AN ADAPTIVE ONLINE SEQUENTIAL EXTREME LEARNING MACHINE FOR SHORT-TERM WIND SPEED PREDICTION BASED ON IMPROVED ARTIFICIAL BEE COLONY ALGORITHM

Z. Tian,* G. Wang,* Y. Ren,* S. Li,* Y. Wang*

Abstract: As an improved algorithm of standard extreme learning machine, online sequential extreme learning machine achieves excellent classification and regression performance. However, online sequential extreme learning machine gives the same weight to the old and new training samples, and fails to highlight the importance of the new training samples. At the same time, the algorithm updates the network weights after obtaining the new training samples. This network weight updating mode lacks flexibility and increases unnecessary computation. This paper proposes an adaptive online sequential extreme learning machine with an effective sample updating mechanism. The new and old samples are given different weights. The effect of new training samples on the algorithm is further enhanced, which can further improve the regression prediction ability of extreme learning machine. At the same time, an improved artificial bee colony algorithm is proposed and used to optimize the parameters of the adaptive online sequential extreme learning machine. The stability and convergence property of proposed prediction method are proved. The actual collected short-term wind speed time series is used as the research object and verify the prediction perfromance of the proposed method. Multi step prediction simulation of short-term wind speed is performed out. Compared with other prediction methods, the simulation results show that the proposed approach has higher prediction accuracy and reliability performance, meanwhile improve the performance indicators.

Key words: *extreme learning machine, improved artificial bee colony algorithm, adaptive online sequential, short-term wind speed, prediction*

---

*Zhongda Tian – Corresponding author; Gang Wang; Yi Ren; Shujiang Li; Yanhong Wang; Shenyang University of Technology, School of Information Science and Engineering, Shenyang, Liaoning, 110870, China, E-mail: tianzhongda@126.com, 1365812827@qq.com, 940261475@qq.com, lisj2005@126.com, sshuang123456@163.com

1.   Introduction

The traditional neural network learning algorithm uses the gradient descent method to train the network. The learning speed of algorithm is slow, and all the parameters of the network need to be adjusted iteratively. This problem seriously restricts the development of forward neural network [31]. In 2006, Huang et al proposed a new neural network - extreme learning machine extreme (ELM) algorithm [17]. The ELM algorithm uses the random mechanism to reduce the parameter setting and choice. It is one kind of simple feasible fast learning algorithm. Compared with other traditional neural network learning algorithms or support vector machine (SVM), least square support vector machine (LSSVM), etc, ELM algorithm has the advantages of fast learning speed and strong generalization ability [7, 41]. The authors pointed out that computing time of ELM is usually several thousand times faster than BP neural network or SVM [37]. Therefore, the ELM algorithm is applied to many classification and regression prediction problems [32–34, 47].

Based on standard ELM algorithm, the authors proposed an online sequential extreme learning machine (OS-ELM) algorithm [26]. OS-ELM first calculates its initial network weights in the initial training stage, and then the corresponding network weights can be obtained on the basis of the initial network weights when a new training sample is added to the training sample set. However, OS-ELM and other improved ELM algorithms (I-ELM [18], EI-ELM [19], EM-ELM [11], and etc) believe that the importance of new and old training samples is the same, giving the same weight to old and new samples, and failing to highlight the role of new training samples. Moreover, as long as new training samples are obtained, OS-ELM updates the network weights recursively. This network weight updating algorithm lacks flexibility to adjust parameters according to actual conditions, and at the same time, it is easy to increase unnecessary computation time [28]. In order to solve the problems existing in OS-ELM effectively, an improved OS-ELM named as adaptive OS-ELM is proposed. This adaptive OS-ELM can overcome the sample data updating disadvantages of OS-ELM.

On the other hand, how to determine the suitable parameters of the hidden layer nodes in OS-ELM algorithm becomes important [48]. As a novel intelligent optimization algorithm, artificial bee colony (ABC) algorithm can be introduced to solve this problem. In order to overcome the disadvantages of adaptive OS-ELM, this paper proposes an improved ABC algorithm to optimize the parameters of the hidden layer nodes in adaptive OS-ELM. This improved ABC algorithm has faster optimization speed and better performance. The proposed improved ABC algorithm can improve the efficiency and performance of adaptive OS-ELM. The convergence analysis of the proposed prediction approach is given. The actual collected short-term wind speed is chosen as the simulation object. The effective of proposed prediction method is verified. The simulation results show that the proposed method has better prediction effect.

The main contents of each part are as follows. Section 2 gives the literature review. Section 3 introduces the methodologies include adaptive OS-ELM and improved ABC algorithm. Section 4 introduces the detailed implementation of the proposed prediction method. Convergence analysis of prediction method is also given. The simulation results are provided in Section 5, and the validity of the

proposed prediction method is given. The conclusions and prospects of the paper are summarized in Section 6.

2. Literature review

2.1   Review of OS-ELM

When a given training sample is used to get the ELM model, if a new sample is produced, the training must be retrained, which leads to a great increase in the training time of the ELM model [25]. In order to solve this problem, OS-ELM algorithm uses the recursive idea to update the output weights online with new samples. Through some Benchmark data sets, the author pointed out that OS-ELM has achieved excellent performance in classification and regression [26]. At the same time, many scholars have also improved the OS-ELM algorithm. The authors combined OS-ELM with adaptive forgetting factor and bootstrap, the performance is tested by Benchmark data sets [29]. In [20], the author proposed an efficient parallel method for batched OS-ELM (BPOS-ELM), the experimental results show that the accuracy of BPOS-ELM has higher training efficiencies. The authors presented an online sequential reduced kernel ELM (OS-RKELM). Experimental results obtained indicate that they proposed OS-RKELM showcases improved prediction accuracy and efficiency [5]. The authors proposed an ensemble of OS-ELM (EOS-ELM) with binary Jaya. The application results on the IEEE 39-bus system and a real provincial system show that EOS-ELM has superior computation speed and prediction accuracy [27]. The implementation of the OS-ELM is described as follows.

For a given training set $D = \{(\mathbf{x}_i, \mathbf{t}_i) \,|\, \mathbf{x}_i \in \mathbf{R}^n, \mathbf{t}_i \in \mathbf{R}^m, i = 1, \ldots, N\}$, activation function $f(x)$, the number of hidden nodes L. ELM regression model can be expressed as

$$
\begin{aligned}
\sum_{i=1}^{L} \beta_i f(\mathbf{a}_i \mathbf{x}_1 + b_i) &= t_1 \\
\sum_{i=1}^{L} \beta_i f(\mathbf{a}_i \mathbf{x}_2 + b_i) &= t_2 \\
&\vdots \\
\sum_{i=1}^{L} \beta_i f(\mathbf{a}_i \mathbf{x}_k + b_i) &= t_k
\end{aligned}
, \tag{1}
$$

where $\mathbf{a}_i, i = 1, \ldots, L$ is the output weights, $b_i, i = 1, \ldots, L$ is bias, k is the numbers of the sample. Eq. (1) is rewritten as

$$
\mathbf{T}_k = \mathbf{H}_k \beta_k, \tag{2}
$$

where $\mathbf{H}_k$ is a neuron matrix and can be represented as

$$
\mathbf{H}_k = \begin{bmatrix}
f(\mathbf{a}_1\mathbf{x}_1 + b_1) & \cdots & f(\mathbf{a}_L\mathbf{x}_1 + b_L) \\
f(\mathbf{a}_1\mathbf{x}_2 + b_1) & \cdots & f(\mathbf{a}_L\mathbf{x}_2 + b_L) \\
\vdots & \ddots & \vdots \\
f(\mathbf{a}_1\mathbf{x}_k + b_1) & \cdots & f(\mathbf{a}_L\mathbf{x}_k + b_L)
\end{bmatrix}. \tag{3}
$$

β_k is the output weights and can be expressed as

$$\beta_k = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_k \end{bmatrix}^{\mathrm{T}}. \tag{4}$$

$\mathbf{T}_k$ is the output weights and can be expressed as

$$\mathbf{T}_k = \begin{bmatrix} t_1 & t_2 & \cdots & t_k \end{bmatrix}^{\mathrm{T}}. \tag{5}$$

The output weights can be obtained by solving Eq. (2).

$$\beta_k = \left(\mathbf{H}_k^{\mathrm{T}}\mathbf{H}_k\right)^{-1}\mathbf{H}_k^{\mathrm{T}}\mathbf{T}_k. \tag{6}$$

Therefore, the prediction model based on ELM can be obtained after training

$$t = \sum_{i=1}^{L} \beta_i f\left(\mathbf{a}_i\mathbf{x} + b_i\right). \tag{7}$$

Based on ELM, OS-ELM calculates the initial output weight under the condition of $k \geq L$.

$$\beta_k = \mathbf{P}_k\mathbf{H}_k^{\mathrm{T}}\mathbf{T}_k, \tag{8}$$

where $\mathbf{P}_k = (\mathbf{H}_k^{\mathrm{T}}\mathbf{H}_k)^{-1}$.

New training sample (x_{k+1}, t_{k+1}) is added into sample set, the current $\mathbf{P}_{k+1}$ and β_{k+1} is calculated according to the following equations.

$$\mathbf{P}_{k+1} = \mathbf{P}_k - \frac{\mathbf{P}_k\mathbf{h}_{k+1}^{\mathrm{T}}\mathbf{h}_{k+1}\mathbf{P}_k}{1 + \mathbf{h}_{k+1}\mathbf{P}_k\mathbf{h}_{k+1}^{\mathrm{T}}}, \tag{9}$$

$$\beta_{k+1} = \beta_k + \mathbf{P}_{k+1}\mathbf{h}_{k+1}^{\mathrm{T}}(t_{k+1} - \mathbf{h}_{k+1}\beta_k). \tag{10}$$

However, OS-ELM considers that the value of the new and old training samples is equal, and the equal weight of the training sample fails to highlight the role of the new training sample [3, 14]. Moreover, as soon as the new training samples are obtained, ELM updates the weights of the network. In order to solve this problem in ELM, this paper proposes an adaptive OS-ELM with more effective sample updating mechanism. This paper considers that the new sample should be added to the training set after the initial network weights are calculated, and the corresponding network weights can be obtained on the basis of the initial network weights. At the same time, the new and old samples are given different weights. The effect of new training samples on the algorithm is further enhanced, which can further improve the regression prediction ability of OS-ELM. The detailed description of the proposed algorithm will be given in the Section 3.1.

2.2 Review of ABC algorithm

The ABC algorithm is an intelligent optimization algorithm, which comes from the behaviour of honey bees [22]. Compared with the genetic algorithm, the differential evolution algorithm and the particle swarm optimization algorithm, the ABC algorithm is very competitive [13,15]. The nectar source of ABC algorithm is abstracted

as a point in the solution space. The quality of nectar source $(i = 1, 2, \ldots, SN)$ is chosen as the fitness of the solution. Assuming the dimension of the problem to be solved is D. For the $t-th$ iterations, the location of nectar source can be expressed as $\mathbf{X}_i^t = [x_{i1}^t \ x_{i2}^t \ x_{iD}^t]$, $x_{iD} \in (L_d, U_d)$. Where, L_d is the lower bound of the search space, U_d is the upper bound of the search space. The location of nectar source is randomly generated in the search space according to Eq. (11).

$$x_{id} = L_d + rand(0, 1)(U_d - L_d). \tag{11}$$

At the start of the search phase, a new nectar source around nectar source is generated by employed foragers according to Eq. 12.

$$v_{id} = x_{id} + \varphi_{id}(x_{id} - x_{jd}), \tag{12}$$

where $j \in \{1, 2, \cdots, SN\}$, $j \neq i$. It represents a random selection of nectar source which is different with i. $\varphi \in [-1, 1]$. When fitness of new nectar source is better than $\mathbf{X}_i$, greedy algorithm is used to carry out $\mathbf{V}_i$ replace $\mathbf{X}_i$. Otherwise $\mathbf{X}_i$ is remained. Then, onlooker bees share information according to nectar source of scout bees. Following probability is as follows [10].

$$p_i = \frac{\text{fit}_i}{\sum\limits_{i=1}^{SN} \text{fit}_i}. \tag{13}$$

That is means onlooker bees generate a random number belong to [0, 1] and compare it with p_i. If this random number is smaller than p_i, then generate a new nectar source according to Eq. (12). In the searching process, the nectar source $\mathbf{X}_i$ can not find a better new nectar source after several iterations, then $\mathbf{X}_i$ is abandoned, the corresponding employed bees change to scout bees. A new nectar source will be random generated according to Eq. (14).

$$x_i^{t+1} = \begin{cases} L_d + rand(0, 1)(U_d - L_d), \text{trail} \geq \text{limit} \\ x_i^t, \text{trail} < \text{limit} \end{cases}. \tag{14}$$

However, ABC algorithm has some disadvantages [21]. In the Eq. (12), φ_{id} is a random number, x_{jd} is a random selection among the neighborhood individuals. Therefore, the new global random search ability obtained by Eq. (12) is very strong. But the solution may be a better solution or a worse solution, the local search ability of the neighborhood in Eq. (12) is inadequate.

2.3 Review of short-term wind speed prediction

As a typical time series with nonlinearity, randomness and non-stationary, short-term wind speed is often used to test the prediction performance of prediction methods. Short-term wind speed prediction refers to the prediction of wind speed in the next 1 to 48 hours [40]. Accurate short-term wind speed prediction has important theoretical significance and practical application value for wind power industry [43, 44]. At present, the prediction method of short-term wind speed is mainly based on historical data. These prediction methods usually use historical

data, through some linear models include autoregressive moving average model (ARMA) [9, 34], autoregressive integrated moving average model (ARIMA) [2]. The nonlinear model include SVM [8,12], LSSVM [36,39], artificial neural network (Elman neural network [44,45], echo state network [38], fuzzy neural network [6,30], RBF neural network [4,23], and etc to predict short-term wind speed.

The results of some related literatures indicate that the short-term wind speed has strong nonlinearity [1, 24], so the nonlinear model is more suitable for short-term wind speed prediction. But the key parameters of SVM and LSSVM have no definite determination method. The inappropriate model parameters will greatly affect the regression prediction performance of SVM and LSSVM. The artificial neural network has some problems, such as the network structure is difficult to be determined, and the algorithm is easy to fall into the local optimal value. Therefore, how to improve the accuracy of short-term wind speed prediction is still a hot research topic. In this paper, actual collected short-term wind speed is used as the simulation object to verify the performance of the prediction method.

3. Methodology

3.1 Adaptive OS-ELM

The proposed adaptive OS-ELM algorithm can be described as follows. Suppose that β_k in Eq. (6) is calculated by sampling sample $(\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), \ldots, (\mathbf{x}_k, \mathbf{t}_k)$. When new sample $(\mathbf{x}_{k+1}, \mathbf{t}_{k+1})$ is added into set, then β_{k+1} can be expressed as

$$
\begin{aligned}
\beta_{k+1} &= \left(\begin{bmatrix} \mathbf{H}_k \\ \mathbf{h}_{k+1} \end{bmatrix} \begin{bmatrix} \mathbf{H}_k \\ \mathbf{h}_{k+1} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{H}_k \\ \mathbf{h}_{k+1} \end{bmatrix} \begin{bmatrix} \mathbf{T}_k \\ t_{k+1} \end{bmatrix} = \\
&= (\mathbf{H}_k^{\mathrm{T}} \mathbf{H}_k + h_{k+1}^{\mathrm{T}} h_{k+1})^{-1} (\mathbf{H}_k^{\mathrm{T}} \mathbf{T}_k + \mathbf{h}_{k+1}^{\mathrm{T}} t_{k+1}),
\end{aligned} \tag{15}
$$

where $\mathbf{h}_{k+1} = \begin{bmatrix} f(\mathbf{a}_1 \mathbf{x}_{k+1} + b_1) & f(\mathbf{a}_2 \mathbf{x}_{k+1} + b_2) & \cdots & f(\mathbf{a}_L \mathbf{x}_{k+1} + b_L) \end{bmatrix}$.
$\mathbf{H}_k^{\mathrm{T}}$ and $\mathbf{H}_k$ are given weights, the above Eq. (15) can be rewritten as

$$
\beta_{k+1} = (\mu \mathbf{H}_k^{\mathrm{T}} \mathbf{H}_k + \mathbf{h}_{k+1})^{-1} (\mu \mathbf{H}_k^{\mathrm{T}} \mathbf{T}_k + \mathbf{h}_{k+1}^{\mathrm{T}} t_{k+1}), \tag{16}
$$

where μ, $0 < \mu < 1$ is weight coefficient. Let

$$
\mathbf{P}_{k+1} = (\mu \mathbf{H}_k^{\mathrm{T}} \mathbf{H}_k + \mathbf{h}_{k+1}^{\mathrm{T}} \mathbf{h}_{k+1})^{-1}. \tag{17}
$$

The inverse of Eq. (17) can be obtained

$$
\mathbf{P}_{k+1}^{-1} = \mu \mathbf{P}_k^{-1} + \mathbf{h}_{k+1}^{\mathrm{T}} \mathbf{h}_{k+1}. \tag{18}
$$

Eq. (18) is substituted into Eq. (16), the next can be obtained.

$$
\begin{aligned}
\beta_{k+1} &= \mathbf{P}_{k+1} (\mu \mathbf{H}_k^{\mathrm{T}} \mathbf{T}_k + \mathbf{h}_{k+1}^{\mathrm{T}} t_{k+1}) = \mathbf{P}_{k+1} (\mu \mathbf{P}_k^{-1} \beta_k + \mathbf{h}_{k+1}^{\mathrm{T}} t_{k+1}) = \\
&= \mathbf{P}_{k+1} ((\mathbf{P}_{k+1}^{-1} - \mathbf{h}_{k+1}^{\mathrm{T}} h_{k+1}) \beta_k + \mathbf{h}_{k+1}^{\mathrm{T}} t_{k+1}) = \\
&= \beta_k + \mathbf{P}_{k+1} \mathbf{h}_{k+1}^{\mathrm{T}} (t_{k+1} - \mathbf{h}_{k+1} \beta_k).
\end{aligned} \tag{19}
$$

When a new sample x_{k+1} is obtained, it is necessary to judge the change trend of the error. When the error value is greater than a threshold value ε, $\mathbf{P}_k$ is updated,

otherwise $\mathbf{P}_k$ remains unchanged. The update mechanism is shown in the following formula.

$$\mathbf{P}_{k+1} = \begin{cases} (\mu \mathbf{H}_k^{\mathrm{T}} \mathbf{H}_k + \mathbf{h}_{k+1}^{\mathrm{T}} \mathbf{h}_{k+1})^{-1} & (E_N > \varepsilon) \\ \mathbf{P}_k & (E_N \leq \varepsilon) \end{cases}, \tag{20}$$

where $E_N = \sqrt{\sum_{i=1}^{N}(x_i - \widehat{x}_i)^2/N}$, $\widehat{x}_i$ is the predictive value of x_i.

The unique updating mechanism of the proposed adaptive OS-ELM makes it more suitable for the on-line prediction of time series for the speediness and accuracy of the prediction.

3.2 Improved ABC algorithm

This paper proposes an improved ABC algorithm to optimize optimal parameters of hidden layers of adaptive OS-ELM. The adaptive OS-ELM prediction model is constructed with the obtained optimal parameters. In order to improve the local search ability of the algorithm, the random step size φ_{id} is improved, so that it can be adjusted adaptively with the change of fitness. The new location update method is as follows.

$$v_{id} = x_{id} + R_{id}(x_{id} - x_{jd}) + c_{id}(x_{bestd} - x_{id}), \tag{21}$$

$$R_{id} = \begin{cases} r_{id}(1 - \frac{f_j - f_i}{f_i - f_{best}}), & f_j \neq f_{best} \\ \varphi_{id}, & f_j = f_{best} \end{cases}, \tag{22}$$

$$c_{id} = c_{\min} + (c_{\max} - c_{\min})[\frac{2}{1 + \exp(-\alpha(\frac{t}{MCN})^{\beta})} - 1], \tag{23}$$

where r_{id} has a random value of $+1$ or -1. φ_{id} is a random number between $[-1, 1]$. f_i and f_j are the objective function of the optimization problem. t is the current iteration number. MCN is the maximum number of iterations. $c_{\max}$, $c_{\min}$, α and β are constant value. x_{bestd} is a d dimensional component of the current optimal solution. Compared with the random step size φ_{id}, r_{id} has a wider range of values. The absolute value of R_{id} may be greater than 1. Thus, in the early iterations, the larger step size is beneficial to enlarge the search space of the algorithm. When f_i is close to f_{best}, R_{id} is more close to 0. In this case, the smaller step size helps the algorithm can quickly find the optimal solution in the local search. R_{id} plays a guiding role in the search trend of the nectar source. In the early phase of iterations, the parameter c_{id} should be smaller, so as to reduce the global optimum and improve the global searching ability. In the later phase of iterations, c_{id} should keep a larger value, so that the algorithm can converge rapidly to the global optimum.

In order to verify the performance of the improved ABC algorithm, the Sphere function is chosen as testing function and compared with the standard ABC algorithm. The Sphere function is shown as in the following formula

$$f = \sum_{j=1}^{N} x_j^2, \tag{24}$$

where $x_j \in [-50, 50]$, $N = 4$.

The number of nectar source SN is 50, maximum number of iterations MCN is 100, the maximum number of nectar source mining $limit$ is 50, c_{max} is 1, c_{min} is 0, α is 50, β is 5. To eliminate the influence of random, all the algorithms are running for 20 times and the average value is chosen as the optimization results. Fig. 1 is the average fitness curve of two algorithms. It can be seen from Fig. 1, improved ABC algorithm has improved convergence accuracy, convergence speed and optimization results compared with standard ABC algorithm. Tab. I gives the comparison results of improved ABC and standard ABC, which includes the best fitness value, the average fitness value, and standard deviation. As can be seen from Tab. I, the best fitness, the success rate and standard deviation of improved ABC algorithm are better than standard ABC algorithm.
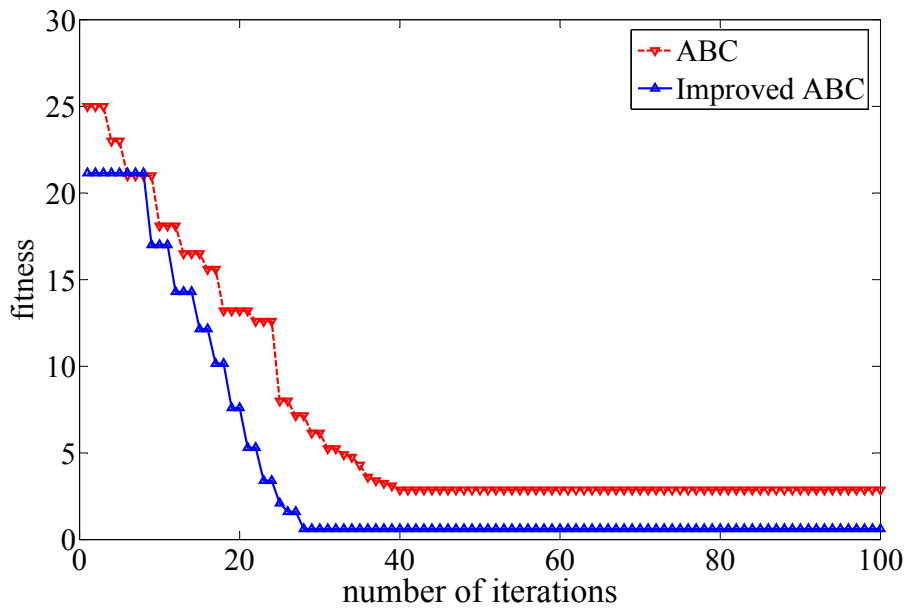
Fig. 1 *Average fitness curve of ABC and improved ABC algorithm.*

Algorithm	Average fitness	Best average	Standard deviation
Improved ABC	**1.3651**	**0.0871**	**0.0775**
ABC	5.4237	2.5266	1.3657

Tab. I *Simulation results of improved ABC and standard ABC algorithm.*

4. Adaptive OS-ELM based on improved ABC algorithm

4.1 The proposed model

The hidden layer parameters $\mathbf{a}^*, b^*$ in adaptive OS-ELM algorithm are chosen as optimal parameters to be optimized. In order to maintain consistency with the improved ABC algorithm, the next Eq. (25) is chosen as fitness function of adaptive OS-ELM prediction model based on improved ABC algorithm.

$$fit_k = \frac{1}{1 + f_k}, \tag{25}$$

where f_k is network training error of adaptive OS-ELM. The network training error E_k can be expressed as Eq. (26).

$$E_k = \|\mathbf{H}_k \beta_k - \mathbf{T}_k\|. \tag{26}$$

The implementation steps of hidden parameters of adaptive OS-ELM optimized by improved ABC algorithm can be described as the follows.

Step 1 The parameters of adaptive OS-ELM are initialized. These parameters include maximum number of hidden layer nodes L, activation function $f(x)$, embedding dimension of samples m, weight coefficient μ, error threshold ε, and etc.

Step 2 The initial N sample data is $x_1, x_2, \cdots x_N$ is transformed into training sample set $(\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), \cdots (\mathbf{x}_k, \mathbf{t}_k)$. $\mathbf{x}_i = \begin{bmatrix} x_i & x_{i+1} & \cdots & x_{i+m-1} \end{bmatrix}^{\mathrm{T}}$ is chosen as input, $t_i = x_{i+m}$ is chosen as output. Where $k = N - m > L$.

Step 3 The output weight is calculated according to Eq. (19).

Step 4 Training error E_k of adaptive OS-ELM network is calculated according to Eq. (26).

Step 5 Input weight vector $\mathbf{a}_k$ and bias b_k of hidden layer node parameters are optimized by improved ABC algorithm.

1. Parameters are initialized. That includes the number of nectar source – SN, the maximum number of iterations - MCN, the maximum number of nectar source mining – $limit$, $c_{\max}$, $c_{\min}$, α, β, etc. The values of the parameters to be optimized are given. Suppose that number of iterations, $t = 1$.

2. An employed forager is assigned for nectar source. The searching process is begun according to Eq. (12). A new nectar source $\mathbf{V}_i$ will be generated.

3. The fitness value E_k is calculated by samples data. The nectar source will be retained according to greedy algorithm.

199

4. The probability of nectar source be followed is calculated by Eq. (13). The onlooker bees search and retain the nectar source according to greedy algorithm.

5. The algorithm determines whether the nectar source should be discarded. If true, the onlooker bees are changed into scout bees. Otherwise, go to Step 5.7.

6. The scout bees will generate a new nectar source according to Eqs. (21) to (23).

7. Let $t = t + 1$. If the maximum numbers of iterations are satisfied, optimal parameters are output. Otherwise, go to Step 5.2 and continue to execution.

Step 6 $\mathbf{x}_{k+1} = \begin{bmatrix} x_{k-m+1} & x_{k-m+2} & \cdots & x_k \end{bmatrix}^{\mathrm{T}}$ is chosen as input, input vector $\mathbf{h}_{k+1}$ is calculated. Then, one step prediction value of x_{k+1} is obtained.

$$\widehat{x}_{k+1} = \mathbf{h}_{k+1}\beta_k. \tag{27}$$

Step 7 If the maximum prediction step is achieved, the prediction process ends. Otherwise, the matrix $\mathbf{P}_k$ is updated according to Eq. (17). Then, β_k is updated according to Eq. (19). Let $k = k+1$, $N = N+1$. Go to **Step 2**.

4.2 Convergence analysis of the prediction model

The following two lemmas are introduced.

Lemma 1 Given arbitrary $\varepsilon > 0$ and activation function. For q discrete samples $(\mathbf{x}_i, \mathbf{t}_i)$, $\mathbf{x} \in \mathbf{R}^n$, $t \in \mathbf{R}^m$, there is q hidden layer neurons and meets

$$||\mathbf{H}(\mathbf{a}, b, \mathbf{x})\widehat{\beta} - \mathbf{t}|| < \varepsilon, \tag{28}$$

where $\mathbf{a} = [a_1, a_2, \cdots, a_q]$, $\mathbf{b} = [b_1, b_2, \cdots, b_q]$, $\widehat{\beta} = \mathbf{P}\mathbf{H}^{\mathrm{T}}\mathbf{T}$.

Lemma 2 Given any bounded continuous or piecewise continuous activation function, for any continuous objective function, if

$$\widehat{\beta}_L = \mathbf{P}_L\mathbf{H}_L^{\mathrm{T}}\mathbf{T}_L, \tag{29}$$

then there are arbitrary output matrix $\mathbf{H}$ and SLFNs makes the following Eq. (30) to be satisfied. At the same time, network error $||E_L||$ decreases monotonically with the decrease of the number of neurons.

$$\lim_{L \to +\infty} ||E_L|| = \lim_{L \to +\infty} ||\mathbf{H}_L\widehat{\beta}_L - \mathbf{t}|| = 0. \tag{30}$$

Theorem 1 Given discrete sample $(\mathbf{x}_i, \mathbf{t}_i)$, $\mathbf{x} \in \mathbf{R}^n$, $t \in \mathbf{R}^m$, for arbitrary $\varepsilon > 0$. If the parameters of adaptive OS-ELM neural network are optimized by improved ABC algorithm, there must be a SLFNs that has q neurons make $||\mathbf{H}_q\widehat{\beta}_q - \mathbf{t}|| < \varepsilon$ be satisfied.

Proof. (1) If the current hidden layer node does not exceed the maximum allowable number of neurons. Suppose that the maximum number of iterations of the improved ABC algorithm is MCN, the number of hidden layer nodes in the network is L. If $L = 0$ and $k = 0$, the corresponding output error $||E_L(\mathbf{H}_k)|| > \varepsilon$. It is assumed that the improved ABC algorithm is iterated K times, then the corresponding output error $||E_L(\mathbf{H}_k)|| \leq ||E_L(\mathbf{H}_0)||$. At this time, the hidden layer node is increased and $L = 1$, there is $||E_1(\mathbf{H}_0)|| \leq ||E_0(\mathbf{H}_K)||$ according to [18]. Therefore, when $L = N$ and $k = K$, there is

$$||E_N(\mathbf{H}_K)|| \leq ||E_N(\mathbf{H}_0)|| \leq \cdots \leq ||E_1(\mathbf{H}_0)|| \leq ||E_0(\mathbf{H}_K)|| \leq ||E_0(\mathbf{H}_0)||. \quad (31)$$

According to [17], if exist $\sum_{L=1}^{q} L > N$ and $\mathbf{H}_q$ is invertible, there is

$$||E_0(\mathbf{H}_0)|| \geq \cdots \geq ||E_q(\mathbf{H}_K)|| = 0, \quad (32)$$

Therefore, there exists $q < N$ make $||\mathbf{H}_q \widehat{\beta}_q - f|| < \varepsilon$ to be satisfied.

(2) If the current hidden layer node is greater than or equal to the maximum allowable number of neurons. Suppose that the maximum number of neurons is $L_{\max}$. The number of iterations is $k = 1$, the corresponding network output error is $||E_{L_{\max}}(\mathbf{H}_1)|| > \varepsilon$. If improved ABC algorithm is iterated to K, there is

$$||E_{L_{\max}}(\mathbf{H}_K)|| \leq ||E_{L_{\max}}(\mathbf{H}_{K-1})|| \leq \cdots \leq ||E_{L_{\max}}(\mathbf{H}_1)||. \quad (33)$$

According to Lemma 1, there is a constant q, $L_{\max} < q < N$ make the corresponding network error $||E_q(\mathbf{H}_1)|| \leq \varepsilon$. At the same time, the number of iterations is increased to $k = K$, there is

$$||E_q(\mathbf{H}_1)|| \geq ||E_q(\mathbf{H}_2)|| \geq \cdots \geq ||E_q(\mathbf{H}_K)||, \quad (34)$$

Therefore, there exists $q < N$ make $||\mathbf{H}_q \widehat{\beta}_q - f|| < \varepsilon$ to be satisfied.

From the above proof process can be obtained, the proposed adaptive OS-ELM based on improved ABC algorithm is stable and convergent.

5. Simulation

In order to verify the predictive performance of the proposed prediction approach, the short-term wind speed data is used to validate the predictive ability. The 500 groups of short-term wind speed data were collected from 6 O'clock on April 1, 2017 to 24 O'clock on April 21, 2017, which were measured from a power plant located in Liaoning Province, China. The sampling period is 1 hour. The first 452 groups of data are used to train the model, and the latter 48 groups of data are used as the test set to verify the accuracy of the prediction model. The short-term wind speed data of the 500 groups are shown in Fig. 2.

The proposed adaptive OS-ELM based on improved ABC algorithm is compared with OS-ELM [26], EI-ELM [19], EM-ELM [11], and standard ELM [17]. The parameters of the proposed adaptive OS-ELM are as the follows: the data embedding dimension m is determined as 48, the number of neurons in the hidden
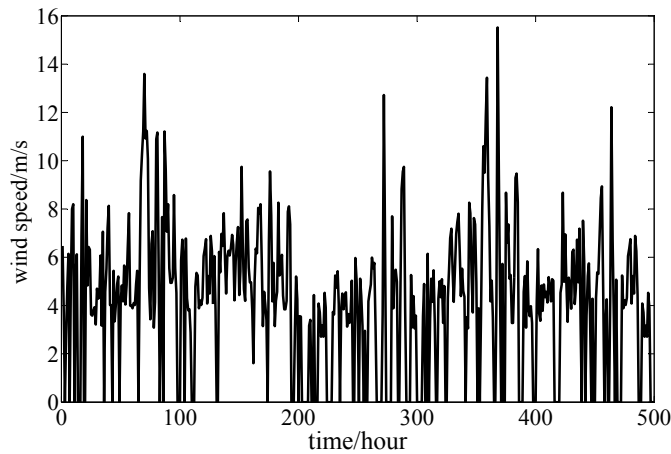
Fig. 2 *Short-term wind speed time series.*

layer L is set to 100. The weight coefficient μ is set as 0.95. The threshold value ε is chosen as 0.2. The parameters of improved ABC algorithm are as the follows: the number of nectar source SN is 20, maximum number of iterations MCN is 100, the maximum number of nectar source mining $limit$ is 30, c_{max} is 1, c_{min} is 0. The parameters of OS-ELM are as the follows: the data embedding dimension m is determined as 48, the number of neurons in the hidden layer L is set to 100. The hidden layer of EI-ELM and standard ELM are chosen as 100. In EM-ELM, the initial hidden layer node is 1, each adding 4 hidden nodes, the node limit is set to 100. The activation functions of all ELM algorithms are Sigmoid functions with λ is 1. The range of input weights and bias values are $[-1, 1]$. All inputs and outputs are normalized to $[-1, 1]$. Program operating environment are CPU (i7-4770, 3.4 GHz), memory (8 GB), simulation software is Matlab 2010b. 20 simulation experiments are carried out under the same conditions. The average value of training time, prediction time (48 steps), training error, and test error in 10 experiments are chosen as evaluation indexes. The training error and test error are judged by the root mean square error between the actual value and prediction value. Tab. II shows the operating results of several prediction models.

It can be observed from Tab. II, after the initial training stage, ELM, EM-ELM, and EI-ELM does not use the new training samples to update the prediction model, so with the increase of the prediction steps, the sample used for training is gradually far away from the current time, thus the ability to track the dynamic change characteristics of the short-term wind speed time series is gradually weakened, the accuracy of the prediction will also be gradually reduced. Because of the continuous use of the new training samples containing the current time information, the prediction errors of adaptive OS-ELM and OS-ELM are all smaller than the prediction errors of ELM, EM-ELM, and EI-ELM. In addition, since the new training sample is closest to the current time, it is more valuable than the old training sample, which is closer to the real short-term wind speed time series of the

Prediction models	Training time(s)	Prediction time (s)	Training error (m/s)	Test error (m/s)
Adaptive OS-ELM based on improved ABC algorithm	**1.2284**	**0.1341**	**1.8827**	**0.2308**
OS-ELM	1.6851	0.1825	2.0247	0.2762
EM-ELM	2.1547	0.2435	2.6874	0.3124
EI-ELM	2.8235	0.2998	2.7728	0.3566
ELM	3.4425	0.3751	3.4015	0.4495

Tab. II *The simulation results of several ELM algorithms.*

current time. Compared with the fairness principle of OS-ELM algorithm treat new and old samples, adaptive OS-ELM pays more attention to the contribution of new training samples, so that the updated prediction model has more information from new training samples. Therefore, adaptive OS-ELM is closer to the current real short-term wind speed time series than OS-ELM, and its prediction error is obviously less than OS-ELM. On the other hand, the improved ABC algorithm is introduced to optimize the hidden parameters of adaptive OS-ELM. The optimized parameters further improve the prediction performance of the adaptive OS-ELM.

Fig. 3 is the comparison of the predictive values and the actual values of the 48 groups short-term wind speed in test set in one simulation by using 5 ELM models. From this graph, we can observe that the proposed method is superior to other 4 kinds of ELM algorithms. The proposed prediction method has better regression prediction ability for short-term wind speed time series.
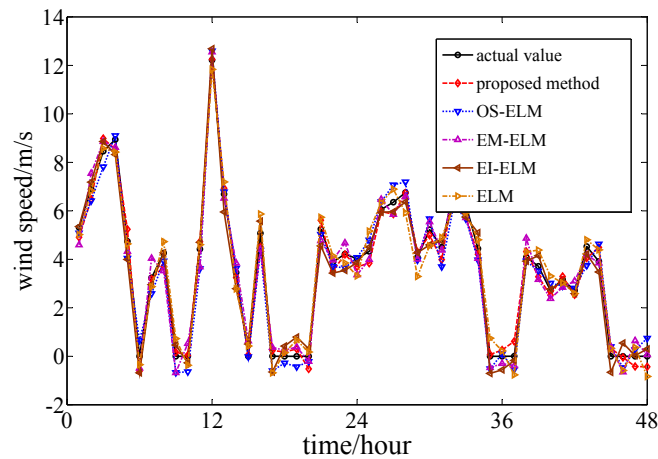
Fig. 3 *Comparison of short-term wind speed prediction and actual value of five ELM models.*

In order to further compare the predictive effects, the proposed prediction method is compared with ARIMA [2], SVM [12], LSSVM [36], and RBF neural network [23], the simulation results are shown in Fig. 4. Fig. 5 is the predictive error histogram distribution of the prediction models mentioned in this paper. Fig. 6 is the absolute predictive error of the prediction models mentioned in this paper.
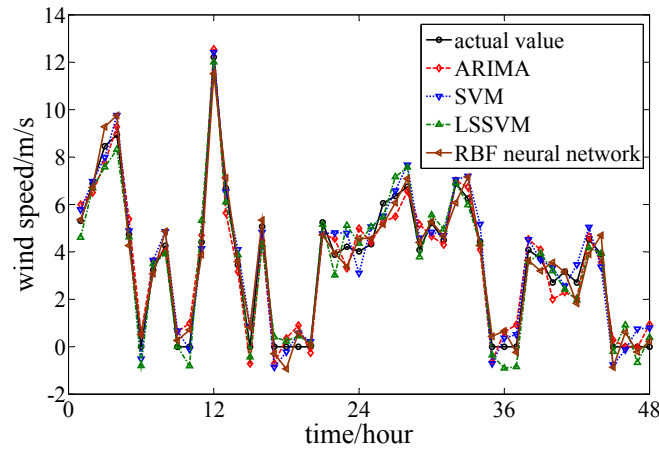

Fig. 4 *Comparison of the short-term wind speed prediction and actual value of other prediction models.*
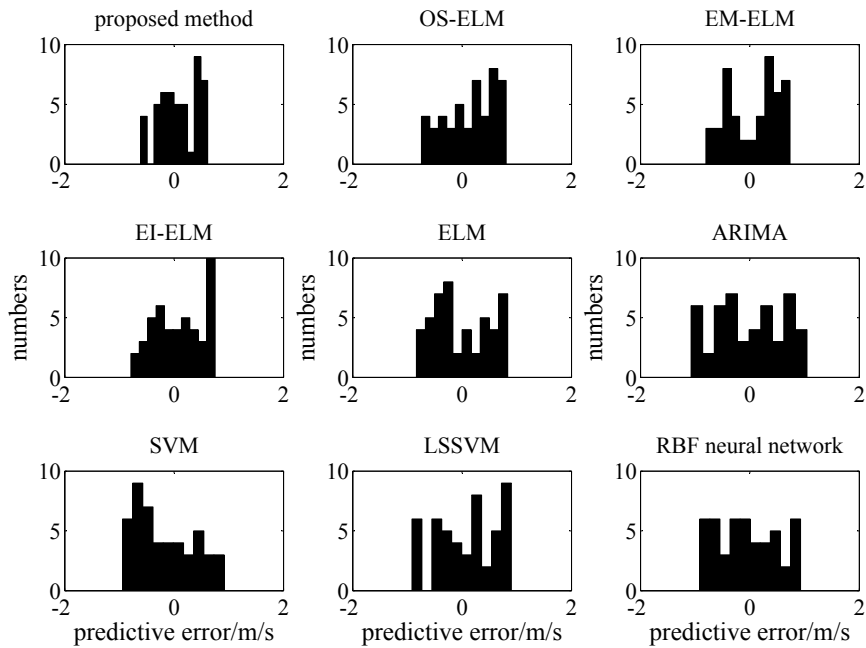
Fig. 5 *Predictive error histogram distribution of prediction models.*

From the results of Fig. 3 to Fig. 6, we can see that the proposed prediction model is better than other models in prediction accuracy and prediction error. Therefore, the prediction value of short-term wind speed of proposed prediction method can reflect the actual value of network traffic more accurately.
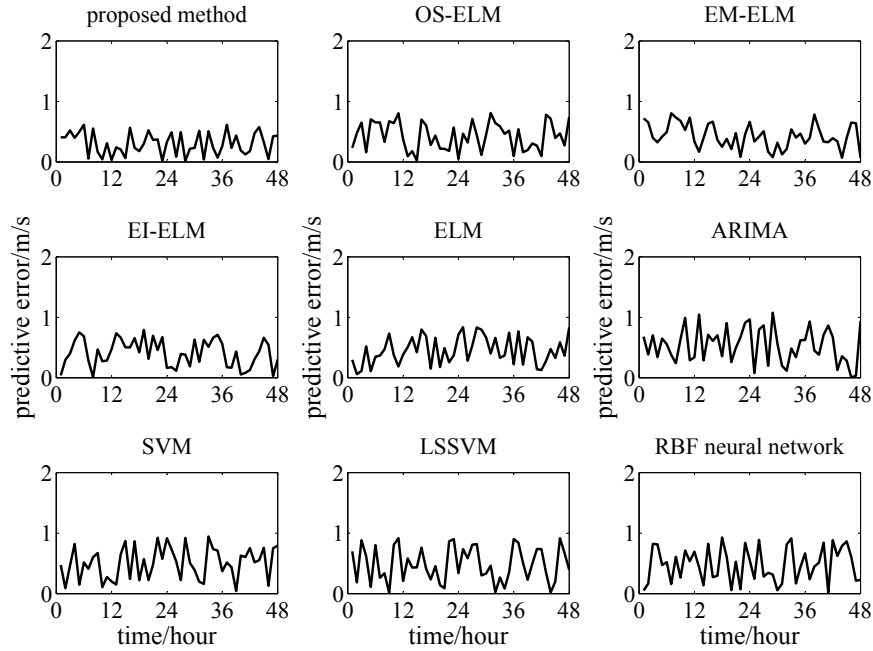
Fig. 6 *Absolute error distribution of prediction models.*

In this paper, we introduce the following four kinds of performance indicators to measure the prediction accuracy of the prediction model.

1. RMSE (root mean square error)

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (w(k) - \widehat{w}(k))^2}, \tag{35}$$

2. MAE (mean absolute error)

$$\mathrm{MAE} = \frac{1}{N} \sum_{k=1}^{N} |w(k) - \widehat{w}(k)|, \tag{36}$$

3. MAPE (mean absolute percentage error)

$$\mathrm{MAPE} = \frac{1}{N} \sum_{k=1}^{N} |w(k) - \widehat{w}(k)| \times 100/w(k), \tag{37}$$

4. Reliability

$$R^{(1-a)} = [\frac{\xi^{(1-a)}}{N} - (1-a)] \times 100\%, \qquad (38)$$

where N is the number of samples, $w(k)$ is actual value of wind speed, $\widehat{w}(k)$ is predictive value of wind speed, $\xi^{(1-a)}$ is the numbers of confidence intervals in which the actual value falls under the confidence level $1 - \alpha$.

Tab. III shows the comparison of RMSE, MAE and MAPE performance indicators of these prediction models. The results in Tab. III also show that the prediction model in this paper is superior to other prediction models in the performance indicators.

Method	RMSE	MAE	MAPE
Proposed method	**0.3549**	**0.3041**	**0.1592**
OS-ELM	0.4838	0.4262	0.1993
EM-ELM	0.4685	0.4242	0.2816
EI-ELM	0.4662	0.4557	0.2978
ELM	0.5095	0.4091	0.2719
ARIMA	0.6093	0.5368	0.2987
SVM	0.5698	0.5071	0.2852
LSSVM	0.5678	0.4885	0.2740
RBF neural network	0.5942	0.4773	0.3297

Tab. III *Comparison of performance indicators.*

Fig. 7 is the reliability and confidence distribution of the prediction models mentioned in this paper. It can be seen from this graph that the prediction model in this paper has higher reliability under the same confidence level. It can be known that the reliability of the proposed prediction model is better than other prediction models.
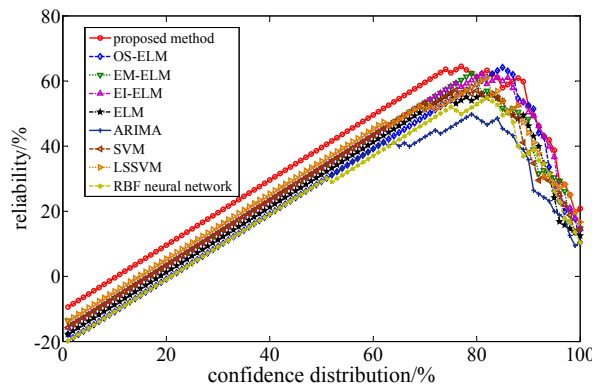
Fig. 7 *Reliability and confidence distribution of prediction models.*

In order to test the statistically significant of the prediction model, Wilcoxon Sign-Rank [16] test and Ranksum test [35] is introduced. Tab. IV gives the Wilcoxon Sign-Rank and Ranksum test results between predictive value of each prediction model and actual value with significant level is 0.05.

Method	P-value (Wilcoxon Sign-Rank test)	P-value (Wilcoxon Ranksum test)
Proposed method	**0.8939**	**0.9737**
OS-ELM	0.8696	0.8574
EM-ELM	0.6518	0.8002
EI-ELM	0.6574	0.8804
ELM	0.6522	0.7496
ARIMA	0.5267	0.7330
SVM	0.5741	0.6256
LSSVM	0.6117	0.6357
RBF neural network	0.5627	0.4884

Tab. IV *The Wilcoxon Sign-Rank and Ranksum test results.*

It can be observed from Tab. IV, the P-value of Wilcoxon Sign-Rank test of the proposed method is bigger than the other methods. It means that compared with other prediction methods, the median difference between the short-term wind speed prediction value and the actual value is even less significant. At the same time, the P-value of Wilcoxon Ranksum test of the proposed method is also bigger than the other methods. It means that compared with other prediction methods, the probability of the average value of the predicted value and the actual value of the proposed method is even greater. Therefore, the predictive value of the proposed prediction method is more consistent with the trend of short-term wind speed time series. In summary, the two test results in Tab. IV show that the statistically significant difference between the predicted value of the prediction model and the actual value of the short-term wind speed series is not obvious, and it has a better prediction performance.

To prove the prediction performance, the Pearson's test [42] is used to test prediction accuracy from the statistical perspective. Pearson's test can measure the association strength between the actual value and the prediction value. The results of association strength based on Pearson's test are performed out to further verify the superiority of the proposed prediction model compared with other models. If Pearson's correlation coefficient is equal to 1, it indicates that the actual value and the prediction value have a linear relationship. On the other hand, if Pearson's correlation coefficient is equal to 0, there is no relationship between actual value and prediction value. The results of Pearson's test are given in Tab. V. From Tab. V, it can be observed that the results of Pearson's test of the proposed prediction method are higher than those of the other prediction models. The results show that the association strength between the actual value and the prediction value of the proposed prediction method is stronger than the other methods.

Method	Pearson's correlation coefficient
Proposed method	**0.9928**
OS-ELM	0.9879
EM-ELM	0.9773
EI-ELM	0.9677
ELM	0.9545
ARIMA	0.9476
SVM	0.9315
LSSVM	0.9324
RBF neural network	0.9017

Tab. V *Results for the Pearson's test.*

In summary, from the point of view of the above modeling training results, curve fitting, predictive error distribution, performance indicators, reliability, Wilcoxon Sign-Rank test and Ranksum test, and Pearson's test, adaptive OS-ELM method based on improved ABC algorithm for short-term wind speed prediction has better prediction effect. The first main reason for the improvement of prediction performance is that adaptive OS-ELM highlights the role of the new training samples and updates the output weights selectively according to their generalization ability. The prediction performance of proposed adaptive OS-ELM is improved. The second main reason is that an improved ABC algorithm is used to optimize the parameters of adaptive OS-ELM. The adaptive OS-ELM constructed with the optimal parameters further improves the performance indicators.

6. Conclusion

1. The traditional ELM training algorithm is only suitable for off-line training. Once a new training sample is added to the training sample which is independent of the training sample, it is necessary to repeat the network training process. Although OS-ELM solves the online training problem of ELM, it considers that the value of the new and old training samples is equal, giving the same weight to the old and new samples, and failing to highlight the role of the new training samples. And, as long as OS-ELM gets new training samples, the network weights are recursively updated. This mechanical network weight updating model lacks flexibility according to the actual situation, and increases the unnecessary amount of calculation. This paper proposes an adaptive OS-ELM with more effective sample updating mechanism. The proposed approach considers that the new sample should be added to the training set after the initial network weights are calculated, and the corresponding network weights can be obtained on the basis of the initial network weights. At the same time, the new and old samples are given different weights. The effect of new training samples on the algorithm is further enhanced, which can further improve the regression prediction ability of OS-ELM.

2. On the other hand, how to determine the suitable parameters of the hidden layer nodes in OS-ELM algorithm becomes important. This paper proposes an improved ABC algorithm to optimize the parameters of the hidden layer nodes in adaptive OS-ELM. This improved ABC algorithm has faster optimization speed and better performance. The proposed improved ABC algorithm can improve the efficiency and performance of adaptive OS-ELM. The convergence analysis of the proposed prediction approach is given.

3. As a typical time series with nonlinearity, randomness and non-stationary, short-term wind speed is often used to test the prediction performance of prediction methods. The actual collected short-term wind speed time series is used as the research object and verify the prediction performance of the proposed method. The simulation results show that the proposed approach has higher prediction accuracy and reliability performance, meanwhile improve the performance indicators.

4. The number of nodes in the hidden layer of adaptive OS-ELM network is constant. Many neurons in the adaptive OS-ELM algorithm have little effect on the final output. Meanwhile, these useless neurons greatly increase the number of iterations of algorithm, and reduce the efficiency of the algorithm. The future work will take into account the structure of the OS-ELM, which will consider the hidden layer nodes not always fixed, and will eliminate some nodes that join the network but have little contribution to the network. At the same time, the short-term wind speed time series is used as the simulation object. In fact, the proposed prediction approach can also be used in the prediction of any time series such as sunspot number, river runoff, stock price, network traffic, wind power and etc. The future research of this paper is also to apply the prediction model to other time series, and further improve the prediction accuracy of the prediction method.

Acknowledgement

References

[1] BALOCH M.H., WANG J., KALOI G.S. Stability and nonlinear controller analysis of wind energy conversion system with random wind speed. *International Journal of Electrical Power & Energy Systems*. 2016, 79(11), pp. 75–83, doi: 10.1016/j.ijepes.2016.01.018.

[2] CADENAS E., RIVERA W., CAMPOS-AMEZCUA R. HEARD, C. Wind speed prediction using a univariate ARIMA model and a multivariate NARX Model. *Energies*. 2016, 9(2), 109, doi: 10.3390/en9020109.

[3] CAO W.P., GAO J.Z., MING Z., CAI S.B., SHAN Z.G. Fuzziness-based online sequential extreme learning machine for classification problems. *Soft Computing*. 2018, 22(1), pp. 3487–3494, doi: 10.1007/s00500-018-3021-4.

[4] CHANG C.W., LU H.J., CHANG Y.R., LEE Y.D. An improved neural network-based approach for short-term wind speed and power forecast. *Renewable Energy*. 2017, 105, pp. 301–311, doi: `10.1016/j.renene.2016.12.071`.

[5] DENG W.Y., ONG Y.S., TAN P.S., ZHENG Q.H. Online sequential reduced kernel extreme learning machine. *Neurocomputing*. 2016, 174, pp. 72–84, doi: `10.1016/j.neucom.2015.06.087`.

[6] DONG Q.L., SUN Y.H., LI, P.Z. A novel forecasting model based on a hybrid processing strategy and an optimized local linear fuzzy neural network to make wind power forecasting: a case study of wind farms in China. *Renewable Energy*. 2017, 102, pp. 241–257, doi: `10.1016/j.renene.2016.10.030`.

[7] DUAN M.X., LI K.L., LIAO X.K., LI K.Q. A parallel multiclassification algorithm for big data using an extreme learning machine. *IEEE Transactions on Neural Networks and Learning Systems*. 2018, 29(6), pp. 2337–2351, doi: `10.1109/TNNLS.2017.2654357`.

[8] DU J.L., LIU Y.Y., YU Y.N., YAN, W.L. A prediction of precipitation data based on support vector machine and particle swarm optimization (PSO-SVM) algorithms. *Algorithms*. 2017, 10(2), pp. 57, doi: `10.3390/a10020057`.

[9] ERDEM E., SHI J. ARMA based approaches for forecasting the tuple of wind speed and direction. *Applied Energy*. 2011, 88(4), pp. 1405–1414, doi: `10.1016/j.apenergy.2010.10.031`.

[10] EBRAHIMNEJAD A., TAVANA M., ALREZAAMIRI H. A novel artificial bee colony algorithm for shortest path problems with fuzzy arc weights. *Measurement*. 2016, 93, pp. 48-56, doi: `10.1016/j.measurement.2016.06.050`.

[11] FENG G.R., HUANG G.B., LIN Q.P., GAY R. Error minimized extreme learning machine with growth of hidden nodes and incremental learning. *IEEE Transactions on Neural Networks*. 2009, 20(8), pp. 1352–1357, doi: `10.1109/TNN.2009.2024147`.

[12] GANI A., MOHAMMADI K., SHAMSHIRBAND S., ALTAMEEM T.A., PETKOVIC D., CH S. A combined method to estimate wind speed distribution based on integrating the support vector machine with firefly algorithm. *Environmental Progress & Sustainable Energy*. 2016, 35, 3, 867–875, doi: `10.1002/ep.12262`.

[13] GONG D.W., HAN Y.Y., SUN J.Y. A novel hybrid multi-objective artificial bee colony algorithm for blocking lot-streaming flow shop scheduling problems. *Knowledge-Based Systems*. 2018, 148, pp. 115–130, doi: `10.1016/j.knosys.2018.02.029`.

[14] GRZEIDAK E., VARGAS J.A.R., ALFARO S.C.A. ELM with guaranteed performance for online approximation of dynamical systems. *Nonlinear Dynamics*. 2018, 91(3), pp. 1587–1603, doi: `10.1007/s11071-017-3966-3`.

[15] HEMANTH D.J., ANITHA J. A pattern-based artificial bee colony algorithm for motion estimation in video compression techniques. *Circuits, Systems, and Signal Processing*. 2018, 37(4), pp. 1609–1624, doi: `10.1007/s00034-017-0613-7`.

[16] HERNANDEZ W., MALDONADO-CORREA J.L. Power performance verification of a wind turbine by using the Wilcoxon Signed-Rank test. *IEEE Transactions on Energy Conversion*. 2018, 32(1), pp. 394–396, doi: `10.1109/TEC.2016.2612889`.

[17] HUANG G.B., ZHU Q.Y., SIEW C.K. Extreme learning machine: theory and applications. *Neurocomputing*. 2006, 70(1-3), pp. 489–501, doi: `10.1016/j.neucom.2005.12.126`.

[18] HUANG G.B., CHEN L., SIEW C.K. Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Transactions on Neural Network*. 2006, 17(4), pp. 879–892, doi: `10.1109/TNN.2006.875977`.

[19] HUANG G.B., CHEN L. Enhanced random search based incremental extreme machine. *Neurocomputing*. 2008, 71, pp. 3460–3468, doi: `10.1016/j.neucom.2007.10.008`.

[20] HUANG S., WANG B.T., CHEN Y.M., WANG G.R., YU G. An efficient parallel method for batched OS-ELM training using MapReduce. *Memetic Computing*. 2017, 9(2), pp. 183–197, doi: `10.1007/s12293-016-0190-5`.

[21] KARABOGA D., GORKEMLI B., OZTURK C., KARABOGA N. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*. 2014, 42(1), pp. 21–57, doi: `10.1007/s10462-012-9328-0`.

[22] KARABOGA D., BASTURK B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization*. 2007, 39(3), pp. 459–471, doi: 10.1007/s10898-007-9149-x.

[23] KIRBAS I., KEREM A. Short-term wind speed prediction based on artificial neural network models. *Measurement & Control*. 2016, 49(6), pp. 183–190, doi: 10.1177/0020294016656891.

[24] LI C.S., XiIAO Z.G., XIA X., ZOU W., ZHANG C. A hybrid model based on synchronous optimisation for multi-step short-term wind speed forecasting. *Applied Energy*. 2018, 215, pp. 131–144, doi: 10.1016/j.apenergy.2018.01.094.

[25] LI K., HAN Y., WANG T. A novel prediction method for down-hole working conditions of the beam pumping unit based on 8-directions chain codes and online sequential extreme learning machine. Journal of Petroleum Science and Engineering. 2018, 160, pp. 285–301, doi: 10.1016/j.petrol.2017.10.052.

[26] LIANG N.Y., HUANG G.B., SARATCHANDRAN P., SUNDARARAJAN N. A fast and accurate online sequential learning algorithm for feedforward networks. *IEEE Transactions on Neural Networks*. 2006, 17(6), pp. 1411–1423, doi: 10.1109/TNN.2006.880583.

[27] LI Y., YANG Z. Application of EOS-ELM with binary Jaya-based feature selection to real-time transient stability assessment using PMU data. *IEEE ACCESS*. 2017, 5, pp. 23092–23101, doi: 10.1109/ACCESS.2017.2765626.

[28] LI Y.C., QIU R.X., JING S.T. Intrusion detection system using online sequence extreme learning machine (OS-ELM) in advanced metering infrastructure of smart grid. *Plos One*. 2018, 13(2), pp. e0192216, doi: 10.1371/journal.pone.0192216.

[29] LIU J.Z. Adaptive forgetting factor OS-ELM and bootstrap for time series prediction. *International Journal of Modeling, Simulation, and Scientific Computing*. 2017, 8(3), pp. 1750029, doi: 10.1142/S1793962317500295.

[30] MA X.J., JIN Y., DONG Q.L. A generalized dynamic fuzzy neural network based on singular spectrum analysis optimized by brain storm optimization for short-term wind speed forecasting. *Applied Soft Computing*. 2017, 54, pp. 296–312, doi: 10.1016/j.asoc.2017.01.033.

[31] MOHAMED K.S., WU W., Liu Y. A modified higher-order feed forward neural network with smoothing regularization. *Neural Network World*. 2017, 27(6), pp. 577–592, doi: 10.14311/NNW.2017.27.032.

[32] NIU W.J., FENG Z.K., CHENG C.T., ZHOU J.Z. Forecasting daily runoff by extreme learning machine based on quantum-behaved particle swarm optimization. *Journal of Hydrologic Engineering*. 2018, 23(3), pp. 04018002–1, doi: 10.1061/(ASCE)HE.1943-5584.0001625.

[33] POPOOLA S.I., MISRA S., ATAYERO A.A. Outdoor path loss predictions based on extreme learning machine. *Wireless Personal Communications*. 2018, 99(1), pp. 441–460, doi: 10.1007/s11277-017-5119-x.

[34] RASOULI M., CHEN Y., BASU A., KUKREJA S.L., THAKOR N.V. An extreme learning machine-based neuromorphic tactile sensing system for texture recognition. *IEEE Transactions on Biomedical Circuits and Systems*. 2018, 12(2), pp. 313–325, doi: 10.1109/TBCAS.2018.2805721.

[35] REDMOND D.P., YEONG S.C., CHASE J.G. The effect of respiratory manoeuvres for patient-specific respiratory mechanics monitoring. *IFAC - Papers Online*, 2015, 48(20), pp. 135–140, doi: 10.1016/j.ifacol.2015.10.128.

[36] REN H.J., LEI X., ZHANG P. A study of wind speed prediction based on particle swarm algorithm to optimize the parameters of sparse least squares support vector. *International Journal of Simulation: Systems, Science and Technology*. 2016, 17(30), pp. 1.1–1.7, doi: 10.5013/IJSSST.a.17.30.01.

[37] SALAKEN, S.M., KHOSRAVI A., NGUYEN T., NAHAVANDI S. Extreme learning machine based transfer learning algorithms: A survey. *Neurocomputing*. 2017, 267, pp. 516–524, doi: 10.1016/j.neucom.2017.06.037.

[38] SUN W., Liu M.H. Wind speed forecasting using FEEMD echo state networks with RELM in Hebei, China. *Energy Conversion and Management*. 2016, 114, pp. 197–208, doi: 10.1016/j.enconman.2016.02.022.

[39] SUN W., LIU M.H., LIANG Y. Wind Speed Forecasting Based on FEEMD and LSSVM Optimized by the Bat Algorithm. *Energies*. 2015, 8(7), pp. 6585–6607, doi: `10.3390/en8076585`.

[40] TIAN Z.D., LI S.J., WANG Y.H., WANG X.D. Wind power prediction method based on hybrid kernel function support vector machine. *Wind Engineering*. 2018, 42, 3, pp. 252–264, doi: `10.1177/0309524X17737337`.

[41] WANG D., WANG P., SHI J.Z. A fast and efficient conformal regressor with regularized extreme learning machine. *Neurocomputing*. 2018, 304, pp. 1–11, doi: `10.1016/j.neucom.2018.04.012`.

[42] WANG J.Z., YANG W.D., DU P., Niu T. A novel hybrid forecasting system of wind speed based on a newly developed multi-objective sine cosine algorithm. *Energy Conversion and Management*. 2018, 163, pp. 134–150, doi: `10.1016/j.enconman.2018.02.012`.

[43] WANG Y., XIE Z.X., HU Q.H., XIONG S.H. Correlation aware multi-step ahead wind speed forecasting with heteroscedastic multi-kernel learning. *Energy Conversion and Management*. 2018, 163, pp. 384–406, doi: `10.1016/j.enconman.2018.02.034`.

[44] YU C.J., LI Y.L., XIANG H.Y., ZHANG M.J. Data mining-assisted short-term wind speed forecasting by wavelet packet decomposition and Elman neural network. *Journal of Wind Engineering and Industrial Aerodynamics*. 2018, 175, pp. 136–143, doi: `10.1016/j.jweia.2018.01.020`.

[45] YU C.J., LI Y.L., Zhang M.J. An improved Wavelet Transform using Singular Spectrum Analysis for wind speed forecasting based on Elman Neural Network. *Energy Conversion and Management*, 2017. 148, pp. 895–904, doi: `10.1016/j.enconman.2017.05.063`.

[46] YU C.J., LI Y.L., XIANG H.Y., ZHANG M.J. Data mining-assisted short-term wind speed forecasting by wavelet packet decomposition and Elman neural network. *Journal of Wind Engineering and Industrial Aerodynamics*. 2018, 175, pp. 136–143, doi: `10.1016/j.jweia.2018.01.020`.

[47] YU Y., KANG S.L., QIU H. A new network intrusion detection algorithm: DA-ROS-ELM. *IEEJ Transactions on Electrical and Electronic Engineering*. 2018, 13(4), pp. 602–612, doi: `10.1002/tee.22606`.

[48] ZHANG C., ZHOU J.Z., LI C.S., FU W. L., PENG T. A compound structure of ELM based on feature selection and parameter optimization using hybrid backtracking search algorithm for wind speed forecasting. *Energy Conversion and Management*. 2017, 143, pp. 360–376, doi: `10.1016/j.enconman.2017.04.007`.