.9135</td>
<td>66.3602</td>
<td>25.4590</td>
<td>74.3240</td>
<td>22.7256</td>
<td>58.1057</td>
<td>199.0248</td>
<td>542.4897</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>RVFL*</td>
<td>25</td>
<td>42.0043</td>
<td>88.9692</td>
<td>...</td>
<td>26.7528</td>
<td>77.4241</td>
<td>130.4685</td>
<td>22.6307</td>
<td>69.2416</td>
<td>22.9990</td>
<td>67.6275</td>
<td>25.4835</td>
<td>73.9961</td>
<td>23.0584</td>
<td>58.7818</td>
<td>164.1982</td>
<td>540.4274</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Table 7: Comparison of the MAE for multiple-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EDNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RVFL</td>
<td>25</td>
<td>440.2042</td>
<td>544.6537</td>
<td>216.6104</td>
<td>258.4190</td>
<td>226.3062</td>
<td>270.3570</td>
<td>408.2153</td>
<td>573.5800</td>
<td>167.6098</td>
<td>213.7452</td>
<td>163.5987</td>
<td>206.4882</td>
<td>211.8578</td>
<td>...</td>
<td>128.4573</td>
<td>184.5987</td>
<td>146.5223</td>
<td>222.4921</td>
<td>138.9022</td>
<td>203.0687</td>
<td>176.5799</td>
<td>257.6755</td>
<td>117.2056</td>
<td>156.8525</td>
<td>605.0503</td>
<td>858.6638</td>
</tr>
<tr>
<td>RVFL*</td>
<td>25</td>
<td>15.5253</td>
<td>...</td>
<td>24.9177</td>
<td>12.9580</td>
<td>34.8300</td>
<td>15.4440</td>
<td>44.3740</td>
<td>16.2257</td>
<td>48.6382</td>
<td>17.4154</td>
<td>50.7295</td>
<td>18.7768</td>
<td>53.7340</td>
<td>...</td>
<td>14.8889</td>
<td>37.7754</td>
<td>46.0677</td>
<td>142.1672</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Note:** The tables continue with similar comparisons for the test datasets.
Table 8: Comparison of the MAPE for multiple-step ahead forecasting on the wind power datasets.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RVFL</td>
<td>250.3634</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RVFL*</td>
<td>25</td>
<td>0.3636</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 1: Critical difference diagram of average ranks of all the algorithms. The average rank of each competitor over all the datasets is plotted on the x-axis. The average ranks of the proposed methods are represented by blue lines. The best average rank is on the right side.

The data in Tables 9 to 11 can be summarized as follows:

1. When comparing the EDNNRW_RVFL with the competitors, the forecasting accuracy of the proposed model is remarkably higher than all of the other methods. The improvement percentages between the proposed EDNNRW_RVFL and the other competing algorithms, except for the EMD-RVFL and the VMD-GSO-ELM, were generally higher than 90%.

2. When comparing the EDNNRW_RVFL with EMD-RVFL and VMD-GSO-ELM, the forecasting accuracy of the former method is higher than the two latter approaches. The improvement percentages of the EDNNRW_RVFL over EMD-RVFL and VMD-GSO-ELM were generally higher than 60%.

3. The improvement percentages between the proposed EDNNRW_RVFL and VMD-GSO-ELM were lower than those of the others because the VMD-GSO-ELM employed the feature selection process to eliminate irrelevant features. Thus, we conclude that the feature selection process can improve the predictive performance of the VMD-GSO-ELM.

4. The improvement percentages between the proposed EDNNRW_RVFL and EMD-RVFL were lower than those of the others. Thus, we conclude that the FLN family promotes the predictive performances of the decomposition-based hybrid approaches. This indicates that the direct connections between the input layer and the output layer within the predictors can significantly improve the
Fig. 2: Results of five step ahead forecasting and the corresponding residual errors for the wind power datasets. Grey shaded regions represent the intervals at night time.
Fig. 2: Results of five step ahead forecasting and the corresponding residual errors for the wind power datasets. Grey shaded regions represent the intervals at night time (Cont.).
predictive performance of the decomposition-based hybrid model.

5. The performance of the proposed EDNNRW\textsubscript{RVFL} is relatively superior to the other comparative algorithms in terms of forecasting capability, thereby indicating a significant improvement exists in the predictive performance of the EDNNRW\textsubscript{RVFL}.

4.6 Comparison of computational times

All experiments were conducted in the MATLAB environment and run on a personal computer with an Intel Core i7-3370 3.40 GHz processor, 8 GB of RAM, and Windows 7 x64 operating system. The computational times of all competitors were obtained using the \textit{tic} and \textit{toc} commands in the MATLAB program. The average computational time of each algorithm across all the problems is shown in Figure 3. This figure shows that the computational times of the NNRW algorithms (ELM, SSN, RVFL, and RVFL*) were faster among all competing algorithms, due to the benefit of random weights generation and the closed-form least-squares solution. The computational speeds of the proposed EDNNRW algorithms were much faster than that of the CEEMDANN-ANN. In the CEEMDANN-ANN, the estimators must be iteratively fine-tuned by the back-propagation algorithm to obtain the optimal weight parameters. Consequently, the CEEMDANN-ANN was the most time-consuming technique. This supports our hypothesis that the algorithm for training the predictors in the decomposition-based method should be a non-iterative learning approach. As seen in Figure 4, the proposed EDNNRW\textsubscript{ELM}, EDNNRW\textsubscript{SSN}, EDNNRW\textsubscript{RVFL}, and EDNNRW\textsubscript{RVFL*} achieved good trade-offs between predictive performance and computational speed compared to other decomposition-based hybrid methods. Although the computational speeds of the proposed EDNNRW\textsubscript{RVFL} and EDNNRW\textsubscript{RVFL*} were slower than some decomposition-based approaches, the forecasting accuracy obtained by the proposed methods is dramatically increased. In practical applications, the additional accuracy is worth the extra computational time.

5. CONCLUSION

We developed an improved decomposition-based hybrid approach for wind power forecasting using EMD, VMD, SSA, DWT, WPD, NNRW, and a linear combiner. In our approach, each decomposition technique is applied to decompose the original time-series data into a collection of components. The NNRW is then exploited as an estimator for each decomposed component. After the reconstruction of the predicted values, the reconstructed results of all of the decomposition techniques are combined with a linear combiner. The main advantage of our approach is that the valuable characteristics of several decomposition techniques are combined.

The experimental results lead to the following conclusions:

1. The proposed EDNNRW\textsubscript{ELM} and EDNNRW\textsubscript{SSN} have good average ranks and were significantly superior to the other decomposition-based ELM methods and single models with a confidence of 95%. This indicates that the heterogeneous combination of different decomposition-based models can improve the forecasting capability of the proposed model.

2. When comparing both the EDNNRW\textsubscript{RVFL} and EDNNRW\textsubscript{RVFL*} with the EDNNRW\textsubscript{ELM} and EDNNRW\textsubscript{SSN}, the forecasting accuracies of the former methods were higher than the latter approaches. The FLN family approaches (RVFL and RVFL*) generated greater forecasting accuracy for the developed decomposition-based hybrid framework.
### Table 9: Improvement percentages of the RMSE results of EDNNWRVFL over the other competitors.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>n-step</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ELM</strong></td>
<td>1-step</td>
<td>99.34</td>
<td>90.19</td>
<td>96.10</td>
<td>98.63</td>
<td>96.91</td>
<td>96.93</td>
<td>97.38</td>
<td>96.99</td>
<td>97.20</td>
<td>96.79</td>
<td>98.55</td>
<td>98.38</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
</tbody>
</table>

### Table 10: Improvement percentages of the MAE results of EDNNWRVFL over the other competitors.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>n-step</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ELM</strong></td>
<td>1-step</td>
<td>99.34</td>
<td>90.19</td>
<td>96.10</td>
<td>98.63</td>
<td>96.91</td>
<td>96.93</td>
<td>97.38</td>
<td>96.99</td>
<td>97.20</td>
<td>96.79</td>
<td>98.55</td>
<td>98.38</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
<tr>
<td></td>
<td>5-step</td>
<td>92.57</td>
<td>84.65</td>
<td>92.85</td>
<td>92.68</td>
<td>92.65</td>
<td>92.59</td>
<td>92.52</td>
<td>92.53</td>
<td>92.49</td>
<td>92.52</td>
<td>92.49</td>
<td>92.42</td>
</tr>
</tbody>
</table>
3. The proposed EDNNRWvRF and EDNNRWvRF* ranked higher and significantly outperformed the comparative algorithms with a 0.05 significance level.

Future research directions and further possible improvements to this work include:

1. The type of prediction model selected has a significant influence on the predictive performance. Thus, the reservoir computing model and other available algorithms should be further investigated.

2. The number of lag orders and structure size of NNRW within the proposed method are user-specified parameters. Therefore, further work on how to automatically determine the optimal lag orders and node sizes is worth further investigation.

ACKNOWLEDGMENT

This work was supported by the Graduate Education of Computer and Information Science Interdisciplinary Research Grant from Department of Computer Science, Faculty of Science, Khon Kaen University #002/2556.

References


Pakarat Musikawan received his B.S. degree in information technology from Ubon Ratchathani University, Thailand. He received his M.S. and Ph.D. degrees in computer science from Khon Kaen University, Thailand. His research interests include machine learning, artificial intelligence, computational intelligence, soft computing, data mining, and knowledge-based systems.

Khamron Sunat received his B.S. degree in chemical engineering from Chulalongkorn University, Thailand, in 1989. He received his M.S. degree in computational science in 1998 and his Ph.D. degree in computer science from Chulalongkorn University in 2004. He now works as a lecturer in the Department of Computer Science at Khon Kaen University, Thailand. His research interests are in neural networks, pattern recognition, computer vision, soft computing, fuzzy systems, evolutionary computing and, optical computing.

Yanika Kongsorot received her B.S., M.S. and Ph.D. degrees in computer science from Khon Kaen University, Thailand. Her research interests include machine learning, pattern recognition, data mining, intelligent information processing, and decision support systems.