IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 February 2024, accepted 2 April 2024, date of publication 5 April 2024, date of current version 15 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3385643

APPLIED RESEARCH

Multi-Step Short-Term Wind Power Prediction Model Based on CEEMD and Improved Snake Optimization Algorithm

MENGLING ZHAO AND XUAN ZHOU
College of Science, Xi'an University of Science and Technology, Xi'an 710054, China
Corresponding author: Xuan Zhou (zx1999_05@163.com)

This work was supported by the National Social Science Fund of China under Grant 18XZW021.

ABSTRACT To effectively mitigate and address the impact of wind power uncertainty on the efficient operation of the power grid, this study proposes a novel multi-step short-term wind power prediction model based on complementary ensemble empirical modal decomposition (CEEMD), Improved Snake Optimization Algorithm (ISCASO), and Kernel Extreme Learning Machine (KELM). Firstly, the non-smooth wind power data are decomposed into a series of relatively smoother components using CEEMD to mitigate the complexity and instability of the original data. Subsequently, an improved snake optimization algorithm is introduced to optimize the KELM parameters, thereby establishing the prediction model of CEEMD-ISCASO-KELM for each stationary component and residual. Finally, by superimposing the prediction results of each component and residual, we obtain the final wind power prediction model. The simulation results show that, in comparison with existing prediction models, the proposed model in this study exhibits exceptional capability in accurately forecasting short-term wind power trends.

INDEX TERMS Improved snake optimization, kernel extreme learning machine, multi-step prediction, short-term wind power prediction.

NOMENCLATURE

ARMA	Autoregressive Moving Average Model.
ARIMA	Autoregressive Integral Moving Average Model.
ASO	Atom Search Optimization.
BBO	Biogeography-based optimization.
BP	Back Propagation.
CEEMD	Complementary EEMD.
CNN	Convolutional Neural Networks.
EMD	Empirical Mode Decomposition.
EEMD	Ensemble EMD.
ELM	Extreme Learning Machine.
GWO	Grey Wolf Optimization.
HPO	Hunter-Prey Optimization Algorithm.
IMF	Intrinsic Mode Function.
IPDL	Interval Probability Distribution Learning.
KELM	Kernel Extreme Learning Machine.
KF	Kalman filter.
NWP	Numerical Weather Prediction.
PSO	Particle Swarm Optimization.
RAE	Rough Autoencoder.
SCA	Sine Cosine Algorithm.
SO	Snake Optimization.
SOA	Seagull Optimization Algorithm.
SVM	Support Vector Machine.
SVR	Support Vector Regression.
VMD	Variable Mode Decomposition.
WOA	Whale Optimization Algorithm.
WPD	Wavelet Packet Decomposition.
WT	Wavelet Transform.
XGboost	eXtreme Gradient Boosting.

I. INTRODUCTION

In recent years, with the introduction of the carbon peak carbon neutral target, the development and utilization of new

The associate editor coordinating the review of this manuscript and approving it for publication was Riccardo Carotenuto.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

energy sources have emerged in order to adjust the energy structure and reduce environmental pollution. Wind energy is an active and widely distributed clean and renewable energy source with significant utilization value, which has now become an important source of energy generation in countries around the world. To fully exploit wind power resources, the implementation of a wind power prediction system assumes paramount importance for optimizing scheduling, ensuring system stability, and controlling operational costs within the power grid [1]. However, the volatile and intermittent nature of wind power sequences poses significant challenges to wind power utilization, thereby impacting the reliability and stability of wind farm power system operation [2]. Therefore, to realize the large-scale application of wind power, it is essential to improve the accuracy and effectiveness of wind power prediction.

A review of previous studies indicates that numerous methods have been proposed and refined by researchers to improve the precision of wind power prediction. These approaches can be broadly classified into three categories: physical models, statistical prediction models, and hybrid models. Physical models are commonly used for medium-term and long-term forecasting, incorporating factors such as terrain and guided by Numerical Weather Prediction (NWP), to compute wind speed and direction at the turbine's hub height, and then predict the wind power according to the characteristics of the wind turbine generator [3]. Currently, a plethora of studies are dedicated to wind power prediction using physical models. For instance, literature [4] proposes a sequence transfer correction algorithm for NWP wind speed, which combines the measured wind speed data to establish a clear mapping relationship between the NWP error and the predicted wind speed, and at the same time proves the effectiveness of the algorithm. Literature [5] proposes a novel method based on the double clustering division of the transitional weather process, and at the same time establishes various scenarios using multiple models to compare wind power prediction effects, which effectively improves the effect of ultra-short-term NWP wind speed revision. Literature [6] utilized multivariate and space-time information to jointly correct the wind speed of NWP, which has a significant advantage in the multivariate prediction accuracy of the wind speed of NWP to meet the real-time requirement. In general, this physical modeling approach relies on the physical properties of the model itself and meteorological expertise in wind power prediction, and is not applicable to short-term wind power prediction due to the computational complexity and the long training time of the model [7]. Therefore, statistical methods are introduced in short-term wind power forecasting for prediction, and the commonly used statistical models are Kalman Filter (KF) [8], Autoregressive Moving Average Model (ARMA) [9], and Autoregressive Integral Moving Average Model (ARIMA) [10]. For example, literature [10] takes relevant electricity data from Ukraine as an example and combines ARIMA and Long Short Term Memory to characterize the residuals to improve the prediction accuracy. Moreover, considering the non-stationary and non-linear characteristics of wind power data, prediction methods such as eXtreme Gradient Boosting (XGboost) [11], Convolutional Neural Network (CNN) [12], Support Vector Machine (SVM) [13], and Extreme Learning Machine (ELM) [14] have been widely adopted in short-term wind power due to their high prediction accuracy and outstanding flexibility. Literature [15] proposed an improved XGboost algorithm based on a Bayesian optimization hyperparameter with higher wind power prediction accuracy under extreme weather conditions and low wind speed ranges. Meanwhile Support Vector Regression (SVR), as an application branch of SVM, plays an important role in the field of wind power prediction [16]. For example, literature [17] proposed an improved genetic algorithm to optimize the hyperparameters of SVR, and the results show that the proposed model can effectively predict wind power. To address the uncertainty of wind power prediction models, researchers have developed various deep learning models, such as Rough Autoencoder (RAE) [18] and Interval Probability Distribution Learning (IPDL) [19], etc. Literature [20] proposes an adaptive graphical neural network prediction model based on spatio-temporal attention computation, which effectively predicts short-term wind power. Compared with the above prediction methods, Extreme Learning Machine (ELM) is a feed-forward neural network that exhibits rapid learning speed and robust generalization ability and has demonstrated promising results across various domains. It employs random weight generation for the input layer and hidden layer thresholds, requiring only the specification of hidden layer nodes. And literature [21] uses an improved Hunter-Prey Optimization Algorithm (HPO) to optimize the weights and thresholds of the ELM, and uses the wind turbine in Inner Mongolia, China to measure data to prove its effectiveness. Literature [22] optimized ELM using an improved Atom Search Optimization (ASO) algorithm and the results proved the good performance of the model. In addition, to enhance the generalization performance of ELM, Kernel Extreme Learning Machine (KELM) incorporates penalty coefficients for model optimization in addition to ELM and applies kernel functions to replace nodes in the implicit layer. Literature [23] combines the improved Universal Tabu Search Algorithm and KELM for wind power prediction, resulting in improved prediction performance.

Numerous wind power prediction methods have been proposed by researchers for physical and statistical models, leading to improved accuracy to some extent. However, due to the nonlinearity and non-smoothness of wind power time series, direct prediction using raw data often yields unsatisfactory results, so effective preprocessing of the data is carried out before the prediction to reduce noise interference and achieve smoother output results. Therefore, a hybrid approach combining data preprocessing methods with prediction models has emerged as the prevailing method for wind power prediction. Currently, widely used data preprocessing methods include Wavelet Transform (WT) [24], Variable Mode Decomposition (VMD) [25], Empirical Mode Decomposition (EMD) [26], Ensemble Empirical Mode Decomposition (EEMD) [27], and Complementary Ensemble Empirical Mode Decomposition (CEEMD) [28]. For example, literature [29] proposes an improved artificial bee colony algorithm based on Wavelet Transform (WT) combined with a kernel-limit learning machine, and experiments prove that the model can effectively improve the prediction accuracy of short-term wind power. Literature [30] proposes a new three-level decomposition method by using Wavelet Packet Decomposition (WPD) as the first level of decomposition, and then the VMD decomposition is performed on the obtained sub-layers, and the decomposed components are utilized for wind power prediction. Considering that the selection of wavelet bases significantly influences the outcomes of wavelet analysis, on this basis Huang and Shen [31] proposed EMD, which performs signal decomposition through the time scale characteristics of the data itself without the need to pre-set the basis function, and achieves a better result in the decomposition of the data. EMD is widely used in the analysis and preprocessing of time series prediction models, which decomposes a non-smooth signal into a finite number of Intrinsic Mode Function (IMF) and residuals step by step. Literature [32] decomposes the wind power series based on EMD data decomposition method, resulting in reduced prediction error and shorter model training time. However, the presence of intermittent signals in the actual signal introduces modal aliasing problems to the decomposition results obtained using the EMD method, thereby leading to inaccurate final predictions. To address the problems of EMD, EEMD [33] was proposed by utilizing noise-assisted processing. Nevertheless, the residual white noise added in the EEMD is not completely neutralized, which leads to the residual white noise being retained in the IMF components, To address this problem, CEEMD [34] incorporates adaptive white noise to further alleviate modal aliasing and minimize the number of iterations. Literature [35] utilized CEEMD to decompose air quality indicators in four cities to accurately predict air pollution.

Based on the above analysis, the hybrid prediction method has become the most widely used and successful method in short-term wind power prediction. Therefore, for the instability of short-term wind power, the application of data preprocessing techniques has gradually become a trend. The rapid learning speed and strong generalization capability of KELM can realize the fast prediction of short-term wind power. However, some scholars argue that the human-set parameters of KELM have a great influence on the prediction, so to overcome the sensitivity of KELM to parameter selection, a large number of literatures use a variety of optimization algorithms to find the optimal KELM parameters. Examples include Grey Wolf Optimization (GWO) for Optimizing KELM [36], Biogeography-based Optimization (BBO) for Optimizing KELM [37], Seagull Optimization Algorithm (SOA) for Optimizing KELM [38], and Snake Optimization (SO) for Optimizing KELM [39]. Among them, in literature [39], the snake optimization algorithm demonstrates superior global search capability and fast convergence speed, enabling it to efficiently and accurately determine the optimal solution for KELM parameters. Therefore, to obtain the optimal parameters of KELM and improve the prediction accuracy of short-term wind power, this paper proposes an improved snake optimization algorithm (ISCASO) to solve the problem of KELM stochastic parameters. Aiming at the problem that the SO algorithm itself has poor global search ability in the early stage and easily falls into the local optimum in the later stage, we incorporate chaotic mapping, dynamic inertia weights [40], and an enhanced positive cosine algorithm [41] [42] for updating the global and local search processes of the SO algorithm, respectively. The ISCASO algorithm is enhanced by implementing the aforementioned improvement measures to enable random parameter search. As a result, the proposed algorithm exhibits superior convergence performance and yields more satisfactory convergence results when compared to existing well-known heuristic algorithms. Meanwhile, combined with the analysis of the above literature, this paper proposes a multi-step prediction model CEEMD-ISCASO-KELM for short-term wind power forecasting. In comparison to the conventional statistical prediction model KELM, the model proposed in this study exhibits superior predictive accuracy and enhanced generalization capability.

The main contributions of this study can be summarized as follows in comparison to the current research: (1) the ISCASO optimization algorithm is proposed, which improves the global search capability of SO and effectively escapes local optima by incorporating Tent chaotic mapping, dynamic inertia weights, and an improved positive cosine algorithm; (2) The ISCASO algorithm is employed to optimize the regularization coefficients C and kernel parameters σ of the KELM model, addressing the issue of random parameter influence on prediction accuracy in the KELM model; (3) The multi-step prediction model named CEEMD-ISCASO-KELM with high accuracy and stability is proposed. By utilizing the CEEMD signal decomposition method for short-term wind power data, this model exhibits robust anti-interference capabilities. Consequently, it effectively mitigates randomness and instability in the data, resulting in more stable and accurate wind power predictions. The proposed model is validated through simulation, ablation experiment, sensitivity analysis, computational time complexity, and other aspects; (4) The wind power datasets of two different wind farms are used and compared with ten related models, and the superiority of the proposed prediction method is verified in different validation indexes.

The remaining sections of the paper are structured as follows: section II presents the theoretical background of CEEMD and KELM. Section III details the optimization strategy and algorithm evaluation experiments for ISCASO proposed in this study. Section IV focuses on discussing the model structures of ISCASO-KELM and CEEMD-ISCASO-KELM. In section V, we validate the effectiveness of the proposed multi-step prediction model by comparing and analyzing the experimental results. Finally, in section VI, we conduct an analysis and summary of this article.

II. METHODOLOGY

A. CEEMD

Empirical Mode Decomposition (EMD) is an adaptive signal time-frequency processing technique that effectively addresses the lack of adaptability in basis functions. It enables the decomposition of non-smooth, non-linear time series and those with high noise ratios into a set of smooth series with varying frequencies [43]. However, there is the problem of modal mixing in EMD, which does not explain the significance of individual IMFs well. Consequently, various approaches have been proposed to address this limitation, such as ensemble empirical modal decomposition (EEMD) [33] and complementary ensemble empirical modal decomposition (CEEMD) [34] etc. Although EEMD solves the problem of modal mixing in EMD, it not only increases the computational volume due to the incorporation of white noise but also suffers from substantial reconstruction errors and incomplete decomposition. The CEEMD method improves upon EEMD by incorporating a set of white noise with opposite signs into the original sequence, thereby significantly enhancing overall computational efficiency and effectively eliminating residual auxiliary noise during signal reconstruction. Assuming the given data sequence is $T(t)$, the specific calculation steps are as follows.

Firstly, a set of Gaussian white noise with opposite signs $(\varepsilon_{nj}^{j+}, \varepsilon_{nj}^{j-})$, $j = 1, 2, \ldots, I$ is introduced to the original sequence $T(t)$ in order to generate a new set of time series.

$$\begin{cases} T_n^{j+} = T + \varepsilon_{nj}^{j+}, & j = 1, 2, \ldots, I \\ T_n^{j-} = T + \varepsilon_{nj}^{j-}, & j = 1, 2, \ldots, I \end{cases} \tag{1}$$

Subsequently, the EMD method is applied to both T_n^{j+} and T_n^{j-} in order to obtain m pairs of IMFs.

$$\begin{cases} T_n^{j+} = \sum_{i=1}^{m} M_{ni}^{j+} \\ T_n^{j-} = \sum_{i=1}^{m} M_{ni}^{j-} \end{cases} \tag{2}$$

where, T_n^{j+} is the nth positive noise sequence, T_n^{j-} is the nth negative noise sequence, and M_{ni}^{j+} and M_{ni}^{j-} is the IMF sequence component obtained by EMD. Next, the final modal component IMF is obtained by calculating the average of all characteristic modal functions, where the nth IMF is defined as follows.

$$IMF_i = \frac{1}{2I} \sum_{j=1}^{I} \left[M_{ni}^{j+} + M_{ni}^{j-} \right] \tag{3}$$

where, IMF_i is the ith characteristic mode component after the decomposition above. Finally, the decomposition of CEEMD is shown as follows:

$$T(t) = \sum_{i=1}^{I} IMF_i + r_i(t) \tag{4}$$

$r_i(t)$ represents the residual of the ith iteration.

B. KERNEL EXTREME LEARNING MACHINE

The Extreme Learning Machine (ELM) [44] is a feedforward neural network-based machine learning algorithm. Building upon ELM, Huang et al. [45] combined different kernel functions to propose an improved extreme learning machine, namely the Kernel Extreme Learning Machine (KELM). Compared with ELM, the KELM method employs kernel mapping instead of random mapping, which has stronger network output stability and generalization ability, leading to significant improvements in learning speed and accuracy.

The specific KELM network structure diagram is shown in Fig. 1, where x is the input variable, y represents the output variable, ω_{ij} and β_{ij} is the connection weight, and O_l is the randomly generated threshold of the hidden layer. Assuming that the output matrix of the hidden layer is H, the regularization coefficient and kernel function are incorporated into the ELM framework to optimize both the output variables y and the weights of the output layer β.

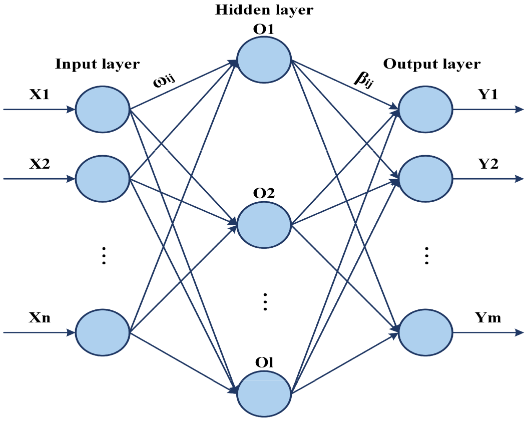

FIGURE 1. Structure diagram of the KELM model.

The kernel matrix in the KELM is as follows.

$$\begin{cases} \Omega_{ELM} = HH^T \\ \Omega_{ELM}(i,j) = h(x_i) \cdot h(x_j) = K(x_i, x_j) \end{cases} \tag{5}$$

where x_i and x_j are the sample input values, $K(x_i, x_j)$ is the kernel function. The RBF kernel function is employed in the present study as follows.

$$K(x_i, x_j) = \exp(-\gamma \left\| x_i - x_j \right\|^2) \tag{6}$$

Equation (6), γ is the kernel parameters, the output function $y(x)$ of the KELM and the connection weights β of the output layer are described as follows.

$$\begin{cases} y(x) = \begin{bmatrix} K(x, x_1) \\ \vdots \\ K(x, x_N) \end{bmatrix}^T (I/C + \Omega_{ELM})^{-1} T \\ \beta = ((I/C + \Omega_{ELM})^{-1} T) \end{cases} \tag{7}$$

C represents the regularization coefficient, which is used to weight the output weights and training errors, Ω_{ELM} denotes the kernel function matrix, $K(x_i, x_j)$ denotes the kernel function, I is the unit matrix, and T denotes the expected output.

III. THE PROPOSED ISCASO ALGORITHM

A. SNAKE OPTIMIZATION ALGORITHM

Snake Optimization Algorithm [46] is a meta-heuristic optimization algorithm proposed by Fatma A. Hashim and Abdelazim G. Hussien in 2022, which draws inspiration from the foraging and mating behavior of snakes. Based on the effect of ambient temperature and food availability on the mating behavior of snakes, snakes search for food or eat the available food when temperatures are high or insufficient food is present, while snakes fight with each other as well as mate and hatch new individuals during periods of low temperature and abundant food resources. Consequently, the algorithm divides the search process into two distinct phases: exploration and exploitation, which are more complex in the exploitation phase and can be subdivided into three stages based on temperature conditions and food availability: predation stage, struggle stage, and mating stage.

The SO algorithm, similar to other meta-heuristic algorithms, initiates the process by generating a uniformly distributed random initial population using the following equation.

$$X_\text{i} = X_{\min} + r \times (X_{\max} - X_{\min}) \tag{8}$$

where, X_i is the position of the $i$ith individual, r is the random number between 0 and 1, $X_{\min}$ and $X_{\max}$ are the upper and lower bounds of the algorithm, respectively.After a random initialization of the search space, the snake population is partitioned into males and females, with males accounting for 50% and females for 50%. The distribution of males and females is represented as follows.

$$N_m = \frac{N}{2} \tag{9}$$

$$N_f = N - N_m \tag{10}$$

N is the total population size, N_m denotes the number of male individuals, and N_f denotes the number of female individuals. Obtain the most superior male individual ($f_{best,m}$), female individual ($f_{best,f}$), and food location (f_{food}) from each group while precisely defining temperature and food quantity.

$$Temp = \exp(\frac{-t}{T}) \tag{11}$$

$$Q = c_1 \times \exp(\frac{t - T}{T}) \tag{12}$$

where, t indicates the current iteration and T indicates the maximum number of iterations. c is 0.5.

During the exploration phase characterized by low temperature and absence of food, i.e. $Q < Threshold(Threshold = 0.25)$, the snake searches for food by selecting any random location and updates the snake's location as follows.

$$X_{i,m}(t+1) = X_{rand,m}(t) \pm c_2 \times A_m \times ((X_{\max} - X_{\min}) \times rand + X_{\min}) \tag{13}$$

where, $X_{i,m}$ denotes the position of the ith male, $X_{rand,m}$ denotes the position of the random male, and A_m is the ability to find food for the male, calculated as follows.

$$A_m = \exp(\frac{-f_{rand,m}}{f_{i,m}}) \tag{14}$$

$f_{rand,m}$ is the fitness of $X_{rand,m}, f_{i,m}$ is the fitness of the ith male, and c_2 is 0.05.

$$X_{i,f}(t+1) = X_{rand,f}(t) \pm c_2 \times A_f \times ((X_{\max} - X_{\min}) \times rand + X_{\min}) \tag{15}$$

where, $X_{i,f}$ denotes the position of the ith male, $X_{rand,f}$ denotes the position of the random female, and A_f is the ability to find food for the female, calculated as follows.

$$A_f = \exp(\frac{-f_{rand,f}}{f_{i,f}}) \tag{16}$$

During the exploitation phase characterized by abundant food availability, i.e. $Q > Threshold(Threshold = 0.25)$, when Temp>0.6 and the temperature is high, the snake population exclusively exhibits movement towards food, and the position update equation is as follows.

$$X_{i,j}(t+1) = X_{food} \pm c_3 \times Temp \times rand \times (X_{food} - X_{i,j}(t)) \tag{17}$$

where, $X_{i,j}(t)$ is the position of the male or female individual, X_{food} is the position of the optimal individual, and c_3 is 2. When the ambient temperature drops below 0.6, the snake enters either a struggle or mating mode. The position update formula during struggle mode is as follows.

$$X_{i,m}(t+1) = X_{i,m}(t) + c_3 \times FM \times rand \times (X_{best,f} \times Q - X_{i,m}(t)) \tag{18}$$

$X_{i,m}(t)$ is the position of the ith male, $X_{best,f}$ is the position of the optimal female, and FM denotes the fighting ability of the male.

$$X_{i,f}(t+1) = X_{i,f}(t) + c_3 \times FF \times rand \times (X_{best,m} \times Q - X_{i,f}(t)) \tag{19}$$

$X_{i,f}(t)$ is the position of the ith female, $X_{best,m}$ is the position of the optimal male, and FF denotes the fighting ability of the female. The FM and FF are provided below.

$$FM = \exp(\frac{-f_{best,f}}{f_i}) \tag{20}$$

$$FF = \exp(\frac{-f_{best,m}}{f_i}) \tag{21}$$

$f_{best,m}$ and $f_{best,f}$ are the optimal fitness in the male and female populations, respectively, and f_i is the fitness of the individual snake.The formula for position update in mating mode is as follows.

$$\begin{aligned} X_{i,m}(t+1) &= X_{i,m}(t) + c_3 \times M_m \times rand \\ &\quad \times (X_{i,f}(t) \times Q - X_{i,m}(t)) \end{aligned} \tag{22}$$

$$\begin{aligned} X_{i,f}(t+1) &= X_{i,f}(t) + c_3 \times M_f \times rand \\ &\quad \times (X_{i,m}(t) \times Q - X_{i,f}(t)) \end{aligned} \tag{23}$$

The mating ability of males and females, denoted as M_m and M_f respectively, is computed according to the following equations.

$$M_m = \exp(\frac{-f_{i,f}}{f_{i,m}}) \tag{24}$$

$$M_f = \exp(\frac{-f_{i,m}}{f_{i,f}}) \tag{25}$$

If the snake eggs hatch, the worst individual males and females are selected for replacement.

$$X_{worst,m} = X_{\min} + rand \times (X_{\max} - X_{\min}) \tag{26}$$

$$X_{worst,f} = X_{\min} + rand \times (X_{\max} - X_{\min}) \tag{27}$$

where, $X_{worst,m}$ and $X_{worst,f}$ denote the worst individuals in males and females, respectively.

B. IMPROVED SNAKE OPTIMIZATION ALGORITHM

The snake optimization algorithm is optimized in three aspects in this study. Firstly, the initialization of the snake population utilizes the Tent chaotic mapping method to enhance randomness and diversity, thereby reducing uncertainty during population initialization. Secondly, an improved inertia weight factor is employed during the global exploration stage to address problems related to slower global search and poorer convergence, thus enhancing the algorithm's global search capability. Finally, an enhanced positive cosine algorithm is introduced during the struggle phase to mitigate the problem of falling into local optimality, resulting in improved global search capability.

1) TENT CHAOS MAPPING

Chaotic sequences possess characteristics of randomness, ergodicity, and regularity. Utilizing these characteristics in optimization search can effectively maintain population diversity, prevent the algorithm from converging to local optima, and enhance its global search capability [47]. Currently, chaotic mapping has been extensively incorporated into swarm intelligence algorithms. Different chaotic mappings exert varying degrees of influence on the optimization process. Two commonly employed chaotic functions are the Tent function and the Logistic function [48]. The Tent chaotic mapping exhibits characteristics such as a simple structure, uniform distribution of population after mapping, high traversal ability, and computational efficiency. Its traversal effect surpasses that of the Logistic function by generating an initial population for optimization algorithms more effectively. Tent chaotic mapping is a kind of segmented mapping with the formula:

$$x_{n+1} = \begin{cases} \dfrac{x_n}{\alpha} & x_n \in (0, \alpha] \\ \dfrac{(1-x_n)}{(1-\alpha)} & x_n \in (\alpha, 1] \end{cases} \tag{28}$$

where, α is a chaos parameter with a variation range of (0, 1], in particular, when α is set to 0.4999, the system presents a chaotic state. This can be observed in Fig. 2(a), which shows the scatter plot of the distribution of Tent chaotic sequences uniformly distributed throughout the entire target region. Additionally, Fig. 2(b) presents a histogram depicting the relatively uniform distribution of values taken by the Tent mapping within each interval for the Tent chaotic sequences. Consequently, incorporating the Tent chaotic mapping into the snake optimization algorithm enhances population diversity and improves algorithm accuracy.

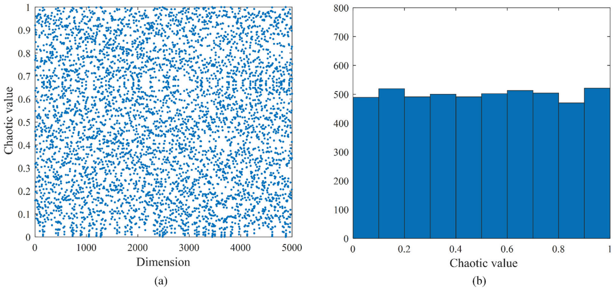

FIGURE 2. **Distribution of tent chaotic mapping.**

2) DYNAMIC INERTIA WEIGHTS

As a crucial parameter in the optimization algorithm, the inertia weight factor plays a significant role in optimizing the objective function. A larger inertia weight factor enhances global search ability, while a smaller one strengthens individual's local search ability [49] [50]. To address the issue of slower search speed and poorer convergence during the exploration phase of the snake optimization algorithm, this study combines the dynamic inertia weight factor proposed by Qing et al. [41] in the particle swarm optimization (PSO) algorithm with an improved position updating formula. This combination aims to enhance global search ability, prevent premature convergence to local optima, and improve overall optimization accuracy. The detailed formulation is described as follows.

$$\begin{aligned} w_1 &= w_{\min} + (\frac{w_{\max} - w_{\min}}{2}) \\ &\quad \times \frac{1}{\ln(e + (t/T)^2)} + (\frac{w_{\max} - w_{\min}}{2}) \times rand \end{aligned} \tag{29}$$

where $\omega_{\max} = 0.9$, $\omega_{\min} = 0.2$, as the parameter t increases, $1/\ln(e + (t/T)^2)$ gradually decreases, resulting in the inertia weight is large in the early stage and decreases gradually in the later stage. By dynamically adjusting the weights over time, it not only ensures a non-linear reduction of inertia weights but also satisfies the necessary conditions for convergence. Moreover, incorporating *rand*() enhances the algorithm's early-stage search capability and prevents premature convergence in later stages, thereby enhancing the algorithm's search accuracy.

The position update formula of the SO algorithm in the exploration stage, incorporating a dynamic inertia weight factor, is expressed as follows:

$$X_{i,m}(t+1) = w_1 \times X_{rand,m}(t) \pm c_2 \times A_m \times ((X_{\max} - X_{\min}) \times rand + X_{\min}) \quad (30)$$

$$X_{i,f}(t+1) = w_1 \times X_{rand,f}(t) \pm c_2 \times A_f \times ((X_{\max} - X_{\min}) \times rand + X_{\min}) \quad (31)$$

3) IMPROVED SINE-COSINE ALGORITHM

Inspired by the trigonometric function, the Sine Cosine Algorithm (SCA) was proposed by Mirjalili et al. [51] in 2016. It optimizes both global and local areas based on oscillation changes of the sine and cosine model to achieve global optimization. The location update formula of the SCA algorithm is as follows.

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times \sin(r_2) \times \left| r_3 \times P_i^t - X_i^t \right|, & r_4 < 0.5 \\ X_i^t + r_1 \times \cos(r_2) \times \left| r_3 \times P_i^t - X_i^t \right|, & r_4 > 0.5 \end{cases} \quad (32)$$

where, X_i^t is the current individual, P_i^t is the current optimal individual. $r_2 \in [0, 2\pi]$, $r_3 \in [0, 2]$, $r_4 \in [0, 1]$, which are all random numbers.

$$r_1 = a - t\frac{a}{T} \quad (33)$$

a represents a fixed value, t denotes the current number of iterations, and T signifies the maximum number of iterations. As the original step size search factor r_1 follows a monotone linear function, this algorithm facilitates global search when r_1 is high and local search when r_1 is low. Therefore, the importance of r_1 in achieving the optimal solution necessitates the introduction of a hyperbolic sinusoidal step size search factor [42].

$$r_{\text{sinh}} = -\frac{T}{\lambda} \times \sinh(-\omega \times \frac{t^3}{T^3}) + \theta \quad (34)$$

$\lambda = 5$, $\omega = 0.01$, both serve as regulatory factors. θ is the displacement and the value is 1. The transformation curve of the hyperbolic sine step size search factor before and after the improvement is depicted in Fig. 3. It can be observed that during the early stage of iteration, the hyperbolic sine step size search factor exhibits a large value, and the decline rate is slow, thereby enhancing the algorithm's global search capability. Conversely, in the late stage of iteration, this factor assumes a smaller value, promoting the local development ability and accelerating optimization speed.Further, to make the current individual position information X_i^t be fully utilized with the increase in the number of iterations, adaptive parameters [43] are introduced into the position update formula as inertial weights. This enables a more comprehensive exploitation of the optimal solution and enhances the optimization accuracy of the algorithm. The specific enhancement formula is presented as follows.

$$\omega_2 = 1 - \frac{e^{t/T} - 1}{e - 1} \quad (35)$$

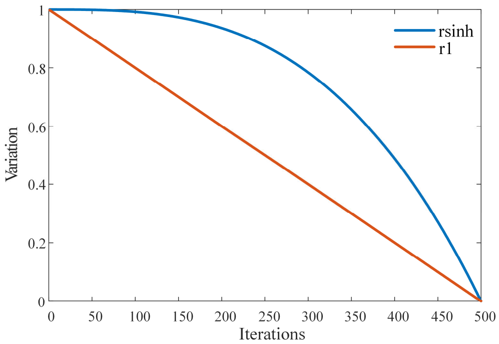

FIGURE 3. The curve for the variation in step size search factor.

In the traditional SO algorithm, the snake population enters into a struggle or mating phase during periods of food availability and lower temperature. While this struggle mode helps to select the most optimal individual by promoting competition for resources and mates, it also has certain drawbacks, i.e., the optimal position of a single individual directly determines the efficiency and accuracy of the whole algorithm to find the optimal position. Therefore, the strong dependence on an optimal individual can hinder global optimization efforts and lead to premature convergence in local optima [52]. Therefore, to further enhance the algorithm's performance under the struggling model, this study introduces the above improved SCA algorithm. By appropriately adjusting various parameters within the SCA algorithm, it enables SO to timely escape from local optima and prevent premature convergence. The update equations (18) (19) in Struggle mode are revised as (36) (37).

$$X_{i,m}(t+1) = \begin{cases} \omega_2 \times X_{i,m}(t) + r_{\text{sinh}} \times FM \times \sin(r_2) \\ \quad \times \left| r_3 \times X_{best,f} - X_{i,m}(t) \right|, r_4 < 0.5 \\ \omega_2 \times X_{i,m}(t) + r_{\text{sinh}} \times FM \times \cos(r_2) \\ \quad \times \left| r_3 \times X_{best,f} - X_{i,m}(t) \right|, r_4 \geq 0.5 \end{cases} \quad (36)$$

$$X_{i,f}(t+1) = \begin{cases} \omega_2 \times X_{i,f}(t) + r_{\text{sinh}} \times FM \times \sin(r_2) \\ \quad \times \left| r_3 \times X_{best,m} - X_{i,f}(t) \right|, r_4 < 0.5 \\ \omega_2 \times X_{i,f}(t) + r_{\text{sinh}} \times FM \times \cos(r_2) \\ \quad \times \left| r_3 \times X_{best,m} - X_{i,f}(t) \right|, r_4 \geq 0.5 \end{cases} \quad (37)$$

where r_1, r_2 and r_3 are the same as above.

Algorithm 1 Pseudo-code of the proposed ISCASO

```
Initialize Problem Setting ($Dim$, $Pop\_size(N)$, $Max\_Iter(T)$, $Curr\_Iter(t)$, $UB$, $LB$)
Initialize the position of the population based on the Tent chaotic mapping.
Divide the population $pop\_size$ to 2 equal groups $N_m$ and $N_f$ using Equations. (9) and (10).
While ($t \leq T$) do do
  Find the best individual
  Calculate Temp and food Quantity(Q) using Equations. (11) and (12)
  if (Q<0.25) then
    Perform exploration using Equations. (30) and (31)
  else if (Temp>0.6) then
     Perform exploitation using Equation. (17)
   else
     if (rand<0.6) then
       Snakes in Fight Mode Equations. (36) and (37)
     else
       Snakes in Mating Mode Equations. (22) and (23)
       Change the worst male and female Equations. (26) and (27)
     end if
   end if
 end while
Return best solution
```

4) ALGORITHM COMPLEXITY ANALYSIS

The complexity of an algorithm serves as a reflection of its operational speed and can also indicate the efficiency with which it operates. When multiple algorithms are available for solving a given problem, selecting the algorithm with the lowest time complexity becomes a crucial criterion in algorithm selection. The time complexity of initializing the population for SO is $O(n \times d \times T)$, where n denotes the population size, d denotes the problem dimension, and T denotes the maximum number of iterations. Compared with SO, ISCASO adds three components. Among them, replacing the original population initialization with Tent chaotic mapping introduces a time complexity of $O(n \times d)$, the introduction of dynamic inertia weights only modifies the update position of the population, thereby expanding the search range of the population and increasing its proximity to the optimal solution. This strategy does not result in an increase in computational complexity of $O(n \times d \times T)$. The time complexity of the improved SCA for positional updating under the struggle phase is $O(n \times d \times T) + O(n \times d \times T) = O(n \times d \times T)$. Since the algorithm complexity retains the highest order term, the time complexity of ISCASO is $O(n \times d \times T)$. Therefore, the time complexity of ISCASO and SO is equivalent, and the incorporation of multiple strategies does not introduce any additional algorithmic complexity to ISCASO.

By combining the above-proposed strategy with the original SO algorithm and improving some of the search links in the SO algorithm, the optimization seeking ability and efficiency of the original SO can be effectively enhanced. Algorithm 1 presents the pseudo-code for the improved ISCASO algorithm.

C. NUMERICAL SIMULATION

1) TEST FUNCTION

In order to verify the efficacy and reliability of the proposed ISCASO algorithm, we have selected four algorithms from recent years as the comparison algorithms in this study, including Snake Optimization Algorithm (SO), Sine Cosine Algorithm (SCA), Whale Optimization Algorithm (WOA) [53], and Gray Wolf Optimization Algorithm (GWO) [54]. The test set primarily comprises 12 benchmark test functions, wherein unimodal functions (F1-F6) which have only one global optimal solution and can be used to estimate the algorithm's exploitation performance, and multimodal functions (F7-F9), and fixed-dimension multimodal functions (F10-F12) contain a large number of local optimal solutions, and thus are usually used to test the algorithm's exploration ability and its ability to avoid local optima [55]. The details of the 12 benchmark test functions are given in Table 1.

2) PARAMETER SETTING

The test environment for this experiment is MATLAB2020b and the hardware platform configuration of the computer is Intel®Core™i5-10500 CPU @3.10GHz. For all the algorithms used in the experiments, the population size is set to 100, the maximum number of iterations is set to 500, and the parameters of the test function are given in Table 2. To reduce the randomness in the experiments, each experiment was independently conducted 30 times, and the mean and standard deviation were computed. The detailed results are shown in Table 3. The mean reflects the optimization accuracy of the algorithms, and the closer the average fitness approaches the theoretical optimal solution,

TABLE 1. Test functions.

Type	Name	Functions	Dim	Range	$f_{\min}$
Unimodal	Sphere Function	$F_1(x)=\sum_{i=1}^{n} x_i^2$	30	[-100,100]	0
	Schwefel's Problem 2.22	$F_2(x)=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	30	[-10,10]	0
	Schwefel's Problem 1.2	$F_3(x)=\sum_{i=1}^{n}(\sum_{j-1}^{i} x_j)^2$	30	[-100,100]	0
	Schwefel's Problem 2.21	$F_4(x)=\max_i\{\lvert x_i\rvert, 1\le i\le n\}$	30	[-100,100]	0
	Generalized Rosenbrock's Function	$F_5(x)=\sum_{i=1}^{n-1}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	30	[-30,30]	0
	Step Function	$F_6(x)=\sum_{i=1}^{n}([x_i+0.5])^2$	30	[-100,100]	0
Multimodal	Griewank Function	$F_7(x)=418.9829n-\sum_{i=1}^{n} x_i\sin(\sqrt{\lvert x_i\rvert})$	30	[-500,500]	0
	Generalized Rastrigin's Function	$F_8(x)=\sum_{i=1}^{30}[x_i^2-10\cos(2\pi x_i)+10]$	30	[-5.12, 5.12]	0
	Generalized Griewank's Function	$F_9(x)=\frac{1}{4000}\sum_{i=1}^{30} x_i^2-\prod_{i=1}^{30}\cos(\frac{x_i}{\sqrt{i}})+1$	30	[-600,600]	0
Fixed-dimension	Cross-in-Tray Function	$F_{10}(x)=-0.0001\left(\left\lvert\sin(x_1)\sin(x_2)\exp\left(\left\lvert 100-\frac{\sqrt{x_1^2+x_2^2}}{\pi}\right\rvert\right)\right\rvert+1\right)^{0.1}$	2	[-10,10]	-2.06261
	Drop-Wave Function	$F_{11}(x)=-\frac{1+\cos(12\sqrt{x_1^2+x_2^2})}{0.5(x_1^2+x_2^2)+2}$	2	[-5.12, 5.12]	-1
	Perm Function 0, D, BETA	$F_{12}(x)=\sum_{i=1}^{d}\left(\sum_{j=1}^{d}(j+\beta)(x_j^i-\frac{1}{j^i})\right)^2$	2	[-2,2]	0

TABLE 2. Related algorithm parameter settings.

Algorithm	Parameters
ISCASO	$w_{max}=0.9$, $w_{\min}=0.2$, $\lambda=5$, $\omega=0.01$, $\theta=1$
SO	a = 2(linearly decreases over iterations)
WOA	β=1, a = 2(linearly decreases over iterations)
SCA	a = 2(linearly decreases over iterations)
GWO	a = 2(linearly decreases over iterations), $r_1\in[0,1], r_2\in[0,1]$

the higher the algorithms' search accuracy. On the other hand, the standard deviation reflects the degree of deviation of the experimental data, the smaller the standard deviation, the higher the stability of the algorithm. The optimal evaluation indexes obtained from different algorithms are denoted in bold black font.

3) COMPARISON WITH OTHER ALGORITHMS

As can be seen from Table 3, the performance of the ISCASO algorithm proposed in this study has a more significant improvement compared to other swarm intelligence optimization algorithms. In the 12 benchmark functions, the mean and standard deviation of the ISCASO algorithm are better than the other four algorithms, especially in the multimodal function (F7-F9), the mean value of ISCASO reaches the optimal value of the test function, and the stability of the algorithm is much better than the other four algorithms. In unimodal functions (F1-F6), ISCASO significantly outperforms the other four algorithms even though it does not reach the optimal function value. Specifically, for the F5 test function, the mean and standard deviation of ISCASO are 3.95e+00 and 1.59e+00, respectively, which is a better performance in the same order of magnitude.

In fixed-dimension multimodal functions (F10-F12), the global optimum is achieved by all algorithms on F10, however, the ISCASO algorithm has the lowest standard deviation and the strongest robustness. The ISCASO algorithm has the same results as SCA and GWO on F11, achieving the optimal value of the function while maintaining algorithmic stability.

To demonstrate the advantages of ISCASO performance, Fig. 4 shows the convergence curves of the 12 tested functions. Among the five algorithms, the ISCASO algorithm proposed in this study performs the best in terms of optimization and can quickly converge to the final solution in the early stages of the algorithm. In unimodal test function F4, the accuracy performance of ISCASO is comparatively inferior to that of GWO, yet the algorithm has the fastest convergence speed. As can be seen from the iteration effect diagram, SO has a slower global search during the iteration process, rendering it susceptible to local optima and it is

TABLE 3. Comparison results between ISCASO and different algorithm.

Function	ISCASO		SO		WOA		SCA		GWO	
	avg	std	avg	std	avg	std	avg	std	avg	std
Unimodal										
F1	**3.25e-132**	**1.78e-131**	9.39e-117	3.18e-116	1.67e-85	9.14e-85	3.84e-14	7.58e-14	2.19e-69	7.18e-69
F2	**5.97e-67**	**3.09e-66**	3.92e-60	1.76e-59	1.14e-55	5.80e-55	2.20e-10	5.89e-10	2.42e-40	4.03e-40
F3	**6.77e-94**	**3.71e-93**	4.77e-83	1.70e-82	6.94e+01	9.26e+01	1.25e-04	4.74e-04	6.24e-33	1.15e-32
F4	**5.14e-60**	**2.70e-59**	6.66e-52	1.49e-51	1.31e+00	3.56e+00	1.78e-04	6.26e-04	2.16e-22	3.29e-22
F5	**3.95e+00**	**1.59e+00**	4.45e+00	4.07e+00	6.43e+00	4.21e-01	7.20e+00	3.73e-01	6.30e+00	6.11e-01
F6	**1.28e-09**	**3.34e-09**	4.84e-08	2.07e-07	1.39e-04	7.37e-05	3.36e-01	1.31e-01	2.81e-06	9.19e-07
Multimodal										
F7	**0.00e+00**	**0.00e+00**	2.47e-02	5.46e-02	5.30e-03	2.05e-02	8.13e-01	3.22e-01	1.890e-03	5.22e-03
F8	**0.00e+00**	**0.00e+00**	1.58e+00	2.62e+00	4.58e-01	2.51e+00	1.35e+00	4.17e+00	8.309e-01	1.99e+00
F9	**0.00e+00**	**0.00e+00**	6.65e-02	4.87e-02	6.94e-02	1.09e-01	1.69e-02	7.17e-02	1.685e-02	2.47e-02
Fixed-dimension Multimodal										
F10	**-2.06e+00**	**9.03e-16**	**-2.06e+00**	9.22e-16	**-2.06e+00**	3.31e-09	**-2.06e+00**	1.42e-05	**-2.06e+00**	7.84e-09
F11	**-1.00e+00**	**0.00e+00**	-9.88e-01	2.41e-02	-9.87e-01	2.59e-02	**-1.00e+00**	**0.00e+00**	**-1.00e+00**	**0.00e+00**
F12	**1.58e-31**	**3.21e-31**	4.19e-31	1.72e-30	1.59e-02	2.10e-02	3.82e-03	3.71e-03	2.70e-06	2.69e-06

difficult to find the optimal value. Whereas the ISCASO algorithm effectively improves the global search capability by introducing the dynamic inertia weight factor and the positive cosine algorithm, jumps out of the local optima, and expedites the accurate discovery of global optimal solutions.

IV. SHORT-TERM WIND POWER PREDICTION MODEL BASED ON CEEMD-ISCASO-KELM

Many existing short-term wind power prediction models suffer from problems such as slow sample training speed, susceptibility to local optima, and low generalization performance. Considering the advantages of ISCASO algorithm with high convergence and global search ability, KELM with high generalization and accuracy in short-term wind power prediction, and CEEMD with good signal decomposition capability. Therefore, based on the above methods, this study proposes a novel multi-step short-term wind power prediction model based on CEEMD, ISCASO, and KELM. The wind power signal is decomposed using CEEMD, and the ISCASO optimization algorithm is used to determine the parameters of the KELM model, so as to improve the accuracy of short-term wind power prediction.

A. *KELM MODEL OPTIMIZED BY ISCASO*

Hyperparameters are model parameters that need to be set artificially before the training of machine learning algorithms, and they cannot be determined through the training process, therefore, the determination of hyperparameters is crucial in the training of models. In the implementation of the KELM model, the selection of regularization coefficients and kernel parameters will directly affect the performance of the model [56]. Therefore, the ISCASO optimization algorithm is combined with KELM to optimize the regularization coefficient C and kernel parameter σ, which improves the prediction accuracy of the KELM model to a certain extent, and the detailed process of ISCASO-KELM is described as follows.

Step 1. The raw wind power sequences are divided into training and test sets and normalized.

Step 2. Initialize the ISCASO parameters and determine the snake population size N, the maximum number of iterations T, and the upper and lower bounds of the regularization coefficients and the kernel parameters.

Step 3. Initialize the KELM structure, input the location information of the initial individuals into the KELM model, and perform training and prediction on both the training set and testing set datasets, and also calculate the fitness function for each initial individual f. The fitness function f in this study is defined as the root mean square error (RMSE) of the KELM prediction.

$$f = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2} \tag{38}$$

where y_i is the true value of the ith sample and $\widehat{y}_i$ is the predicted value of the ith sample.

Step 4. The fitness function value of each individual is used to determine whether it has reached the individual's historical optimum. If so, the global optimal snake individual position and fitness value are updated accordingly. And the position of the best-fit snake is retained as C and σ of the KELM.

Step 5. Determine whether the algorithm satisfies the termination condition. If it does, exit the loop and output the prediction results; otherwise, continue with loop calculations.

Step 6. The wind power data utilized for training and testing are substituted into the optimized model, and the model is trained to output the prediction results.

Based on the above steps, the corresponding flowchart is shown in Fig. 5.

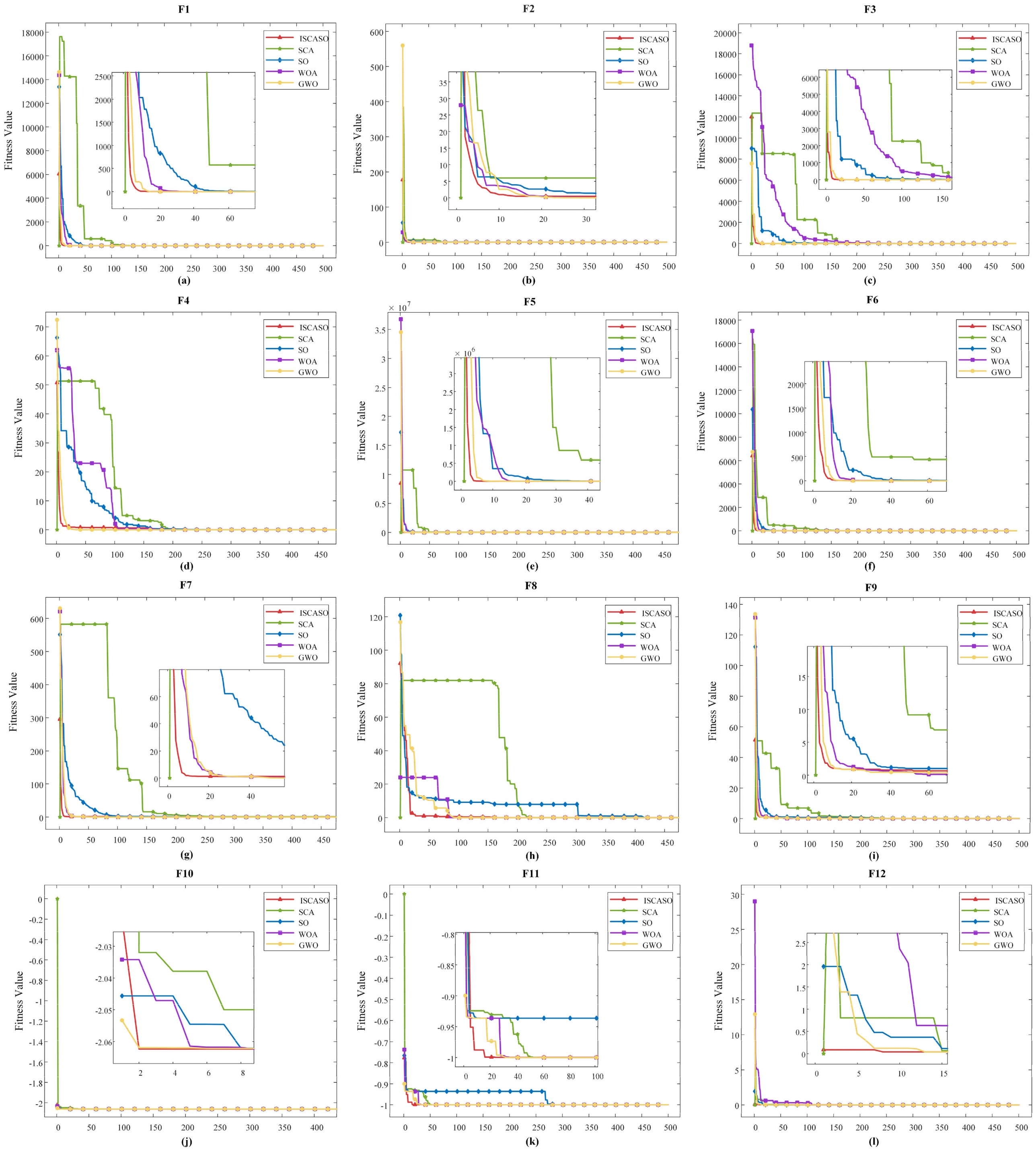

FIGURE 4. Convergence curves of competing algorithms for test functions.

B. MULTI-STEP SHORT-TERM WIND POWER PREDICTION MODEL

The wind power data is a class of time series signals with nonlinearity, non-stationarity, and stochasticity, and this data is commonly decomposed by suitable signal decomposition methods to improve the periodicity and stability of the data. The CEEMD integrates the characteristics of time series to complete the adaptive decomposition, while also addressing problems such as modal aliasing and sieve iteration stopping criterion that exist in the EMD and EEMD. ISCASO can adaptively optimize the main hyperparameters of KELM with good robustness and easy convergence. The present study proposes a novel multi-step wind power prediction model, namely CEEMD-ISCASO-KELM, which integrates

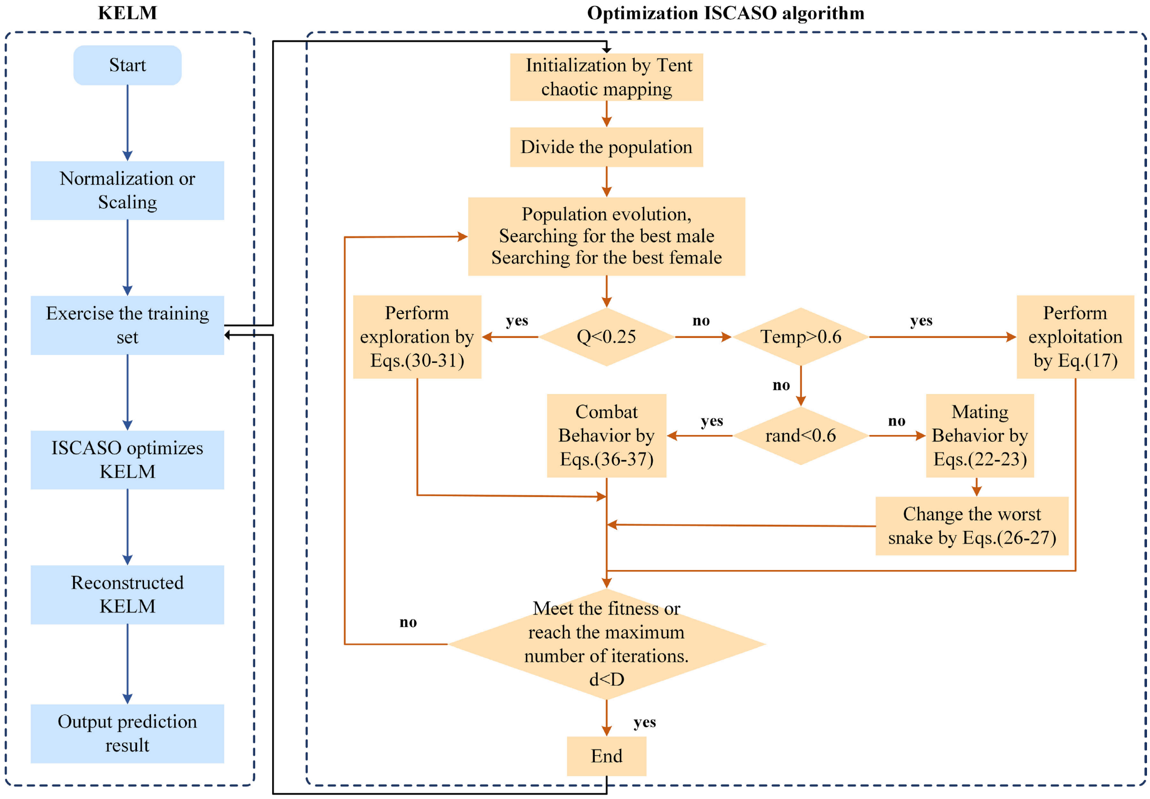
KELM
Start
Normalization or Scaling
Exercise the training set
ISCASO optimizes KELM
Reconstructed KELM
Output prediction result
Optimization ISCASO algorithm
Initialization by Tent chaotic mapping
Divide the population
Population evolution, Searching for the best male Searching for the best female
Perform exploration by Eqs.(30-31)
yes
Q<0.25
no
Temp>0.6
yes
Perform exploitation by Eq.(17)
no
Combat Behavior by Eqs.(36-37)
yes
rand<0.6
no
Mating Behavior by Eqs.(22-23)
Change the worst snake by Eqs.(26-27)
no
Meet the fitness or reach the maximum number of iterations. d<D
yes
End

FIGURE 5. Flowchart of ISCASO optimized KELM model.

the CEEMD and ISCASO-KELM models. The prediction steps of the CEEMD-ISCASO-KELM model are shown below.

Step 1. Collect raw wind power data.

Step 2. The CEEMD method is utilized to decompose the raw data into $IMF_1, IMF_2, \ldots, IMF_n$, and a residual $r(t)$, and the decomposed $n+1$ variables are normalized to avoid the effect of different scales.

Step 3. Different ISCASO-KELM models are constructed according to the IMFs in different frequency bands, divided into a training set and a testing set, and optimize the regularization coefficient C and kernel parameter σ in the KELM using ISCASO. The training is terminated when the ISCASO convergence condition is satisfied, and the KELM with optimal performance for each IMF is obtained.

Step 4. The IMF_1, IMF_2, $\ldots$, IMF_n, and a residual $r(t)$ were estimated using $n+1$ the ISCASO-KELM prediction models, respectively. And the predicted components are superimposed and summed to get the final short-term wind power prediction.

The flowchart of the CEEMD-ISCASO-KELM model is shown in Fig. 6.

V. EXPERIMENTAL DESIGN

A. DATE ACQUISITION AND PREPROCESSING

In this study, two wind farms (designated as NO.1 and NO.2) located in China were utilized to randomly select two sets of one-week actual wind power data with distinct fluctuation scenarios from 2019. Specifically, dataset A represents the wind power data collected from January 9th to January 15th, 2019 at wind measurement tower A in wind farm NO.1, while dataset B corresponds to the data obtained from July 24th to July 30th, 2019 at wind measurement tower B in wind farm NO.2. The two datasets represent two widely varying meteorological conditions, mainly including wind power, wind speed at 10 m, 30 m, 50 m, and 70 m; wind direction at 10 m, 30 m, 50 m, and 70 m; as well as wind speed at hub height, wind direction at hub height, temperature, pressure, humidity, etc. All experimental data points were sampled every 15-minute intervals resulting in a total of 672 data points. Fig. 7 illustrates the wind power of the two data groups, demonstrating distinct levels of randomness and volatility. Table 4 provides statistical descriptions of the wind power data, revealing that Dataset A exhibits a larger standard deviation and extreme deviation, indicating more violent fluctuations. On the other hand, Dataset B displays smaller standard deviation and extreme deviation, suggesting overall moderate fluctuations but with greater randomness. To evaluate the predictive performance of the model, the samples from the first 5 days in each dataset are utilized as the training dataset while the samples from the last 2 days serve as the testing dataset. The universality and robustness of the prediction model are further validated by the variability exhibited in the dataset. To ensure data consistency

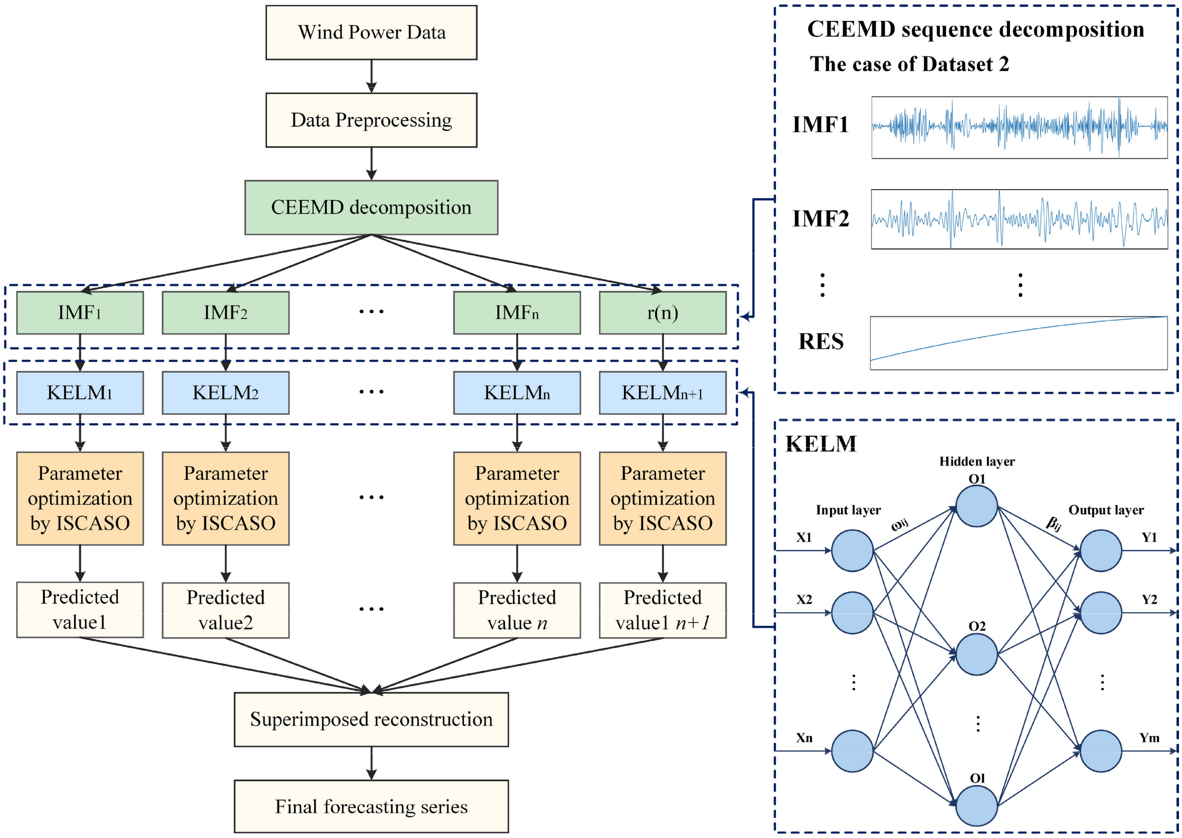

FIGURE 6. Flowchart of the CEEMD-ISCASO-KELM model.

across dimensions, mitigate the impact of diverse factors and disparate data units on the model, and expedite model training and convergence, all input and output variables were standardized within the range of [0,1] as follows:

$$y' = \frac{y - y_{\min}}{y_{\max} - y_{\min}} \tag{39}$$

where, y' is the normalized data, y is the original data value, $y_{\max}$ and $y_{\min}$ are the maximum and minimum values in the original data, respectively.

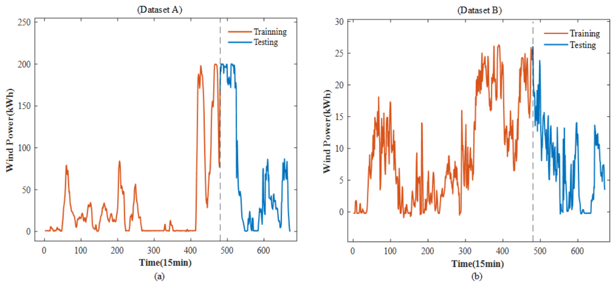

FIGURE 7. Original wind power sequence (a) Dataset A (b) Dataset B.

TABLE 4. Statistical characteristics of the two dataset.

Dataset	Sample	Num	Min	Max	Mean	Std
Dataset A	Training	480	0.11	199.76	30.01	49.33
	Testing	192	0.38	199.98	67.58	71.61
Dataset B	Training	480	-0.90	26.31	9.39	8.02
	Testing	192	-0.30	23.85	7.19	5.94

B. ENVIRONMENT CONFIGURATION AND PARAMETER SETTING

The simulation environment in this study is based on Intel®Core$^{\text{TM}}$i5-10500 CPU @3.10GHz, the simulation uses MATLAB2020b to build a deep learning framework and VISIO as a drawing tool.

To assess the reliability and validity of the CEEMD-ISCASO-KELM, this study compares its performance with a series of wind power prediction models using two datasets. The parameter settings in Table 5 for the models employed in this experiment were determined based on a comprehensive review of previous literature and extensive experimentation. According to previous experimental studies [58], [59], the parameters of CEEMD are set as follows: the standard deviation of the added white noise (Nstd) is set to 0.02, and the number of selected white noise pairs (NE) is 100.

To provide a more detailed explanation of the selection of the KELM hyperparameters, this study employ a Grid Search Cross Validation (GridSearchCV) experiment in conjunction with prior research when utilizing the KELM model for

TABLE 5. Hyperparameter settings of models.

Model	Hyperparameter
SVR	Kernel_Type = Linear, C=1.25
BP	Hidden = 7, epochs = 300, learning rate = 0.001
IPDL	Epoch = 300, learning rate = 0.001, batch size = 64
RAE	Epoch = 300, learning rate = 0.001, batch size = 64
KELM	Kernel_Type = RBF, C=100, tho=100
SO	T = 100, N = 100, ub = 100, lb = 0.001
ISCASO	$w_{max} = 0.9$, $w_{\min} = 0.2$, T = 100, N = 100, ub = 100, lb = 0.001

dataset prediction. The ten-fold cross validation is executed in a given parametric grid, and the Mean Square Error (MSE) is used to determine the prediction performance of the KELM with different the regularization coefficient C and kernel parameter σ when employing an RBF kernel function. The hyperparameters for the GridSearchCV to determine the KELM are presented in Fig. 8. In dataset A, the MSE reaches a minimum of 0.0049 when C and σ are set to 100 and 10, respectively. And in dataset B, the MSE is minimized at 0.0288 with the regularization coefficient C of 50 and kernel parameter σ of 10, but the MSE is also small at 0.0302 when C is 100 and σ is 10, which is not a significant difference between the two combinations. Therefore, to enhance the predictive performance on both datasets, the regularization coefficient C of the KELM model is set to 100 and kernel parameter σ is 10 in this study.

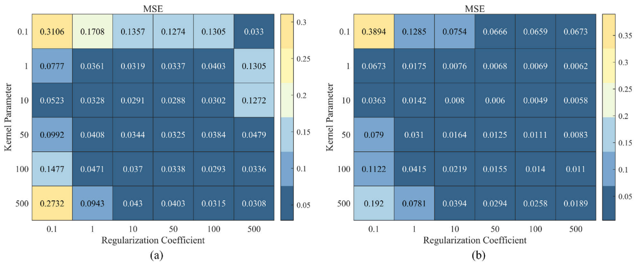

FIGURE 8. 10-fold cross-validation of KELM parameters (a) Dataset A (b) Dataset B.

The hyperparameter determination of the aforementioned KELM still involves human-determined factors. To further enhance the predictive accuracy of KELM, the study optimizes the regularization coefficients C and kernel parameters σ using the proposed ISCASO algorithm. Referring to the previous study by Hariharan et al. [57], to ensure optimal accuracy of the optimization parameters, the search range for the regularization coefficient C and kernel parameter σ is set to $[2^{-5}, 2^{5}]$. The population size and maximum number of iterations of the algorithm are set to 100 and 100, respectively.

C. SIGNAL DECOMPOSITION

Given the substantial randomness and volatility of the data, we employed Complete Ensemble Empirical Mode Decomposition (CEEMD) as a signal preprocessing technique for the original wind power dataset. This makes the individual Intrinsic Mode Function (IMF) components of the decomposed wind power sequence smoother when the frequency decreases, with clearer periodicity. By analyzing and processing different IMFs, it is possible to separate long-term and short-term trends in the data, ultimately aiming at improving the accuracy of short-term wind power prediction. The two sets of wind power sequences are decomposed into 8 IMFs components and one residual component RES by CEEMD. The results after decomposition are shown in Fig. 9, where the IMF1 has the largest fluctuations, highest frequency, and shortest wavelength. The frequencies of IMF2 to IMF8 gradually decrease, but their wavelengths gradually increase. After processing by CEEMD, the volatility of the two sets of wind power data is greatly reduced, and the original sequence is decomposed into IMFs with periodicity, thereby alleviating the difficulty of prediction.

D. EVALUATION METRICS

To better observe and evaluate the prediction performance of the model, we introduce root mean squared error (RMSE), mean absolute error (MAE), and coefficient of determination (R^2) as evaluation metrics for the model. Furthermore, according to the relevant national regulations and previous prediction experiments, the maximum prediction error $\delta_{\max}$ is added as a common measure of the model prediction results. Among these metrics, RMSE, MAE, $\delta_{\max}$ are commonly employed to quantify the disparity between predicted and actual values, with smaller values indicating superior predictive performance. R^2 serves as an indicator of model accuracy in fitting the regression curve, with higher values approaching 1 denoting better fit.

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n}(\widehat{y}_i - y_i)^2} \tag{40}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|(\hat{y}_i - y_i)\right| \tag{41}$$

$$\delta_{\max} = \max(\left|\widehat{y}_i - y_i\right|) \tag{42}$$

$$R^2 = 1 - \frac{\sum\limits_{i=1}^{n}(\widehat{y}_i - y_i)^2}{\sum\limits_{i=1}^{n}(\bar{y}_i - y_i)^2} \tag{43}$$

where, n is the number of samples, $\widehat{y}_i$ is the predicted value of wind power in the ith sample. y_i is the actual value of wind power in the ith sample. $\bar{y}_i$ is the average value of wind power for the sample.

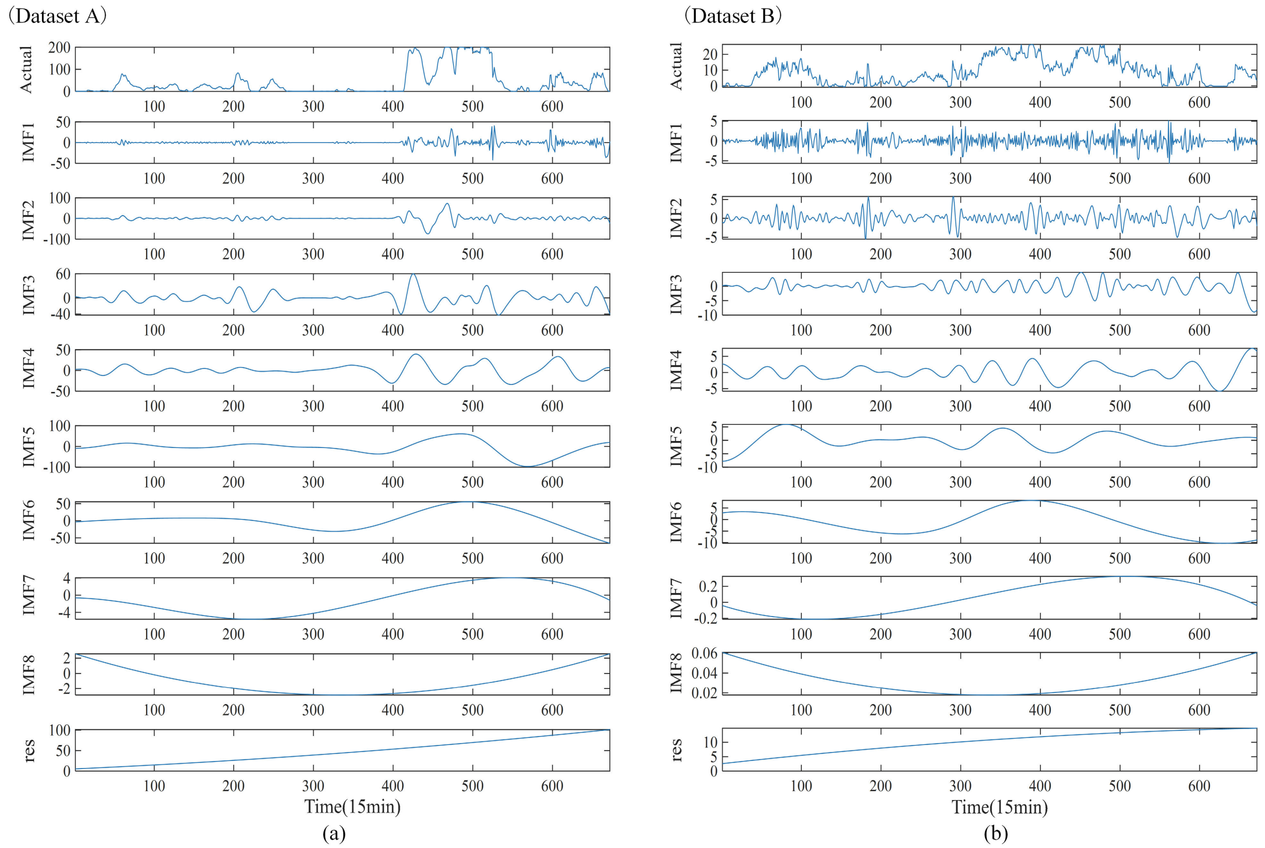

FIGURE 9. Results of CEEMD decomposition (a) Dataset A (b) Dataset B.

E. EXPERIMENTAL RESULTS ANALYSIS

1) ANALYSIS BASED ON PREDICTED VALUES

The prediction accuracy of CEEMD-ISCASO-KELM is validated in this section through the analysis of forecast-real graph and scatter plot, and evaluation indexes as follows: (1) We compare the predictive accuracy of ISCASO-KELM with that of KELM, SO-KELM, and demonstrate that ISCASO optimizes KELM parameters to improve prediction accuracy. (2) We compare KELM, EMD-KELM, and CEEMD-KELM models to show that sequence smoothing effectively enhances the predictive performance of KELM on wind power data. (3) Several commonly used machine learning prediction models, including BP, SVR, IPDL, RAE, and CEEMD-ISCASO-KELM, are selected to demonstrate the effectiveness of the proposed model in enhancing prediction accuracy for a single prediction model. (4) The comparative analysis of ten models reveals that the CEEMD-ISCASO-KELM model demonstrates superior predictive accuracy, thereby affirming its exceptional performance in prediction.

Fig. 10 shows the prediction simulation plots of the three models, KELM, SO-KELM, and ISCASO-KELM, using 192 test data from the two datasets respectively. Overall, all three models exhibit a similar trend to the actual curve on both datasets. However, KELM displays greater deviation from the actual curve at certain data points and demonstrates the poorest predictive performance among them. In contrast, ISCASO-KELM closely approximates the actual curve and achieves superior prediction accuracy compared to both KELM and SO-KELM. This finding confirms that by utilizing ISCASO for optimizing KELM parameters instead of manual tuning, it is possible to significantly improve its predictive accuracy while enhancing model robustness.

Fig. 11 shows the prediction simulation plots of the three types of models, KELM, EMD-KELM, and CEEMD-KELM, and the predicted values of the three models on the two datasets have a relatively similar trend to the real curves. However, we observed that the predictions obtained from the CEEMD-KELM model are closer to the actual values compared to those from the KELM and EMD-KELM models. A longitudinal comparison of different empirical modal decomposition methods reveals that the CEEMD incorporates auxiliary noises consisting of positive and negative white noise with opposite magnitudes added to the original sequence. This technique effectively eliminates residual noise, and reduces modal mixing and iteration processes, resulting in a smoother decomposed sequence which subsequently enhances the prediction accuracy of the KELM model. Furthermore, it is evident that both EMD-KELM and CEEMD-KELM models outperform

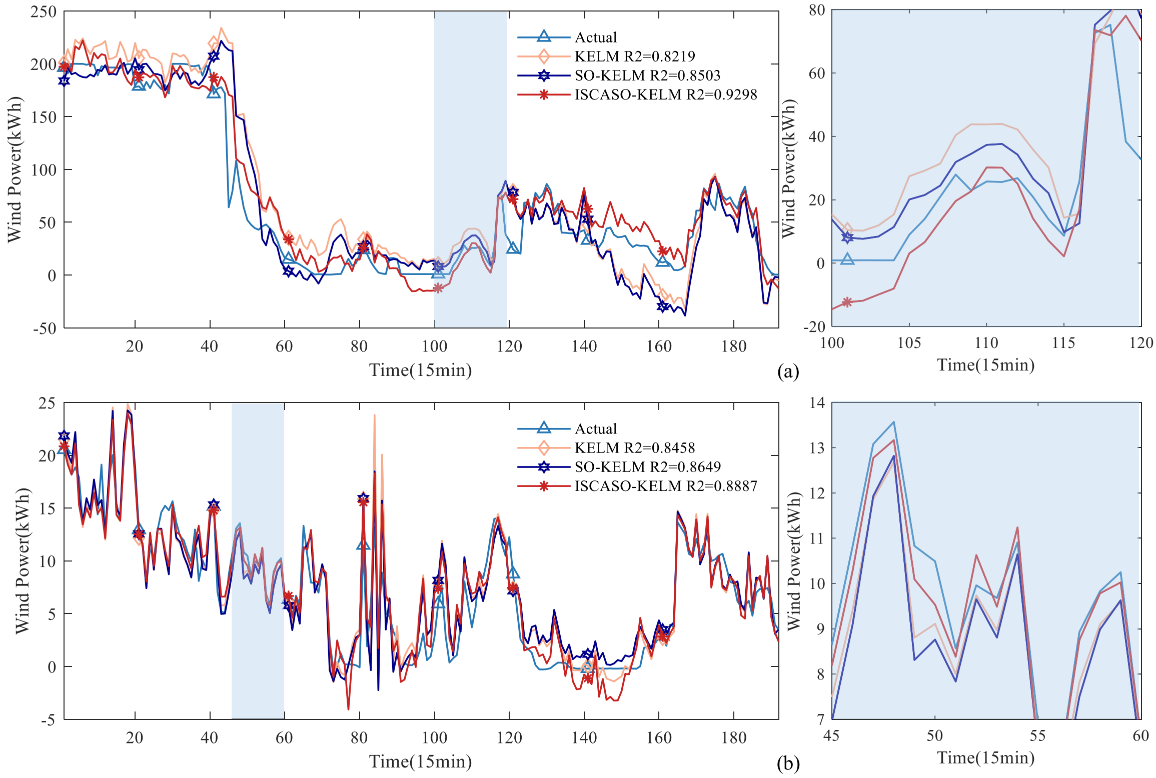

FIGURE 10. Prediction simulation graphs for KELM, SO-KELM, and ISCASO-KELM (a) Dataset A (b) Dataset B.

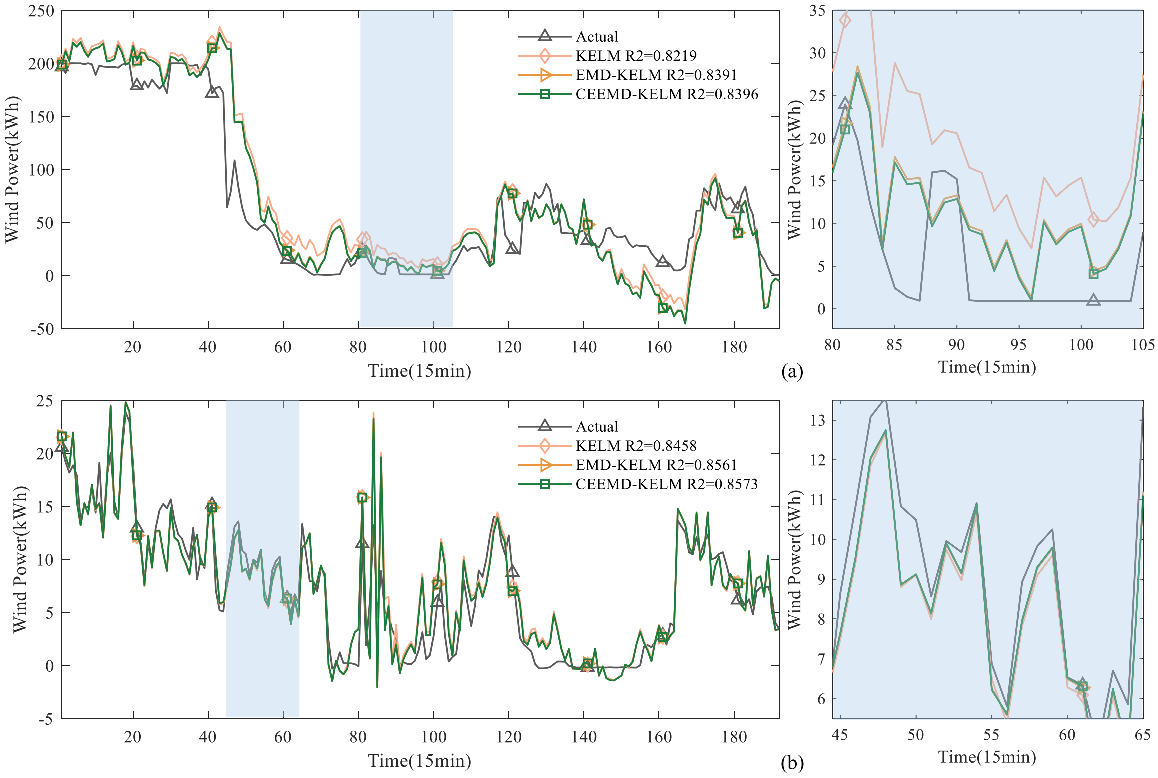

FIGURE 11. Prediction simulation graphs for KELM, EMD-KELM, and CEEMD-KELM (a) Dataset A (b) Dataset B.

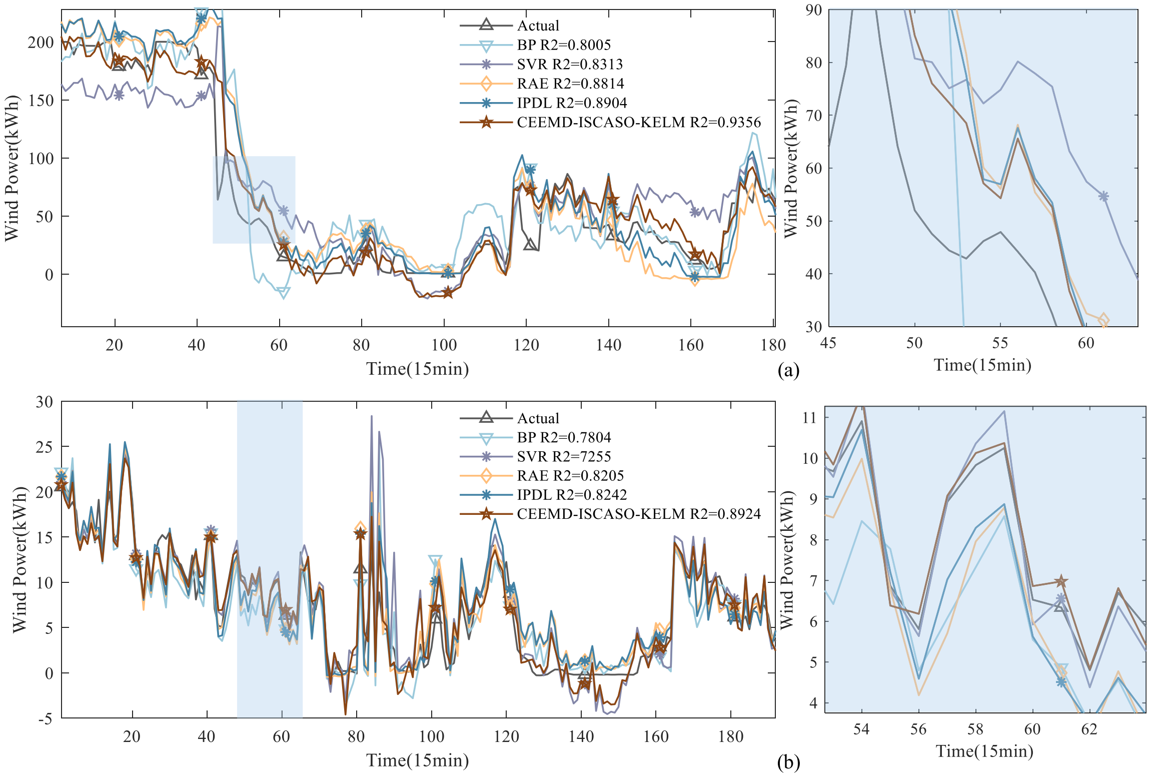

FIGURE 12. Prediction simulation graphs for BP, SVR, RAE, IPDL, and CEEMD-ISCASO-KELM (a) Dataset A (b) Dataset B.

KELM in terms of prediction performance. This can be attributed to their ability to smoothen irregular fluctuations and high-frequency noise present in the original sequence through a sequence smoothing process. Consequently, this enables the KELM model to capture periodic as well as long-term trends exhibited by wind power sequences with diverse characteristics leading to significant improvement in prediction accuracy.

Fig. 12 shows the prediction simulation graphs of five models, namely SVR, BP, IPDL, RAE, and CEEMD-ISCASO-KELM, It is evident that the single prediction methods exhibit significant instability when forecasting non-stationary wind power series, particularly during periods of high fluctuation, especially at the data inflection point, the prediction becomes more challenging and presents a large error. Specifically, the prediction of wind power trend by the BP neural network lacks precision and exhibits extreme values at the data extremes. Moreover, SVM's wind power prediction shows a significant discrepancy at the data plateau, particularly for high wind power, with results tending to underestimate actual power. However, the prediction results obtained by IPDL, RAE, and CEEMD-ISCASO-KELM models on the two datasets are more similar to the fluctuations of the actual data, thereby yielding more accurate predictions. However, by observing the local zoomed-in comparison graphs, it can be seen that the prediction curve of the proposed CEEMD-ISCASO-KELM multi-step model is closer to the actual wind power curve than that of the above single prediction model, which indicates that the proposed multi-step prediction model can effectively improve the prediction accuracy of the single prediction model.

Fig. 13 shows the prediction simulation of the ten models involved in this study. It can be observed from the R^2 values that the single prediction models exhibit poor accuracy, while the multi-step model demonstrates relatively high accuracy across datasets with different characteristics. Notably, the proposed ISCASO-KELM and CEEMD-ISCASO-KELM prediction curves exhibit the highest overlap with actual wind power curves and offer smoother fitting. The prediction accuracy of CEEMD-ISCASO-KELM is superior in both datasets. This indicates that the proposed CEEMD-ISCASO-KELM model synthesizes the advantages of sequence smoothing and hyperparameter search more effectively for better short-term wind power prediction. Furthermore, it demonstrates good adaptability and robustness across different datasets.

Fig. 14 illustrates the linear regression relationship between short-term wind power prediction results and actual power for two different characteristics. The CEEMD-ISCASO-KELM model exhibits a denser scatter, with a more concentrated error distribution in dataset A that has a larger variance, and all its linear regression goodness-of-fit values are above 93% in both datasets. In contrast, both types of data in a single model have more discrete error distributions, whereas the combined model shows relatively denser

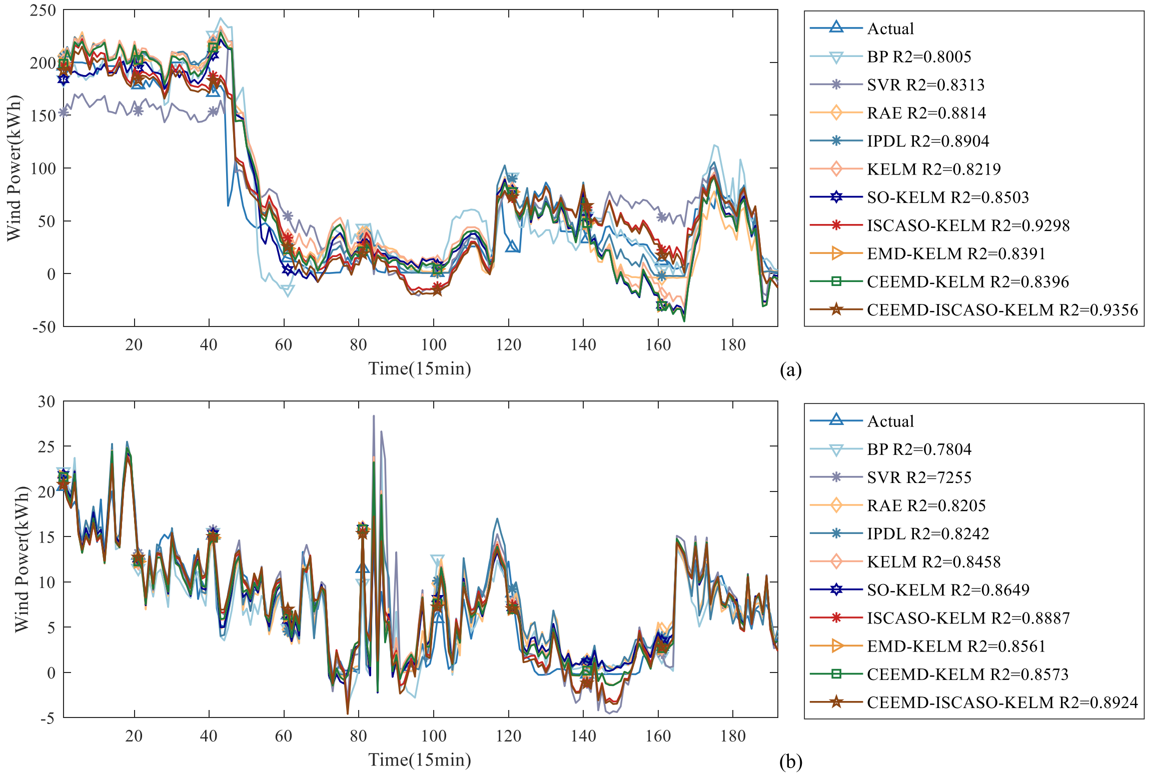

FIGURE 13. Prediction simulation graphs for ten prediction models (a) Dataset A (b) Dataset B.

error distributions. Moreover, the integration of ISCASO with KELM significantly enhances the goodness-of-fit of linear regression. Additionally, adding signal decomposition improves prediction accuracy across all models, specifically, the CEEMD-ISCASO-KELM outperforms single-model KELM. Therefore, the proposed CEEMD-ISCASO-KELM in this study integrates the benefits of sequence smoothing and hyperparameter optimization, thereby enhancing its capability to accurately forecast short-term wind power. And, it exhibits remarkable adaptability and robustness across diverse datasets.

2) ANALYSIS BASED ON PREDICTED VALUES

The corresponding prediction error evaluation indexes of the six prediction algorithms are shown in Table 6, and the histograms of prediction error evaluation indexes are shown in Fig. 15 and Fig. 16.

By comparing the prediction errors of SO-KELM, ISCASO-KELM, and KELM, it is evident that the optimization of KELM parameters using SO leads to improved predictions. The proposed ISCASO algorithm in this study significantly enhances the prediction results by optimizing the KELM parameters. In dataset A, both RMSE and MAE are enhanced by 37.23% and 38.36%, respectively, and R2 improves by 0.1079. Furthermore, in dataset B, RMSE and MAE improve by 16.34% and 13.04%, respectively, with a reduction of maximum prediction error by 4.7123 units. Therefore, the ISCASO can update the optimal fitness of the population particles based on the optimal fitness values of the current particles, and iteratively determine the optimal hyperparameters of KELM, which overcomes the problem of low accuracy caused by KELM's experience-based parameter selection.

Comparing the prediction results of EMD-KELM, CEEMD-KELM, and KELM models, signal decomposition using CEEMD demonstrated improved evaluation metrics compared to EMD signal decomposition. Among them, in dataset A the EMD-KELM and CEEMD-KELM models exhibited enhancements in RMSE values by 1.5351 and 1.4913 respectively, as well as MAE improvements of 10.64% and 8.86%, and CEEMD-KELM had the highest goodness-of-fit and the smallest maximum error of prediction. Similarly, in dataset B, the prediction performance of CEEMD-KELM surpassed that of EMD-KELM and KELM models. Therefore, combining the features of original sequences through CEEMD and EMD enables adaptive sequence decomposition while enhancing the prediction accuracy of the KELM model. Furthermore, CEEMD effectively addresses issues such as modal aliasing encountered in the EMD technique while outperforming it in terms of sequence smoothing performance.

Comparing the prediction effects of BP, SVR, IPDL, RAE, KELM, and CEEMD-ISCASO-KELM, the prediction accuracy of the multi-step prediction model is significantly enhanced compared to the single prediction model and demonstrates greater robustness. In the two datasets with

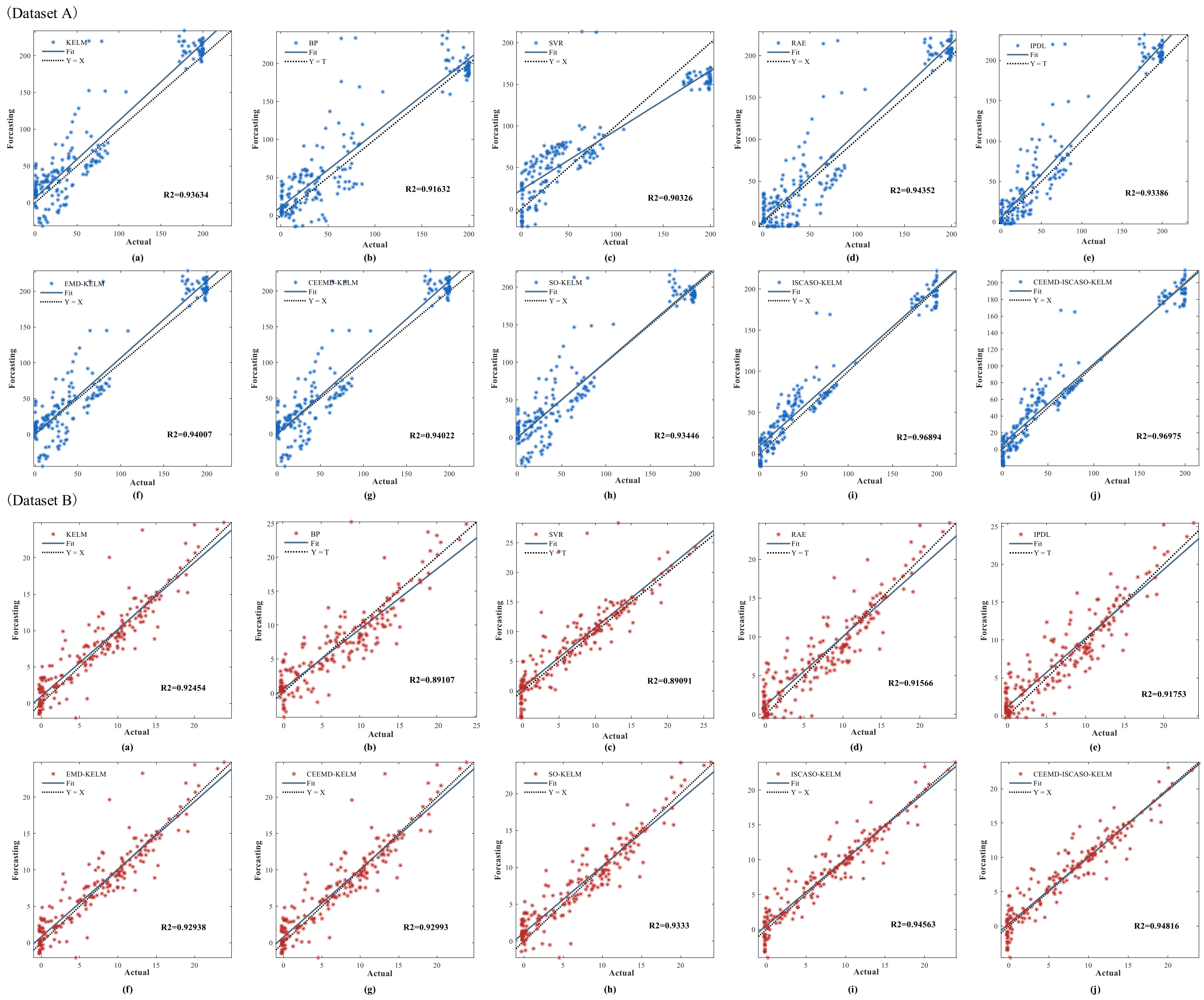

FIGURE 14. Linear regression graph of predicted and true values.

significantly different data distributions, the prediction results exhibited substantial variations among each single models. For instance, in dataset A, IPDL demonstrated superior predictive performance as a single model, while its predictions were unsatisfactory in dataset B. This indicates that the single prediction models lack robustness and fail to meet practical requirements. Conversely, the proposed CEEMD-ISCASO-KELM consistently outperformed in both datasets, exhibiting significant improvements across all evaluation indices and demonstrating strong robustness. Compared with IPDL, CEEMD-ISCASO-KELM reduced RMSE and RAE by 31.86% and 28.36%, respectively, and increased R^2 by 13.83% in dataset A. On dataset B, CEEMD-ISCASO-KELM resulted in decreases of 20.72% and 25.22% in RMSE and MAE, respectively, along with an increase of 0.0682 in R^2.

Compared to the aforementioned comparative models, both ISCASO-KELM and CEEMD-ISCASO-KELM proposed in this study exhibit satisfactory performance in short-term wind power prediction, leading to significant improvements in prediction evaluation metrics. However the prediction accuracy of CEEMD-ISCASO-KELM is less improved compared with ISCASO-KELM. The prediction evaluation metrics are employed to further quantify the efficacy of the proposed CEEMD-ISCASO-KELM model in this study, demonstrating a reduction in RMSE by 39.88% and 16.47%, as well as a decrease in MAE by 41.66% and 13.71% for dataset A and dataset B, respectively, when compared to KELM. Moreover, the goodness-of-fit R^2 values of 0.9356 and 0.8924 for dataset A and dataset B, respectively, closest to 1, and the maximum prediction error is 102.8426 and 5.5999, respectively, affirming that our model exhibits superior predictive accuracy.

3) ABLATION STUDY

To investigate the contribution of individual components in enhancing prediction performance, this study conducts an ablation study on the proposed CEEMD-ISCASO-KELM

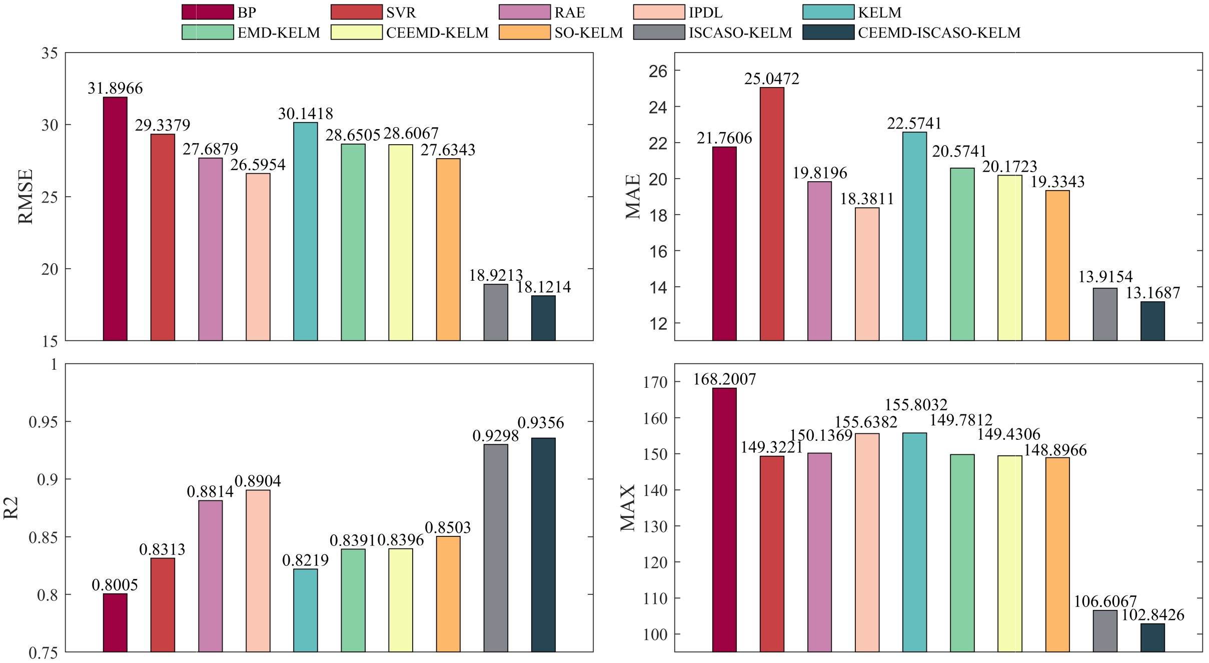

FIGURE 15. Comparison of the evaluation criteria of Dataset A.

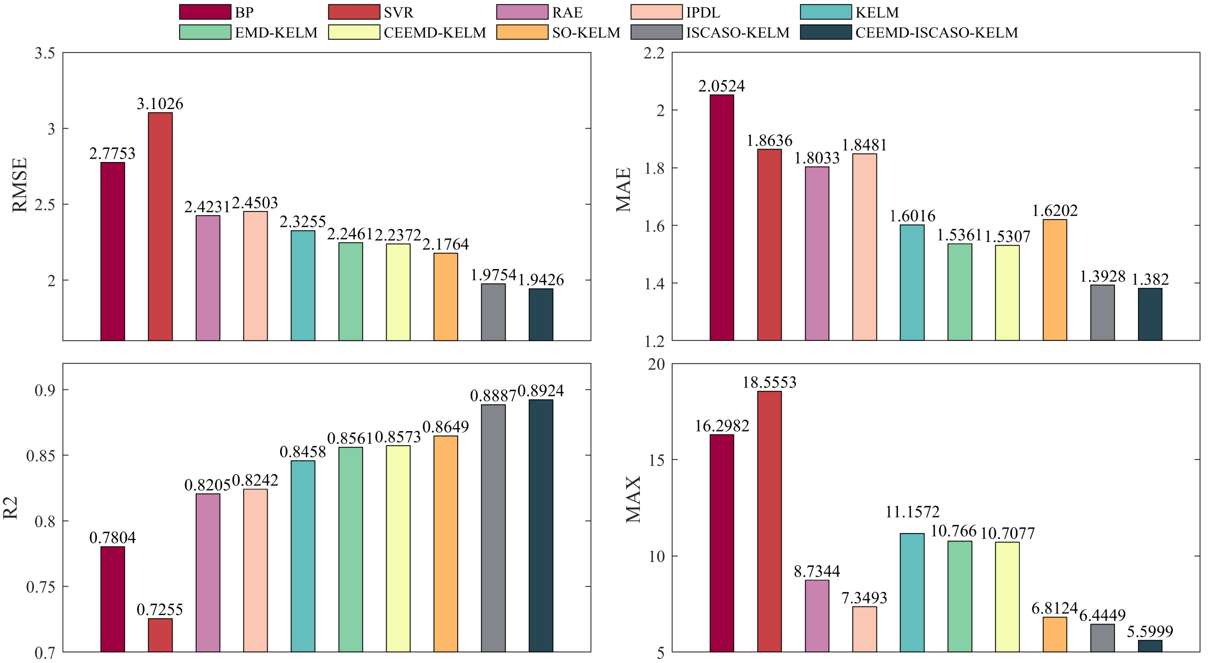

FIGURE 16. Comparison of the evaluation criteria of Dataset B.

prediction model using two datasets. The data decomposition component of CEEMD and the algorithm optimization component of ISCASO were separately extracted from the proposed model framework to obtain four models for comparison in the ablation study: KELM, ISCASO-KELM, CEEMD-KELM, and CEEMD-ISCASO-KELM. The aforementioned models all maintain the parameter settings mentioned above.

The ablation study's prediction results are presented in Table 7, demonstrating that both CEEMD and ISCASO provide a partial enhancement to the model on both datasets compared to the prediction results of the original KELM.

TABLE 6. Predictive evaluation indicator values for the model.

Dataset	Model	RMSE	MAE	R2	δ_{max}
Dataset A	BP	31.8966	21.7606	0.8005	168.2007
	SVR	29.3379	25.0472	0.8313	149.3221
	RAE	27.6879	19.8196	0.8814	150.1369
	IPDL	26.5954	18.3811	0.8904	155.6382
	KELM	30.1418	22.5741	0.8219	155.8032
	EMD-KELM	28.6505	20.5741	0.8391	149.7812
	CEEMD-KELM	28.6067	20.1723	0.8396	149.4306
	SO-KELM	27.6343	19.3343	0.8503	148.8966
	ISCASO-KELM	18.9213	13.9154	0.9298	106.6067
	CEEMD-ISCASO-KELM	**18.1214**	**13.1687**	**0.9356**	**102.8426**
Dataset B	BP	2.7753	2.0524	0.7804	16.2982
	SVR	3.1026	1.8636	0.7255	18.5553
	RAE	2.4231	1.8033	0.8205	8.7344
	IPDL	2.4503	1.8481	0.8242	7.3493
	KELM	2.3255	1.6016	0.8458	11.1572
	EMD-KELM	2.2461	1.5361	0.8561	10.7660
	CEEMD-KELM	2.2372	1.5307	0.8573	10.7077
	SO-KELM	2.1764	1.6202	0.8649	6.8124
	ISCASO-KELM	1.9754	1.3928	0.8887	6.4449
	CEEMD-ISCASO-KELM	**1.9426**	**1.3820**	**0.8924**	**5.5999**

TABLE 7. Predicted results of ablation study.

Dataset	Backbone	CEEMD	ISCASO	RMSE	MAE	R2	δ_{max}
Dataset A	KELM	×	×	30.1418	22.5741	0.8219	155.8032
	KELM	√	×	28.6067	20.1723	0.8396	149.4306
	KELM	×	√	18.9213	13.9154	0.9298	106.6067
	KELM	√	√	18.1214	13.1687	0.9356	102.8426
Dataset B	KELM	×	×	2.3255	1.6016	0.8458	11.1572
	KELM	√	×	2.2372	1.5307	0.8573	10.7077
	KELM	×	√	1.9754	1.3928	0.8887	6.4449
	KELM	√	√	1.9426	1.3820	0.8924	5.5999

After incorporating the data decomposition model CEEMD alone, both the RMSE and MAE of the two datasets were reduced, leading to further improvement in fitting accuracy. This finding highlights the significant impact of signal decomposition on enhancing the performance of wind power prediction models, while also indicating that the original wind power's nonlinearity and non-stationarity limit the predictability of KELM. When ISCASO was separately introduced for optimizing two KELM model parameters, a notable enhancement in prediction accuracy was observed for KELM. In dataset A, there was a reduction of 37.22% and 38.36% in RMSE and MAE respectively, whereas, in dataset B, reductions of 15.05% and 13.04% were achieved for RMSE and MAE respectively. This demonstrates the effectiveness of using ISCASO for parameter optimization in KELM. Finally, after further fusing the two parts, the RMSE and MAE in dataset A were reduced by 39.88% and 41.66% respectively, while R^2 increased by 13.83%, and the maximum prediction error decreased by 52.9606 in numerical terms. In dataset B, RMSE, MAE, and maximum prediction error were also significantly reduced, and R^2 was increased by 5.5%. These results demonstrate that incorporating CEEMD and ISCASO into KELM can effectively enhance model accuracy through wind power series decomposition and optimization of KELM parameters. Thus, the ablation study confirms the effectiveness and necessity of the proposed CEEMD-ISCASO-KELM model design, with each component showcasing its unique advantages.

4) SENSITIVITY ANALYSIS

To demonstrate the robustness of the proposed CEEMD-ISCASO-KELM model to noisy data, this study takes Dataset B as an example, noisy data with varying standard deviations are added to the original dataset, and predictions are made using the proposed model. The amount of noise added in each case is presented in Table 8. Fig. 17 illustrates the probability density function (PDF) for each case, revealing that as the standard deviation increases, the distribution of prediction errors widens; however, they are all concentrated around 0. This sensitivity analysis highlights that the CEEMD-ISCASO-KELM model exhibits excellent adaptability to diverse environmental factors such as wind speed and direction, demonstrating remarkable stability, generalization ability, and robustness against noise and data uncertainty.

TABLE 8. Noise details of the different cases.

Case	Standard deviation
Case 1	1 kWh
Case 2	2 kWh
Case 3	3 kWh
Case 4	4 kWh
Case 5	5 kWh
Case 6	6 kWh
Case 7	7 kWh
Case 8	8 kWh
Case 9	9 kWh
Case 10	10 kWh

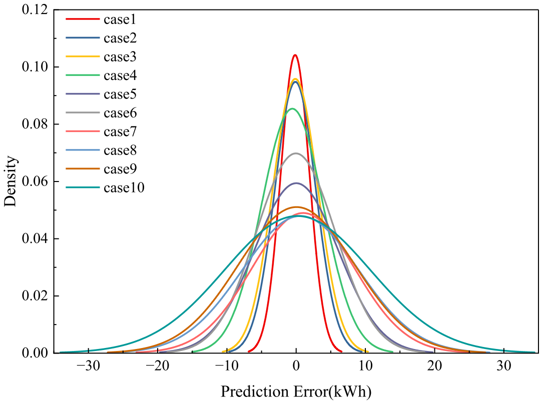

FIGURE 17. PDF of the prediction errors for each case.

TABLE 9. Computational time complexity.

Model	TIME(s)
KELM	9.0571
EMD-KELM	17.4697
CEEMD-KELM	25.0897
SO-KELM	60.4442
ISCASO-KELM	67.9117
CEEMD-ISCASO-KELM	**203.9572**

F. COMPUTATIONAL TIME COMPLEXITY

In terms of computational time cost, it is worth noting that a novel multi-step prediction model proposed in this study requires a longer runtime compared to the single prediction model. Therefore, the training time of CEEMD-ISCASO-KELM related models was compared and the specific results are presented in Table 9. Despite the long training time of the multistep prediction model, the prediction using the constructed multistep model takes only a few seconds to run in practice given the input feature variables. Consequently, the CEEMD-ISCASO-KELM proposed in this study is capable of obtaining more accurate prediction results in a short timeframe during practical applications.

VI. CONCLUSION

Wind power generation occupies an important position in the field of renewable energy, but the inherent volatility and intermittency of wind power can affect the accuracy of wind power prediction. Therefore, we propose a novel multi-step prediction model combining CEEMD, ISCASO, and KELM with ultra-short-term wind power as the research object. The CEEMD can effectively alleviate the volatility of the original signal and improve the stability of the signal. To improve the prediction accuracy of KELM, the SO algorithm based on the SCA algorithm and dynamic inertia weight improvement is proposed to optimize the KELM parameters. Based on the measured wind power data from two wind farms in China, the performance of the novel multi-step prediction model is evaluated from multiple perspectives using different evaluation indexes. We can draw several conclusions as follows:

(1) Complete Ensemble Empirical Mode Decomposition (CEEMD) effectively mitigates the issue of modal aliasing, leading to improved modal decomposition results and enhanced computational efficiency. Employing CEEMD for decomposing the original wind power sequence successfully addresses the problem of large errors caused by unstable data, thereby reducing volatility and randomness in the dataset.

(2) The proposed ISCASO effectively improves the defects of SO by introducing chaotic mapping to improve the population diversity and make the initial snake position distribution more uniform. The introduction of a dynamic inertia weight factor and an improved SCA algorithm enhances global search capability, facilitates escape from local optima, and mitigates premature convergence. Experimental results show that the improved SO optimizes the problem that the traditional SO has poor early iteration period global search ability and susceptibility to local optima in later stages, thereby offering significant practical implications. Compared with the single KELM model, the ISCASO optimizes the regularization coefficients C and kernel parameters σ of KELM, resulting in a significant reduction in RMSE and MAE, as well as an increase in R^2. This can be verified that ISCASO enhances the predictive accuracy of the KELM by effectively optimizing its parameters.

(3) By comparing the ten models on two datasets, the multi-step model CEEMD-ISCASO-KELM developed in this study generates a higher consistency between the prediction curves and the actual curves. Meanwhile, it has the lowest root mean square error (RMSE) and mean absolute error (MAE), which provides better performance and higher efficiency in decomposing the wind power series to achieve more accurate prediction results compared to other models.

The proposed multi-step prediction model CEEMD-ISCASO-KELM in this study demonstrates robustness and effectiveness in wind power prediction. In addition to its application in wind power prediction, the model exhibits great potential for various fields including finance and engineering, such as stock market index prediction, air quality index forecasting, traffic flow estimation, and cold mill vibration analysis.

REFERENCES

[1] Y. Ding, Z. Chen, H. Zhang, X. Wang, and Y. Guo, "A short-term wind power prediction model based on CEEMD and WOA-KELM," *Renew. Energy*, vol. 189, pp. 188–198, Apr. 2022, doi: 10.1016/j.renene.2022.02.108.

[2] B. Yang, L. Zhong, J. Wang, H. Shu, X. Zhang, T. Yu, and L. Sun, "State-of-the-art one-stop handbook on wind forecasting technologies: An overview of classifications, methodologies, and analysis," *J. Cleaner Prod.*, vol. 283, Feb. 2021, Art. no. 124628, doi: 10.1016/j.jclepro.2020.124628.

[3] X. Dong, D. Wang, J. Lu, and X. He, "A wind power forecasting model based on polynomial chaotic expansion and numerical weather prediction," *Electr. Power Syst. Res.*, vol. 227, Feb. 2024, Art. no. 109983, doi: 10.1016/j.epsr.2023.109983.

[4] H. Wang, S. Han, Y. Liu, J. Yan, and L. Li, "Sequence transfer correction algorithm for numerical weather prediction wind speed and its application in a wind power forecasting system," *Appl. Energy*, vol. 237, pp. 1–10, Mar. 2019, doi: 10.1016/j.apenergy.2018.12.076.

[5] M. Yang, Y. Guo, and Y. Huang, "Wind power ultra-short-term prediction method based on NWP wind speed correction and double clustering division of transitional weather process," *Energy*, vol. 282, Nov. 2023, Art. no. 128947, doi: 10.1016/j.energy.2023.128947.

[6] Y. Chen, M. Bai, Y. Zhang, J. Liu, and D. Yu, "Multivariable space-time correction for wind speed in numerical weather prediction (NWP) based on ConvLSTM and the prediction of probability interval," *Earth Sci. Informat.*, vol. 16, no. 3, pp. 1953–1974, Jul. 2023, doi: 10.1007/s12145-023-01036-1.

[7] W. Sun and M. Liu, "Wind speed forecasting using FEEMD echo state networks with RELM in Hebei, China," *Energy Convers. Manage.*, vol. 114, pp. 197–208, Apr. 2016, doi: 10.1016/j.enconman.2016.02.022.

[8] T. Ishikawa, T. Kojima, and T. Namerikawa, "Short-term wind power prediction for wind turbine via Kalman filter based on JIT modeling," *Electr. Eng. Jpn.*, vol. 198, no. 3, pp. 86–96, Nov. 2016, doi: 10.1002/eej.22888.

[9] A. Joshuva and V. Sugumaran, "A machine learning approach for condition monitoring of wind turbine blade using autoregressive moving average (ARMA) features through vibration signals: A comparative study," *Prog. Ind. Ecol., Int. J.*, vol. 12, nos. 1–2, p. 14, Oct. 2018, doi: 10.1504/pie.2018.095867.

[10] T. González Grandón, J. Schwenzer, T. Steens, and J. Breuing, "Electricity demand forecasting with hybrid classical statistical and machine learning algorithms: Case study of Ukraine," *Appl. Energy*, vol. 355, Feb. 2024, Art. no. 122249, doi: 10.1016/j.apenergy.2023.122249.

[11] L. Zhang, Z. Guo, Q. Tao, Z. Xiong, and J. Ye, "XGBoost-based short-term prediction method for power system inertia and its interpretability," *Energy Rep.*, vol. 9, pp. 1458–1469, Sep. 2023, doi: 10.1016/j.egyr.2023.04.065.

[12] J. Gao, X. Ye, X. Lei, B. Huang, X. Wang, and L. Wang, "A multichannel-based CNN and GRU method for short-term wind power prediction," *Electronics*, vol. 12, no. 21, p. 4479, Oct. 2023, doi: 10.3390/electronics12214479.

[13] B. Qin, X. Huang, X. Wang, and L. Guo, "Ultra-short-term wind power prediction based on double decomposition and LSSVM," *Trans. Inst. Meas. Control*, vol. 45, no. 14, pp. 2627–2636, Feb. 2023, doi: 10.1177/01423312231153258.

[14] R. Babu and M. S. Syed, "Wind power prediction using hybrid ELM algorithm," *J. Comput. Sci. Control Syst.*, vol. 16, pp. 13–15, May 2023.

[15] X. Xiong, G. Jie, Z. Liang, Z. Ling, and W. Long, "A short-term wind power forecast method via XGBoost hyper-parameters optimization," *Frontiers Energy Res.*, vol. 10, May 2022, Art. no. 905155, doi: 10.3389/fenrg.2022.905155.

[16] J. Wang and X. Tang, "An intensive decomposition integration paradigm for short-term wind power forecasting based on feature extraction and optimal weighted combination strategy," *Measurement*, vol. 223, Dec. 2023, Art. no. 113811, doi: 10.1016/j.measurement.2023.113811.

[17] L. Yu, G. Meng, G. Pau, Y. Wu, and Y. Tang, "Research on hierarchical control strategy of ESS in distribution based on GA-SVR wind power forecasting," *Energies*, vol. 16, no. 4, p. 2079, Feb. 2023, doi: 10.3390/en16042079.

[18] M. Khodayar, O. Kaynak, and M. E. Khodayar, "Rough deep neural architecture for short-term wind speed forecasting," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 2770–2779, Dec. 2017, doi: 10.1109/TII.2017.2730846.

[19] M. Khodayar, J. Wang, and M. Manthouri, "Interval deep generative neural network for wind speed forecasting," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3974–3989, Jul. 2019, doi: 10.1109/TSG.2018.2847223.

[20] H. Li, L. Liu, and Q. He, "A spatiotemporal coupling calculation-based short-term wind farm cluster power prediction method," *IEEE Access*, vol. 11, pp. 131418–131434, 2023, doi: 10.1109/ACCESS.2023.3335629.

[21] X. Wang, J. Li, L. Shao, H. Liu, L. Ren, and L. Zhu, "Short-term wind power prediction by an extreme learning machine based on an improved Hunter–Prey optimization algorithm," *Sustainability*, vol. 15, no. 2, p. 991, Jan. 2023, doi: 10.3390/su15020991.

[22] L. Hua, C. Zhang, T. Peng, C. Ji, and M. S. Nazir, "Integrated framework of extreme learning machine (ELM) based on improved atom search optimization for short-term wind speed prediction," *Energy Convers. Manage.*, vol. 252, Jan. 2022, Art. no. 115102, doi: 10.1016/j.enconman.2021.115102.

[23] Q. Zhou, Y. Ma, Q. Lv, R. Zhang, W. Wang, and S. Yang, "Short-term interval prediction of wind power based on KELM and a universal tabu search algorithm," *Sustainability*, vol. 14, no. 17, p. 10779, Aug. 2022, doi: 10.3390/su141710779.

[24] H. Jiajun, Y. Chuanjin, L. Yongle, and X. Huoyue, "Ultra-short term wind prediction with wavelet transform, deep belief network and ensemble learning," *Energy Convers. Manage.*, vol. 205, Feb. 2020, Art. no. 112418, doi: 10.1016/j.enconman.2019.112418.

[25] Y. Zhang, L. Zhang, D. Sun, K. Jin, and Y. Gu, "Short-term wind power forecasting based on VMD and a hybrid SSA-TCN-BiGRU network," *Appl. Sci.*, vol. 13, no. 17, p. 9888, Aug. 2023, doi: 10.3390/app13179888.

[26] M. Ran, J. Huang, W. Qian, T. Zou, and C. Ji, "EMD-based gray combined forecasting model—Application to long-term forecasting of wind power generation," *Heliyon*, vol. 9, no. 7, Jul. 2023, Art. no. e18053, doi: 10.1016/j.heliyon.2023.e18053.

[27] H. Li and H. Zou, "Short-term wind power prediction based on data reconstruction and improved extreme learning machine," *Arabian J. Sci. Eng.*, vol. 47, no. 3, pp. 3669–3682, Feb. 2022, doi: 10.1007/s13369-020-05311-x.

[28] Z. Shang, Z. He, Y. Chen, Y. Chen, and M. Xu, "Short-term wind speed forecasting system based on multivariate time series and multi-objective optimization," *Energy*, vol. 238, Jan. 2022, Art. no. 122024, doi: 10.1016/j.energy.2021.122024.

[29] J.-N. Shan, H.-Z. Wang, G. Pei, S. Zhang, and W.-H. Zhou, "Research on short-term power prediction of wind power generation based on WT-CABC-KELM," *Energy Rep.*, vol. 8, pp. 800–809, Nov. 2022, doi: 10.1016/j.egyr.2022.09.165.

[30] Y. Han and X. Tong, "Multi-step short-term wind power prediction based on three-level decomposition and improved grey wolf optimization," *IEEE Access*, vol. 8, pp. 67124–67136, 2020, doi: 10.1109/ACCESS.2020.2984851.

[31] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. Roy. Soc. London A, Math., Phys. Eng. Sci.*, vol. 454, no. 1971, pp. 903–995, Mar. 1998, doi: 10.1098/rspa.1998.0193.

[32] N. Li, J. Dong, L. Liu, H. Li, and J. Yan, "A novel EMD and causal convolutional network integrated with transformer for ultra short-term wind power forecasting," *Int. J. Electr. Power Energy Syst.*, vol. 154, Dec. 2023, Art. no. 109470, doi: 10.1016/j.ijepes.2023.109470.

[33] Z. Wu and N. E. Huang, "Ensemble empirical mode decomposition: A noise-assisted data analysis method," *Adv. Adapt. Data Anal.*, vol. 1, no. 1, pp. 1–41, Jan. 2009, doi: 10.1142/s1793536909000047.

[34] J.-R. Yeh, J.-S. Shieh, and N. E. Huang, "Complementary ensemble empirical mode decomposition: A novel noise enhanced data analysis method," *Adv. Adapt. Data Anal.*, vol. 2, no. 2, pp. 135–156, Apr. 2010, doi: 10.1142/s1793536910000422.

[35] S. Zhu, P. Wang, R. Wang, M. Li, J. Che, and X. Wang, "CEEMD-LASSO-ELM nonlinear combined model of air quality index prediction for four cities in China," *Environ. Ecol. Statist.*, vol. 30, no. 3, pp. 309–334, Jun. 2023, doi: 10.1007/s10651-023-00562-x.

[36] K. Roushangar, S. Shahnazi, and A. A. Sadaghiani, "An efficient hybrid grey wolf optimization-based KELM approach for prediction of the discharge coefficient of submerged radial gates," *Soft Comput.*, vol. 27, no. 7, pp. 3623–3640, Nov. 2022, doi: 10.1007/s00500-022-07614-7.

[37] J. Li and M. Li, "Prediction of ultra-short-term wind power based on BBO-KELM method," *J. Renew. Sustain. Energy*, vol. 11, no. 5, Sep. 2019, Art. no. 056104, doi: 10.1063/1.5113555.

[38] X. Chen, Y. Li, Y. Zhang, X. Ye, X. Xiong, and F. Zhang, "A novel hybrid model based on an improved seagull optimization algorithm for short-term wind speed forecasting," *Processes*, vol. 9, no. 2, p. 387, Feb. 2021, doi: 10.3390/pr9020387.

[39] Y. Dai, J. Pang, Z. Li, W. Li, Q. Wang, and S. Li, "Modeling of thermal error electric spindle based on KELM ameliorated by snake optimization," *Case Stud. Thermal Eng.*, vol. 40, Dec. 2022, Art. no. 102504, doi: 10.1016/j.csite.2022.102504.

[40] J. Wang, X. Wang, X. Li, and J. Yi, "A hybrid particle swarm optimization algorithm with dynamic adjustment of inertia weight based on a new feature selection method to optimize SVM parameters," *Entropy*, vol. 25, no. 3, p. 531, Mar. 2023, doi: 10.3390/e25030531.

[41] H. Qing, X. Shuai, and W. Yuan, "Enhanced sine cosine algorithm based node deployment optimization of wireless sensor network," *J. Comput. Appl.*, vol. 39, pp. 2035–2043, Jul. 2019, doi: 10.11772/j.issn.1001-9081.2018112282.

[42] G. Hu, J. Wang, Y. Li, M. Yang, and J. Zheng, "An enhanced hybrid seagull optimization algorithm with its application in engineering optimization," *Eng. Comput.*, vol. 39, no. 2, pp. 1653–1696, Nov. 2022, doi: 10.1007/s00366-022-01746-y.

[43] J. Cao, A. Xue, Y. Yang, W. Cao, X. Hu, G. Cao, J. Gu, L. Zhang, and X. Geng, "Deep learning based soft sensor for microbial wastewater treatment efficiency prediction," *J. Water Process Eng.*, vol. 56, Dec. 2023, Art. no. 104259, doi: 10.1016/j.jwpe.2023.104259.

[44] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jul. 2004, pp. 985–990, doi: 10.1109/IJCNN.2004.1380068.

[45] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012, doi: 10.1109/TSMCB.2011.2168604.

[46] F. A. Hashim and A. G. Hussien, "Snake optimizer: A novel meta-heuristic optimization algorithm," *Knowl.-Based Syst.*, vol. 242, Apr. 2022, Art. no. 108320, doi: 10.1016/j.knosys.2022.108320.

[47] M. Liu, Y. Zhang, J. Guo, J. Chen, and Z. Liu, "An adaptive lion swarm optimization algorithm incorporating tent chaotic search and information entropy," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, p. 39, Mar. 2023, doi: 10.1007/s44196-023-00216-7.

[48] Q. Zhang, H. Liu, J. Guo, Y. Wang, L. Liu, H. Liu, and H. Cong, "Improved GWO-MCSVM algorithm based on nonlinear convergence factor and tent chaotic mapping and its application in transformer condition assessment," *Electr. Power Syst. Res.*, vol. 224, Nov. 2023, Art. no. 109754, doi: 10.1016/j.epsr.2023.109754.

[49] Y. Duan, N. Chen, L. Chang, Y. Ni, S. V. N. S. Kumar, and P. Zhang, "CAPSO: Chaos adaptive particle swarm optimization algorithm," *IEEE Access*, vol. 10, pp. 29393–29405, 2022, doi: 10.1109/ACCESS.2022.3158666.

[50] S. Janakiraman and M. D. Priya, "Hybrid grey wolf and improved particle swarm optimization with adaptive intertial weight-based multi-dimensional learning strategy for load balancing in cloud environments," *Sustain. Comput., Informat. Syst.*, vol. 38, Apr. 2023, Art. no. 100875, doi: 10.1016/j.suscom.2023.100875.

[51] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowl.-Based Syst.*, vol. 96, pp. 120–133, Mar. 2016, doi: 10.1016/j.knosys.2015.12.022.

[52] L. Yao, P. Yuan, C.-Y. Tsai, T. Zhang, Y. Lu, and S. Ding, "ESO: An enhanced snake optimizer for real-world engineering problems," *Expert Syst. Appl.*, vol. 230, Nov. 2023, Art. no. 120594, doi: 10.1016/j.eswa.2023.120594.

[53] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[54] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[55] Y. Wang, Y. Xiao, Y. Guo, and J. Li, "Dynamic chaotic opposition-based learning-driven hybrid Aquila optimizer and artificial rabbits optimization algorithm: Framework and applications," *Processes*, vol. 10, no. 12, p. 2703, Dec. 2022, doi: 10.3390/pr10122703.

[56] M. Zhou, Z. Zhu, F. Hu, K. Bian, and W. Lai, "An industrial load classification method based on a two-stage feature selection strategy and an improved MPA-KELM classifier: A Chinese cement plant case," *Electronics*, vol. 12, no. 15, p. 3356, Aug. 2023, doi: 10.3390/electronics12153356.

[57] H. Muthusamy, S. Ravindran, S. Yaacob, and K. Polat, "An improved elephant herding optimization using sine–cosine mechanism and opposition based learning for global optimization problems," *Expert Syst. Appl.*, vol. 172, Jun. 2021, Art. no. 114607, doi: 10.1016/j.eswa.2021.114607.

[58] J. Xiong, T. Peng, Z. Tao, C. Zhang, S. Song, and M. S. Nazir, "A dual-scale deep learning model based on ELM-BiLSTM and improved reptile search algorithm for wind power prediction," *Energy*, vol. 266, Mar. 2023, Art. no. 126419, doi: 10.1016/j.energy.2022.126419.

[59] T. Jiang and Y. Liu, "A short-term wind power prediction approach based on ensemble empirical mode decomposition and improved long short-term memory," *Comput. Electr. Eng.*, vol. 110, Sep. 2023, Art. no. 108830, doi: 10.1016/j.compeleceng.2023.108830.

MENGLING ZHAO received the bachelor's degree in applied mathematics from the Department of Mathematics, North-West University, in July 1998, and the Ph.D. degree from Xidian University, Xi'an, China. She is currently the Vice Professor of Science with Xi'an University of Science and Technology, China. She has teaching experience of 25 years. She has completed five scientific research projects. She has published more than 20 articles in various journals. Her current research interests include optimization algorithms and mathematical methods in data mining.

XUAN ZHOU is currently pursuing the M.Sc. degree with the Faculty of Science, Xi'an University of Science and Technology. Her current research interests include intelligent optimization algorithms and machine learning.