

Research Article

Prediction Interval Construction for Byproduct Gas Flow Forecasting Using Optimized Twin Extreme Learning Machine

Xueying Sun,^{1,2} Zhuo Wang,¹ and Jingtao Hu¹

¹Department of Information Service and Intelligent Control, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China

²University of Chinese Academy of Sciences, Beijing 100049, China

Correspondence should be addressed to Jingtao Hu; hujingtao@sia.cn

Received 20 March 2017; Revised 28 May 2017; Accepted 27 July 2017; Published 23 August 2017

Academic Editor: Dan Simon

Copyright © 2017 Xueying Sun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Prediction of byproduct gas flow is of great significance to gas system scheduling in iron and steel plants. To quantify the associated prediction uncertainty, a two-step approach based on optimized twin extreme learning machine (ELM) is proposed to construct prediction intervals (PIs). In the first step, the connection weights of the twin ELM are pretrained using a pair of symmetric weighted objective functions. In the second step, output weights of the twin ELM are further optimized by particle swarm optimization (PSO). The objective function is designed to comprehensively evaluate PIs based on their coverage probability, width, and deviation. The capability of the proposed method is validated using four benchmark datasets and two real-world byproduct gas datasets. The results demonstrate that the proposed approach constructs higher quality prediction intervals than the other three conventional methods.

1. Introduction

In the iron and steel industry, the utilization of byproduct gas is of great significance to reduce the cost of fuel consumption and greenhouse gas emissions [1, 2]. The byproduct gas is generated from the iron and steel making progress and supplied to many plants (e.g., hot-rolling and power plant) via gasholders to be used as a fuel. However, since gas holders are limited in capacity, temporary excess or shortage of byproduct gas usually occurs. To solve this problem, the byproduct gas users need to be scheduled in advanced according to prediction information. Therefore, accurate forecasting for byproduct gas flow generation and consumption has become a meaningful tool to ensure the reliability and economy of energy system control and scheduling [3, 4], and this forecasting problem has attracted a large amount of interest in both industry and academia [5–10].

In recent years, artificial intelligence methods such as neural networks (NNs) [6–8] and support vector regression (SVR) [9, 10] have been adopted for handling byproduct gas forecasting. The outstanding advantage of artificial intelligence is that it can address nonlinear problems.

However, most of these models focus on point forecasting, which provides only deterministic forecasts with no information about the prediction probability. In practice, there is a large amount of uncertainty originating from temperature fluctuations and measurement errors, for example. The forecasting uncertainty will affect the decision-making process and increase the risk of scheduling, so it is imperative to quantify the uncertainty of prediction. The prediction interval (PI) is a well-known tool for quantifying the uncertainty of prediction. The PI provides not only a range within which the target values are highly likely to lie but also an indication of their accuracy [11, 12].

Although there have been only a few studies on byproduct gas flow interval forecasting, the construction of PIs in other application areas [13–17] has been studied for many years. Traditional approaches, such as the Bayesian method [14], mean-variance estimation (MVE) method [15], and bootstrap method [16], have been proposed for constructing PIs. The central idea of the Bayesian method is that the model weights are considered random variables, and the probability density distribution of the weights is estimated by the recursive Bayesian method. Despite the strength of the supporting

theories, the Bayesian method lacks proper adaptability to complex noise (e.g., heteroscedastic noise) because the model weights are assumed to be in accordance with an isotropic normal distribution [17]. The MVE method first generates a point prediction; then, a model is established to estimate the variance. The underlying assumption of this method is that the point prediction is equal to the true mean of the targets, but this condition is difficult to meet [18]. The bootstrap method is a widely used method for the construction of PIs. An ensemble of models is established to produce a less biased estimation of the point and variance prediction. The main disadvantage of this method is the high computational cost [18]. In recent years, a direct interval forecasting method called lower and upper bound estimation (LUBE) was proposed [19]. The main idea behind the method is to construct lower and upper bounds of PIs directly by optimizing the coefficients of the NN according to the interval quality evaluation indices. Because it offers good performance and does not require strict data distribution assumptions, the LUBE method has been widely used in many real-world problems, such as bus travel time prediction [20], electrical load forecasting [21, 22], and Landslide displacement prediction [23]. However, conventional NNs employed in the LUBE method suffer from the problems of overtraining and a high computational burden. Alternatively, the extreme learning machine (ELM) [24] is a new kind of feed-forward neural network with a single hidden layer. The output weights are the only coefficients that need to be trained. Thus, ELM exhibits a faster learning rate and a better generalization capability than conventional NNs [25].

However, when using NNs to perform interval forecasting, the initial values of the connection weights are usually generated randomly [19–22]. The effects of connection weight initialization on the final constructed PIs are usually ignored. In contrast to conventional NNs, ELM trains only the output weights instead of training all of the connection weights. However, these output weights still constitute a relatively large search space. Furthermore, the cost function of LUBE is the coverage width-based criterion (CWC), which cannot comprehensively describe the performance of the constructed PIs. Considering the shortcomings of the LUBE method, we propose a new two-step method based on the optimal twin ELM for constructing the PIs. Specifically, in the first stage, the twin ELM is pretrained using a pair of symmetric weighted objective functions to construct the raw PIs. Then, the values of the input weights and bias are fixed, and only the output weights are further adjusted by PSO. Furthermore, a new cost function called coverage width and deviation-based criterion (CWDC) is proposed. In the CWDC, in addition to the coverage probability and the width of the PIs, the deviation of the PIs is also considered, which gives a more comprehensive description of PI performance. The main contributions of this paper are as follows:

- (i) A twin ELM is adopted to construct PIs and the output weights of the twin ELM are pretrained using a pair of symmetric weighted objective functions. The pretraining method offers reasonable initial values and, as a characteristic of the ELM, only the output

weights need to be tuned, which benefits the subsequent optimization process.

- (ii) A modified cost function called CWDC that considers the deviation of the PIs is proposed. CWDC provides a more comprehensive description of the PI performance.
- (iii) Experiments based on four benchmark datasets and two real-world byproduct gas flow datasets are performed to illustrate the capability of the proposed approach.

The remainder of this paper is organized as follows. Section 2 introduces the related work, namely, generic ELM, formulation, and performance indices of PI. Section 3 presents the method proposed for PI construction. Performance evaluation on benchmark datasets is presented in Section 4. In Section 5, the proposed method is applied to byproduct gas forecasting problems. Finally, Section 6 concludes the paper.

2. Preliminaries

2.1. Generic ELM. ELM is a type of single-layer neural network proposed by Huang et al. [24]. It is well known for its fast training speed. Given a set $D = \{(\mathbf{x}_i, y_i), i = 1, 2, \dots, N\}$, where \mathbf{x}_i is an input vector, and y_i is the corresponding target, we assume that ELM has h hidden nodes and that the activation function is $g(x)$. The mathematical formula of ELM can be expressed as

$$y_i = \sum_{j=1}^h \beta_j g(\omega_j, b_j, \mathbf{x}_i), \quad (1)$$

where β_j is the weight vector that connects the j th hidden neurons and the output neuron. Here, ω_j is the connection weight between the corresponding input neuron and the j th hidden neuron, and b_j is the bias of the j th hidden neuron. Additionally, ω_j and b_j are generated randomly, and they remain unchanged during the training progress. Therefore, formula (1) can be simplified as

$$\mathbf{y} = \mathbf{H}\boldsymbol{\beta}, \quad (2)$$

where $\mathbf{H} = \begin{bmatrix} g(\omega_1, b_1, \mathbf{x}_1) & \dots & g(\omega_h, b_h, \mathbf{x}_1) \\ g(\omega_1, b_1, \mathbf{x}_N) & \dots & g(\omega_h, b_h, \mathbf{x}_N) \end{bmatrix}$, $\boldsymbol{\beta} = [\beta_1, \dots, \beta_h]^T$, and $\mathbf{y} = [y_1, \dots, y_N]^T$.

The objective function for training the ELM is

$$\min (\|\mathbf{H}\boldsymbol{\beta} - \mathbf{y}\|_2). \quad (3)$$

According to the Moore-Penrose inverse theorem, $\boldsymbol{\beta}$ can be calculated in the sense of a least-square estimation as follows:

$$\boldsymbol{\beta} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{y}. \quad (4)$$

2.2. Formulation of PIs. Given a set $D = \{(\mathbf{x}_i, y_i), i = 1, 2, \dots, N\}$, \mathbf{x}_i is an input vector, and y_i is the corresponding

prediction target. The PI with nominal confidence (PINC) $100(1 - \alpha)\%$ for the target y_i can be expressed as

$$I^\alpha(\mathbf{x}_i) = [L^\alpha(\mathbf{x}_i), U^\alpha(\mathbf{x}_i)], \quad (5)$$

where $L^\alpha(\mathbf{x}_i)$ and $U^\alpha(\mathbf{x}_i)$ are the lower and upper bounds of the target y_i , respectively. Thus, the probability that y_i lies within $I^\alpha(\mathbf{x}_i)$ is expected to be $100(1 - \alpha)\%$, as expressed by the following equation:

$$\text{PINC} = \Pr(y_i \in I^\alpha(\mathbf{x}_i)) = 100(1 - \alpha)\%. \quad (6)$$

2.3. Evaluation Indices of PIs. To evaluate the performance of PIs, PI coverage probability (PICP) and PI normalized average width (PINAW) are two typical indicators. The PICP is used to evaluate the reliability of the constructed PIs and is defined as

$$\text{PICP} = \frac{1}{N} \sum_{i=1}^N c_i, \quad (7)$$

where N is the number of data samples and

$$c_i = \begin{cases} 1, & \text{if } y_i \in I^\alpha(\mathbf{x}_i) \\ 0, & \text{if } y_i \notin I^\alpha(\mathbf{x}_i). \end{cases} \quad (8)$$

According to the concept of a PI, the value of the PICP is expected to be greater than or equal to the predetermined confidence level; otherwise, the PIs are invalid.

A relatively high PICP can be easily achieved if the width of the PIs is sufficiently large, but wider PIs are less informative in practice. The PINAW, which can quantitatively describe the width of the PIs, is defined as

$$\text{PINAW} = \frac{1}{N} \sum_{i=1}^N \frac{U^\alpha(\mathbf{x}_i) - L^\alpha(\mathbf{x}_i)}{y_{\max} - y_{\min}}, \quad (9)$$

where y_{\max} and y_{\min} , respectively, represent the maximum and minimum values of the targets.

To construct PIs to provide satisfactory performance, a higher PICP and lower PINAW are expected. However, the two indices conflict with each other. To find a compromise between them, a combined measure called coverage width-based criterion (CWC) has been proposed [19]. The CWC is defined as

$$\text{CWC} = \text{PINAW} \left(1 + \gamma (\text{PICP}) e^{-\eta(\text{PICP} - \mu)} \right), \quad (10)$$

where

$$\gamma = \begin{cases} 0, & \text{if } \text{PICP} \geq \mu \\ 1, & \text{if } \text{PICP} < \mu. \end{cases} \quad (11)$$

μ and η are two hyperparameters of the CWC. Usually, μ is set to the PINC, and η determines the amount of punishment when the PICP is lower than PINC.

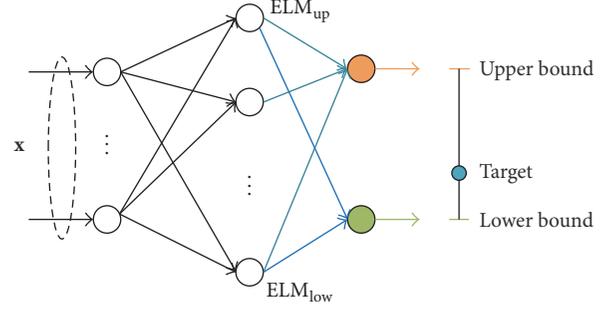


FIGURE 1: The overall structure of the proposed method.

3. The Proposed Method

3.1. Framework Overview. Our proposed method is derived from the LUBE method. As ELM is a new type of NN with simple structure and fast training speed, we adopt a twin ELM to construct the lower and upper bounds of PIs. The overall structure of the proposed method is shown in Figure 1. A two-step approach is proposed to determine the connection weights of the twin ELM. The flowchart of the proposed method is described in Figure 2.

In the first step, the connection weights of the twin ELM are pretrained using a pair of symmetric weighted objective functions to construct raw prediction intervals. By using a symmetric pair of weights, two ELM models (ELM_{up} and ELM_{low}) can be obtained to generate the lower and upper bounds of the raw PIs. Then, as a characteristic of ELM, the values of the input weights are fixed, and only the output weights are further adjusted by the later optimal process.

Then, the second step is to improve the quality of the PIs by further adjusting the output weights based on PSO.

3.2. Objective Function. In LUBE method, CWC can provide a good compromise between PICP and PINAW. It cannot describe the deviation of PIs, however. In order to evaluate PIs comprehensively, a modified cost function called CWDC that considers the deviation of the PIs is proposed. The CWDC is defined as

$$\begin{aligned} \text{CWDC} &= (\text{PINAW} + \varphi \text{PINAD}) \left(1 + \gamma (\text{PICP}) e^{-\eta(\text{PICP} - \mu)} \right), \end{aligned} \quad (12)$$

where φ is a parameter to adjust the punishment of the deviation. In this paper, it is set to 20. PI normalized average deviation (PINAD) is to quantitatively describe the deviation degree from the real values to the PIs and is defined as

$$\text{PINAD} = \frac{1}{N} \sum_{i=1}^N \frac{d_i}{y_{\max} - y_{\min}}, \quad (13)$$

where

$$d_i = \begin{cases} L_i^\alpha - y_i, & \text{if } y_i < L_i^\alpha \\ 0, & \text{if } L_i^\alpha \leq y_i \leq U_i^\alpha \\ y_i - U_i^\alpha, & \text{if } y_i > U_i^\alpha. \end{cases} \quad (14)$$

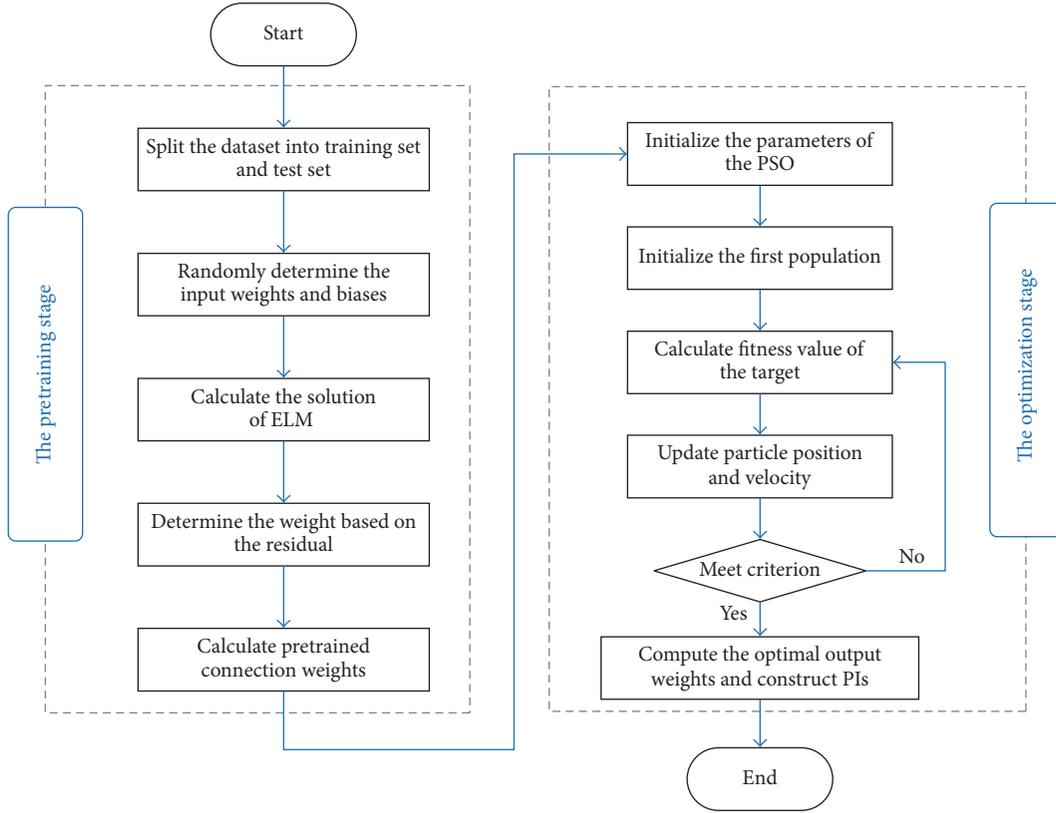


FIGURE 2: Flowchart of the proposed method.

The training process is to minimize the objective function. However, when the PIs are poor-quality with zero-width (PINAW = 0%) and extremely low coverage probability (PICP = 0%), CWC will equal the minimum value 0. This will lead to a misleading evaluation of PIs. As for CWDC, by taking the deviation of PIs into consideration, the problem can be solved.

3.3. Weighted Pretraining of the Twin ELM. In the proposed method, the pretraining is designed to get reasonable initial values for the output weights before optimization. In the LUBE method, the initial values of the connection weights are usually generated randomly and the training process is generally slow because there is a large search space. In this paper, the output weights of the twin ELM are pretrained using a pair of symmetric weighted objective functions. At the beginning of the following optimization process, initial population of PSO is generated around the pretrained output weights as reasonable initial values.

The details of the pretraining process are elaborated as follows. For a given set $D = \{(\mathbf{x}_i, y_i), i = 1, 2, \dots, N\}$, \mathbf{x}_i is an input vector, and y_i is the corresponding prediction target, according to (2) and (4), and point forecasting \hat{y} of an ELM can be formulated as

$$\hat{y} = \mathbf{H}\hat{\beta}, \quad (15)$$

where

$$\hat{\beta} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{y}. \quad (16)$$

The residual ε_i of the i th sample can be formulated as

$$\varepsilon_i = y_i - \mathbf{H}(\mathbf{x}_i) \hat{\beta}, \quad i = 1, 2, \dots, N. \quad (17)$$

The twin ELM consists of two ELMs, which have the same input weights and bias as shown in Figure 1. For the ELM that generates the upper bound, a greater weight is given to the data that has a positive error. Otherwise, the ELM that generates the lower bound gives greater weight to the data that has a negative error. The weight functions are defined as follows:

$$w_i^{\text{up}}(\alpha) = \begin{cases} 1 - \frac{1}{2}\alpha, & \text{if } \varepsilon_i \geq 0 \\ \frac{1}{2}\alpha, & \text{if } \varepsilon_i < 0, \end{cases} \quad (18)$$

$$w_i^{\text{low}}(\alpha) = \begin{cases} \frac{1}{2}\alpha, & \text{if } \varepsilon_i \geq 0 \\ 1 - \frac{1}{2}\alpha, & \text{if } \varepsilon_i < 0. \end{cases}$$

In the above two equations, $\alpha \in (0, 1)$ represents the confidence level. To reduce ELM overfitting, the norms of the

weights are introduced to the loss function. As a result, the mathematical model of the twin ELM can be described as

$$\min_{\beta} \mathbf{W}^{\text{up}} \|\boldsymbol{\varepsilon}\|_2 + \lambda \|\boldsymbol{\beta}^{\text{up}}\|_2, \quad (19)$$

$$\min_{\beta} \mathbf{W}^{\text{low}} \|\boldsymbol{\varepsilon}\|_2 + \lambda \|\boldsymbol{\beta}^{\text{low}}\|_2, \quad (20)$$

where $\mathbf{W}^{\text{up}} = \text{diag}\{w_1^{\text{up}}, w_2^{\text{up}}, \dots, w_N^{\text{up}}\}$ and $\mathbf{W}^{\text{low}} = \text{diag}\{w_1^{\text{low}}, w_2^{\text{low}}, \dots, w_N^{\text{low}}\}$. Here, λ is a positive weight factor for adjusting the proportion of empirical risk and structural risk.

The Lagrangian of (19) is defined as

$$L(\beta, \boldsymbol{\varepsilon}, \boldsymbol{\gamma}) = \mathbf{W} \|\boldsymbol{\varepsilon}\|_2 + \lambda \|\boldsymbol{\beta}\|_2 + \boldsymbol{\gamma}^T (\mathbf{y} - \mathbf{H}\boldsymbol{\beta} - \boldsymbol{\varepsilon}), \quad (21)$$

where $\boldsymbol{\gamma}$ is the Lagrangian multiplier. In (21), the subscript up is omitted for brevity. According to the Karush-Kuhn-Tucker (KKT) theorem, the optimality conditions are given by

$$\begin{aligned} \frac{\partial L}{\partial \beta} &\longrightarrow 2\lambda\boldsymbol{\beta} - \mathbf{H}^T \boldsymbol{\gamma} = 0, \\ \frac{\partial L}{\partial \boldsymbol{\varepsilon}} &\longrightarrow 2\boldsymbol{\varepsilon}\mathbf{W} - \boldsymbol{\gamma} = 0, \\ \frac{\partial L}{\partial \boldsymbol{\gamma}} &\longrightarrow \mathbf{y} - \mathbf{H}\boldsymbol{\beta} - \boldsymbol{\varepsilon} = 0. \end{aligned} \quad (22)$$

According to (22), we can calculate β by the following equation:

$$\boldsymbol{\beta} = (\lambda\mathbf{I} + \mathbf{H}^T \mathbf{W} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{W} \mathbf{y}. \quad (23)$$

The method for solving (20) is the same as that for (19) and is therefore omitted for brevity. Then, the raw PIs constructed by the twin ELM can be expressed as

$$I_{\text{raw}}^{\alpha}(\mathbf{x}_i) = [y_i^{\alpha}(\mathbf{x}_i), y_u^{\alpha}(\mathbf{x}_i)], \quad (24)$$

where

$$\begin{aligned} y_i^{\alpha}(\mathbf{x}_i) &= \sum_{j=1}^h \beta_j^{\text{low}} g(\boldsymbol{\omega}_j, b_j, \mathbf{x}_i), \\ y_u^{\alpha}(\mathbf{x}_i) &= \sum_{j=1}^h \beta_j^{\text{up}} g(\boldsymbol{\omega}_j, b_j, \mathbf{x}_i). \end{aligned} \quad (25)$$

3.4. Weight Adjustments by PSO. The optimization stage is performed to further adjust the connection weights of the pretrained twin ELM. The goal of the optimization is to achieve the PIs of the best quality through optimizing the connection weights with respect to the objective function (12). Because the objective function of the proposed method is complex and nondifferentiable, gradient descent based techniques are not suitable for this problem. PSO is an efficient and gradient-free algorithm based on the principle of bionic metaheuristic [26] and was applied to solve a large number of optimization problems in recent years [27–29]. Accordingly, PSO is used to determine the final values of the twin ELM output weights.

Suppose that there are a group of M particles in a d -dimensional search space. The position of the i th particle is $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{id})$, and its velocity is $\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{id})$. Each particle stores the optimal location of its flight \mathbf{p}_{id} . The optimal location of all of the particles in the particle swarm is the globally optimal position \mathbf{p}_{gd} . Each particle adjusts its state according to (26):

$$\begin{aligned} \mathbf{v}_{id}^{t+1} &= \omega \mathbf{v}_{id}^t + c_1 r_1 (\mathbf{p}_{id}^t - \mathbf{x}_{id}^t) + c_2 r_2 (\mathbf{p}_{gd}^t - \mathbf{x}_{id}^t), \\ \mathbf{x}_{id}^{t+1} &= \mathbf{x}_{id}^t + \mathbf{v}_{id}^{t+1}, \end{aligned} \quad (26)$$

where t represents the number of iterations and ω is the inertia weight that affects the flying speed of the particle. Here, c_1 and c_2 are the acceleration coefficients, and r_1 and r_2 are random variables that are uniformly distributed on the domain $[0, 1]$. With the continuous updating of the position and velocity, the particle swarm can find the optimal solution in the space. The inertia weight has an effect on the velocity of the particle. A larger value is appropriate for the global search, and a smaller value is appropriate for the local search. In the standard PSO algorithm, the inertia weight factor is a fixed value. To balance the search ability of the algorithm, we adopt the exponential decline algorithm to adjust the inertia weight. The inertia weight in the k th interaction can be calculated based on the following equation:

$$\omega(k) = \omega z^k. \quad (27)$$

In (27), z is an attenuation coefficient.

The details of how PSO tunes the output weights of the twin ELM are as follows:

- (1) Initialize the parameters of the modified PSO. Specify the maximum number of iterations, the population size, and the parameters of the flying progress, including the inertia weight ω and the acceleration coefficients c_1 and c_2 .
- (2) Initialize the first population. Note that each particle in the population represents a set of output weights, including $\boldsymbol{\beta}^{\text{up}}$ and $\boldsymbol{\beta}^{\text{low}}$. The output weights are generated around the pretrained output weights by adding small random values, which are uniformly distributed between -0.1 and 0.1 to preserve the diversity.
- (3) Calculate the fitness value of the target. For each particle in the population, the fitness value is calculated using (12).
- (4) Update the optimal values. The local best position $\mathbf{p}_i^{\text{best}}$ and the global best position $\mathbf{p}_g^{\text{best}}$ are updated through comparing the fitness values.
- (5) Update the particle position and velocity according to (26) and (27).
- (6) Repeat steps (3)–(5) until the iteration number reaches the preset threshold value. Then, return the smallest value of the cost function CWDC and the best positions $\boldsymbol{\beta}^{\text{up}}$ and $\boldsymbol{\beta}^{\text{low}}$.
- (7) Evaluate the PIs that are generated by the optimal the twin ELM based on the test data.

4. Benchmark Test

4.1. Datasets. The effectiveness of the proposed method is tested on four benchmarks. The details of the four benchmarks are described as follows.

Case 1 is a 5D mathematical function that is usually used for evaluating regression models, and it is defined as

$$y = 0.0647 \left(12 + 3x_1 - 3.5x_2^2 + 7.2x_3^3 \right) \cdot (1 + \cos(4\pi x_4)) (1 + 0.8 \sin(3\pi x_5)), \quad (28)$$

where $x_i \in [0, 1]$, $i = 1, \dots, 5$. A normally distributed noise (S/N ratio = 4) is added to the outputs.

Case 2 is a synthetic dataset used to model the data with heteroscedastic noise. The dataset is generated by the following function:

$$y = \sin x + 2 \sin(0.3x), \quad (29)$$

where $x \in [-10, 10]$. A normally distributed zero-mean noise that has an input-dependent variance $\sigma(x) = 0.45 \sin(0.6x - 2) + 0.55$ is added to the outputs.

Case 3 investigates the relationship between the personal characteristics, dietary factors, and plasma levels of beta-carotene [30]. The datasets contain 315 observations on 12 input variables and 1 output variable.

Case 4 is taken from the UCI repository [31], which is collected from a combined cycle power plant. The input variables contain the temperature, ambient pressure, relative humidity, and exhaust vacuum. The target is to predict the next hourly electrical energy output.

4.2. Experiment Methodology. The comparison results of various prediction methods are discussed in this section. To evaluate the performance of the pretraining method, the proposed method and the ELM without pretraining are first compared. Then the proposed method using CWDC as the objective function is compared with the ELM using CWC (ELM-CWC method). The effectiveness of proposed method is also demonstrated by comparing it with the bootstrap method and the Bayesian method. The PI evaluation indices PICP, PINAW, and PINAD for the four cases are shown in tables. Considering the inherent randomness of PSO and ELM, each experiment is repeated 10 times for all the methods. The results of the median values of the indices are recorded.

The parameters of the PSO are assigned as follows: $c_1 = 1.5$, $c_2 = 2.0$, and $z = 0.99$. The population size is set to 100, and the maximum number of iterations is assigned to 400. In this case, the PINC is set to 90%, and μ equals 0.1. For CWDC, the parameters are set as follows: $\mu = 90\%$, $\varphi = 20$, and $\eta = 50$.

For each dataset, the training set includes 200 samples and additional 100 observations is used for testing the models. To determine the optimal number of hidden nodes of ELM, k -fold cross validation is adopted. The training data are divided into k equal parts. In our study, k is set to 5. The model is trained on $k-1$ parts, and the remaining part is used to validate the trained model. As a result, the optimal numbers of hidden nodes for Cases 1 and 2 are selected to be 12 and 15,

respectively. The optimal numbers of hidden nodes selected for Cases 3 and 4 are 20 and 18, respectively, based on the experimental results.

4.3. Experiment Results and Discussion

4.3.1. Comparison of Different Initialization Methods. In the proposed approach, pretraining method is designed to provide a reasonable initial value for the following optimization process. To evaluate the effectiveness of the pretraining strategy, randomly initialized ELM (RI-ELM method) is employed as a comparative approach. In RI-ELM, all the weights of ELM are given randomly and the output weights are directly tuned by PSO. To take a better comparison, CWDC is employed as the cost function of both methods.

The iterative processes of PSO training of the two methods in the four cases are shown in Figure 3. As can be seen in Figure 3, the proposed method converges faster than RI-ELM and it indicates that pretraining method accelerates the convergence speed of PSO. Test results of the four cases are shown in Table 1. It is demonstrated that the proposed method can generate PIs with higher performance compared to RI-ELM. The reason is that proposed method gives more reasonable output values of ELM in the initialization stage.

4.3.2. Comparison of ELM Based Methods Using CWC and CWDC. In order to test the effectiveness of the proposed criterion CWDC, the ELM using CWC (ELM-CWC) is compared with the proposed method. For the superior performance of pretraining based ELM, it is used to construct PIs. The results of ELM-CWC and the proposed method are summarized in Table 2. As shown in Table 2, the proposed method achieves lower PINAWs and smaller PINADs with valid PICPs for all four cases. Because the deviation of the PIs is considered in the objective function, PIs with higher performance is obtained.

PINAD is a new index that is considered in CWDC and is proposed to show the deviation PIs. PINAD and PICP have some correlation. A higher PICP implies that more prediction points lie in the constructed PIs, which may lead to a smaller PINAD. However, the PICP can only represent the proportion of the prediction points lying in the constructed PIs, while the PINAD can show the deviation degree of the data that do not lie in the constructed PIs. Thus, the PINAD is a necessary supplement to the PICP.

4.3.3. Comparison Experiment with Conventional Methods. PIs obtained by the proposed method of the four case studies are illustrated in Figure 4. This figure visually demonstrates that the constructed PIs cover the real targets in a high percentage.

As shown in Table 3, for Cases 1 and 2 the proposed method obtains the narrowest PINAWs and smallest PINADs with satisfactory PICPs ($\geq 90\%$). The median PINAWs of the proposed method are 62.14% for Case 1 and 29.64% for Case 2, which are the narrowest for both cases. Additionally, the median PINADs of the proposed method are 0.61% for Case 1 and 0.27% for Case 2, which are smaller than the results of other methods.

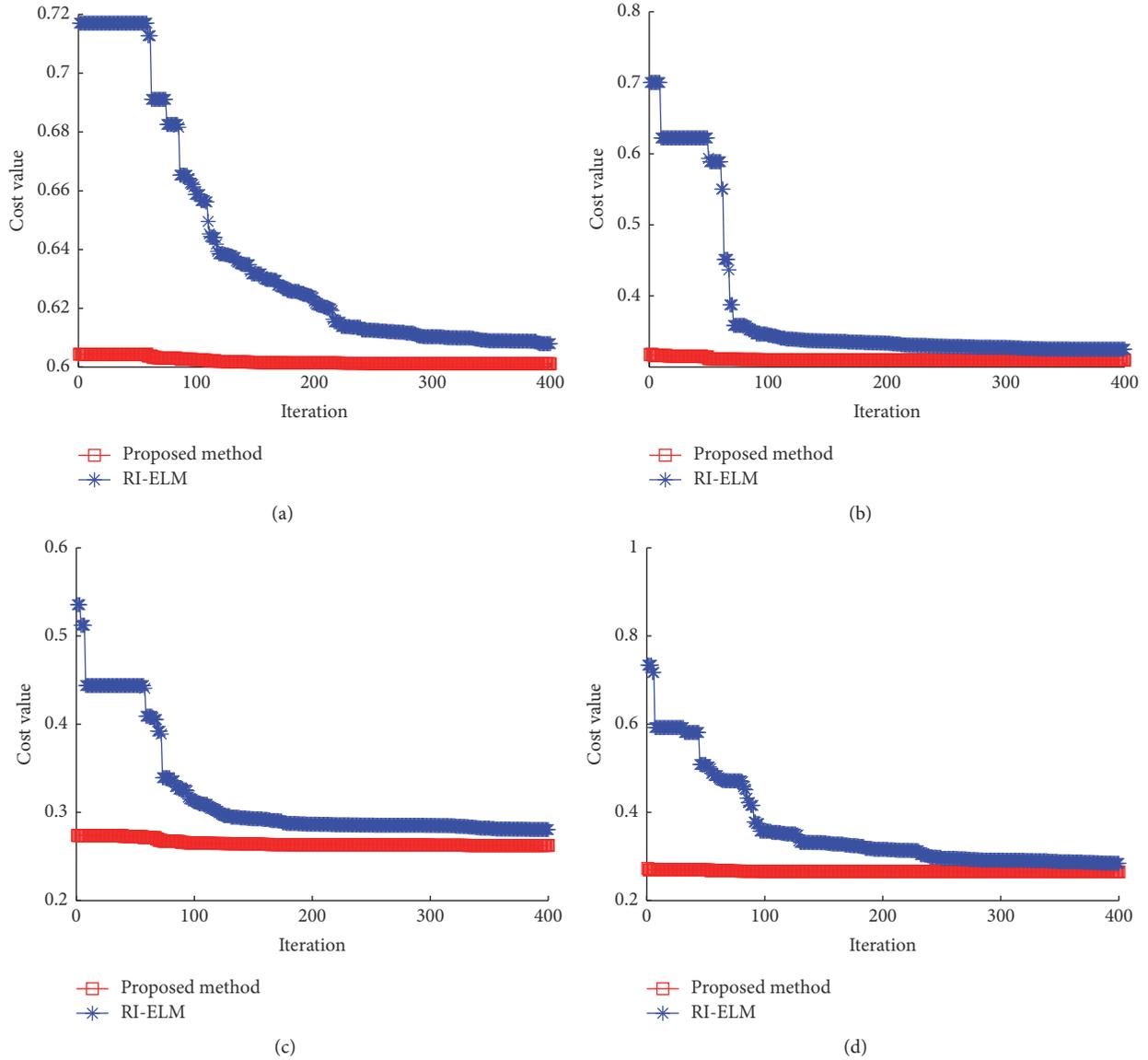


FIGURE 3: Training processes of proposed method and RI-ELM for case 1 (a), case 2 (b), case 3 (c), and case 4 (d).

TABLE 1: Comparisons of proposed method and RI-ELM.

PINC	Method	PICP	PINAW	PINAD
Case 1	RI-ELM	90.50%	64.59%	0.66%
	Proposed method	91.50%	62.14%	0.61%
Case 2	RI-ELM	91.50%	30.84%	0.28%
	Proposed method	93.00%	29.64%	0.27%
Case 3	RI-ELM	90.50%	34.59%	1.66%
	Proposed method	91.00%	30.18%	1.63%
Case 4	RI-ELM	91.50%	24.88%	0.14%
	Proposed method	92.00%	22.48%	0.14%

TABLE 2: Comparisons of proposed method and ELM-CWC.

PINC	Method	PICP	PINAW	PINAD
Case 1	ELM-CWC	90.00%	62.98%	0.70%
	Proposed method	91.50%	62.14%	0.61%
Case 2	ELM-CWC	91.50%	29.80%	0.32%
	Proposed method	93.00%	29.64%	0.27%
Case 3	ELM-CWC	91.00%	33.60%	1.75%
	Proposed method	91.50%	30.18%	1.63%
Case 4	ELM-CWC	91.00%	23.63%	0.16%
	Proposed method	92.00%	22.48%	0.14%

As reported in Table 3, the median PINAWs and PINADs of the proposed method for Cases 3 and 4 are lower than the results of the bootstrap method. It should also be noted that

the proposed method does not provide the highest PICP and the lowest PINAD in these two cases. Relative to the Bayesian method, the PICPs of the proposed method are 3.33% and

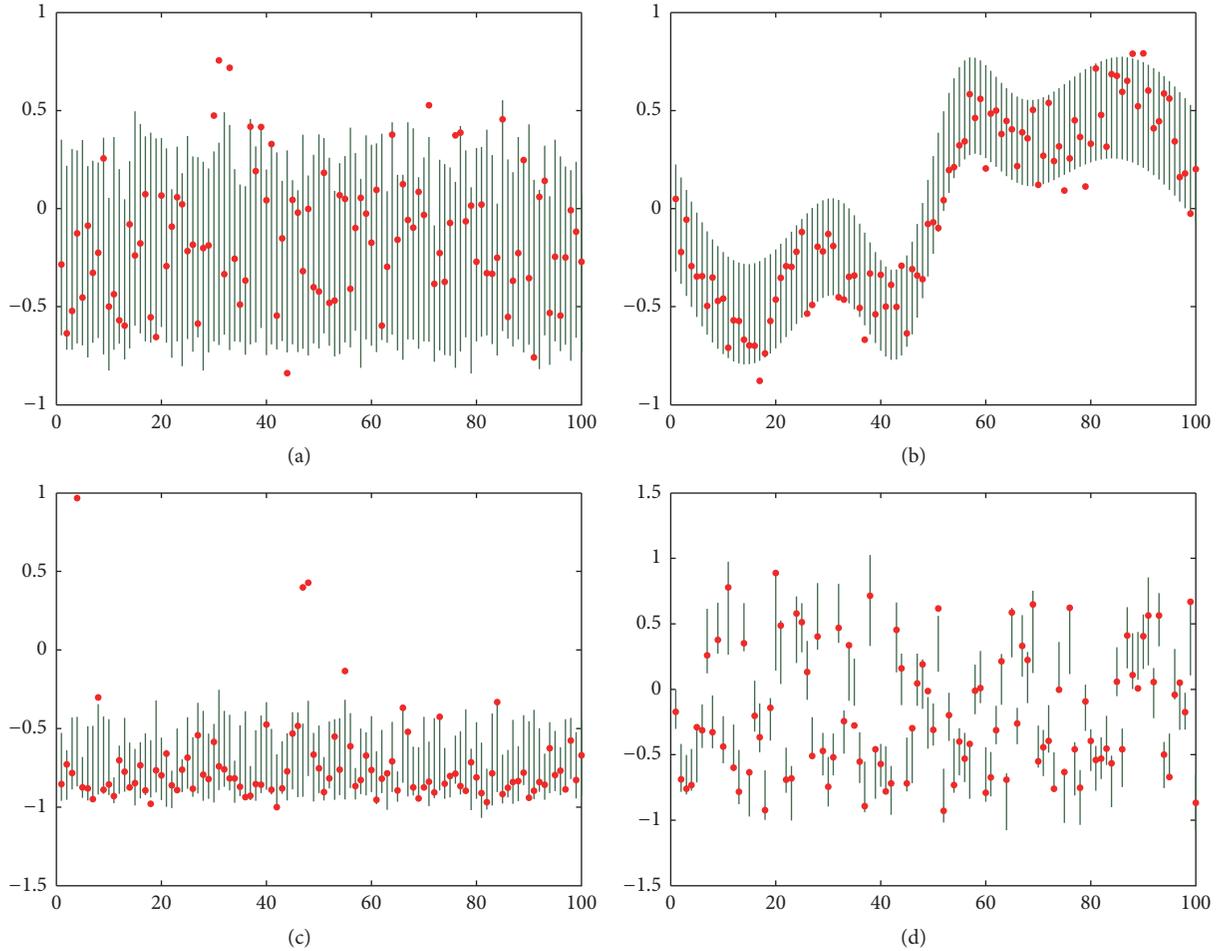


FIGURE 4: PIs obtained by the proposed method for Case 1 (a), Case 2 (b), Case 3 (c), and Case 4 (d).

TABLE 3: Test results in the four benchmark datasets.

Methods	Index	Case 1			Case 2			Case 3			Case 4		
		PICP	PINAW	PINAD									
Bayesian	Median	91.00%	65.78%	0.79%	86.00%	31.37%	0.44%	94.50%	37.90%	1.49%	94.00%	26.50%	0.11%
	STDEV	1.48%	0.78%	0.10%	1.29%	0.86%	0.05%	0.98%	1.31%	0.12%	1.57%	0.85%	0.03%
Bootstrap	Median	88.00%	62.50%	0.78%	92.00%	29.98%	0.28%	91.50%	36.93%	1.69%	92.00%	24.11%	0.17%
	STDEV	1.52%	0.79%	0.09%	1.16%	0.72%	0.04%	0.94%	1.37%	0.14%	1.52%	0.87%	0.04%
Proposed	Median	91.50%	62.14%	0.61%	93.00%	29.64%	0.27%	91.50%	30.18%	1.63%	92.00%	22.48%	0.14%
	STDEV	1.45%	0.65%	0.09%	1.06%	0.60%	0.04%	0.82%	1.17%	0.10%	1.49%	0.75%	0.02%

2% lower in Cases 3 and 4, respectively, and the PINADs are 0.14% and 0.03% higher in cases 3 and 4, respectively. However, the proposed method generates much narrower PIs, and the PINAWs are 7.72% and 4.02% narrower than the Bayesian method in Cases 3 and 4, respectively. That finding implies that the PIs constructed by our method are more informative than those of the Bayesian method with satisfactory PICP ($\geq 90\%$).

5. Prediction of the BFG Flow

In this section, the proposed method is used to solve a real-world problem; in this case, the method is used to handle the uncertainty of blast furnace gas (BFG) generation and consumption prediction in the iron and steel industry.

5.1. Datasets. BFG is an important type of byproduct gas from the steel production process; it is used for coking, sintering,

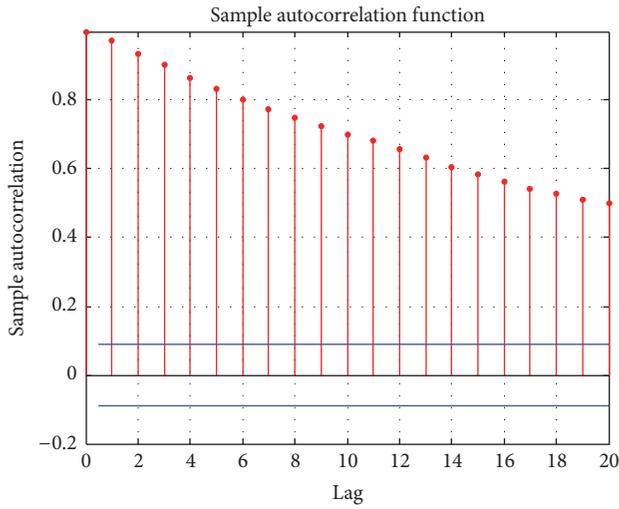


FIGURE 5: ACF of BFG generation series.

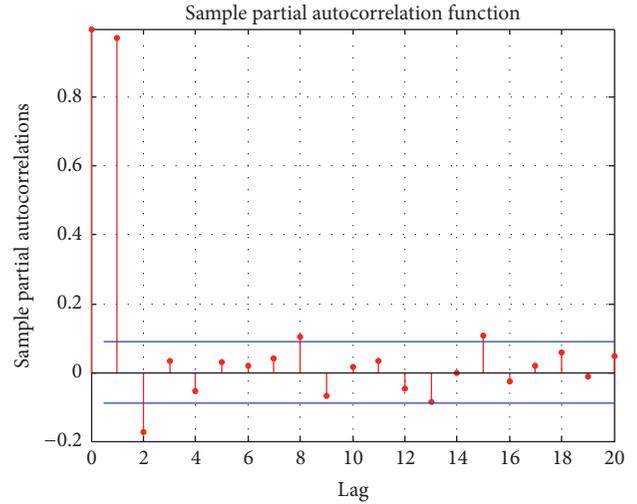


FIGURE 6: PACF of BFG generation series.

hot-rolling, and other production processes. The historical datasets of two typical BFG generation and consumption processes are chosen as case studies to illustrate the advantages of the proposed method in practical applications. The first dataset is BFG generation. The second dataset covers BFG consumption in a hot-rolling process. Both datasets come from the energy management center of a steel plant in China, Jiangsu Province. The sample interval is 5 min, and the total number of sampling points in each dataset is 288. The dataset is divided into two parts: the training set (188 points) and the test set (100 points). One-step-ahead forecasting using the proposed method is implemented and compared with the other three methods.

5.2. Determination of the Model Structure and Parameters.

Inputs of an NN are usually determined based on the characteristics of the system. BFG generation and consumption prediction can be seen as a time series prediction problem. The autocorrelation function (ACF) and partial autocorrelation function (PACF) are adopted as the criteria to determine the time lags of the input. The ACF and PACF of the BFG generation are displayed in Figures 5 and 6. The ACF has a decaying tendency, while the PACF shows a cutoff pattern after lag 3. As a result, the model order of this time series is 3. Similarly, the suitable order for the BFG consumption in the hot-rolling process is chosen to be 4. The number of hidden nodes of the ELM is selected based on the 5-fold validation method. According to the experiments, the number of hidden nodes for the blast furnace and hot-rolling process are chosen to be 12 and 18, respectively.

For these two cases, the parameters of the PSO are assigned as follows: $c_1 = 0.5$, $c_2 = 1.5$, and $z = 0.99$. The population size is set to 100, and the maximum number of iterations is assigned to be 400. To further validate the capacity of the proposed method, PINC is set to three different values, namely, 90%, 95%, and 99%, and thus, $\alpha = 0.1$, 0.5, and 0.01, respectively. For the CWDC, the parameters are set as follows: $\mu = 90%$, 95%, and 99% and $\varphi = 20$ and $\eta = 50$.

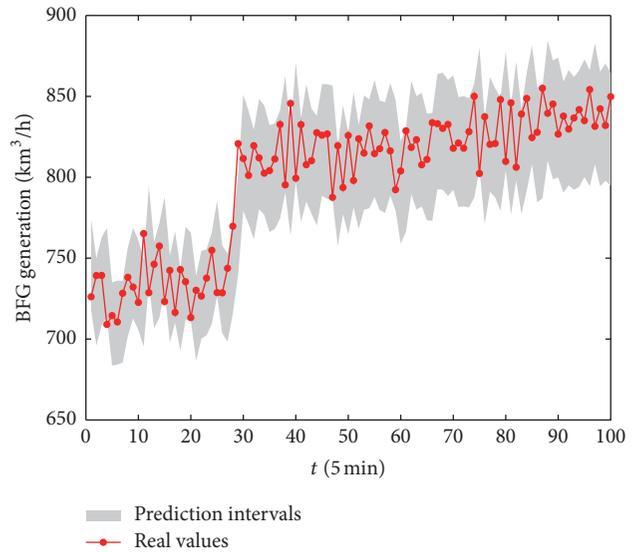


FIGURE 7: PIs with PINC = 95% obtained by the proposed method for BFG generation.

5.3. Experiment Results and Discussion.

PIs with PINC = 95% obtained by the proposed method are shown in Figures 7 and 8. These figures visually demonstrate that actual gas flow values are mostly covered by the generated PIs for the two cases. The detailed testing results for BFG generation and BFG consumption are given in Tables 4 and 5, respectively. According to Tables 4 and 5, the following conclusions can be drawn.

- (1) Relative to the Bayesian method, the proposed method provides satisfactory PICPs at all confidence levels in both cases. For example, at the confidence level with PINC = 90%, the PICPs of the PIs constructed by the proposed method are 91% for the blast furnace and 91% for the hot-rolling process. The PICPs of the PIs constructed by the Bayesian method

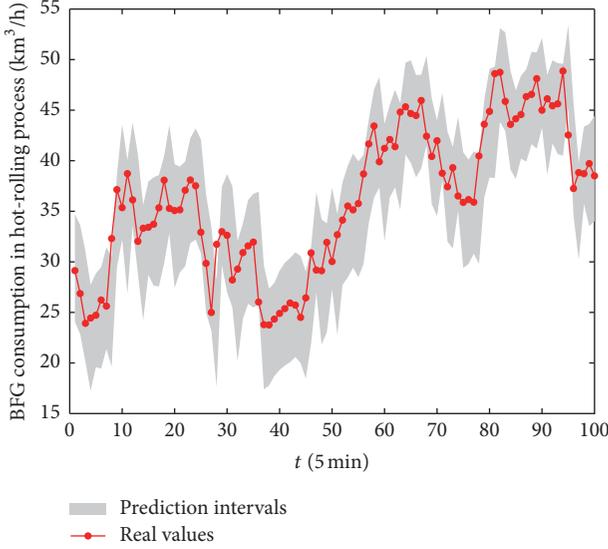


FIGURE 8: PIs with PINC = 95% obtained by the proposed method for BFG consumption in the hot-rolling process.

TABLE 4: Comparisons of different methods for BFG generation.

PINC	Method	PICP	PINAW	PINAD
90%	Bayesian	87.00%	43.22%	0.60%
	Bootstrap	90.00%	48.16%	0.57%
	LUBE	90.00%	46.45%	0.52%
	Proposed method	91.00%	45.36%	0.46%
95%	Bayesian	95.00%	57.50%	0.16%
	Bootstrap	96.00%	56.81%	0.20%
	LUBE	96.00%	54.33%	0.21%
	Proposed method	96.00%	53.73%	0.18%
99%	Bayesian	99.00%	70.74%	0.01%
	Bootstrap	99.00%	72.12%	0.01%
	LUBE	99.00%	68.56%	0.01%
	Proposed method	99.00%	65.54%	0.01%

TABLE 5: Comparisons of different methods for BFG consumption in the hot-rolling process.

PINC	Method	PICP	PINAW	PINAD
90%	Bayesian	89.00%	32.76%	0.47%
	Bootstrap	90.00%	35.71%	0.52%
	LUBE	91.00%	33.52%	0.50%
	Proposed method	91.00%	32.29%	0.44%
95%	Bayesian	96.00%	41.90%	0.23%
	Bootstrap	96.00%	42.87%	0.20%
	LUBE	95.00%	40.90%	0.27%
	Proposed method	96.00%	39.06%	0.22%
99%	Bayesian	99.00%	59.17%	0.03%
	Bootstrap	99.00%	58.50%	0.04%
	LUBE	99.00%	55.62%	0.04%
	Proposed method	99.00%	54.60%	0.03%

are 87% for the blast furnace and 89% for the hot-rolling process, which are lower than the PINC. This result occurs because the Bayesian method makes assumptions about the error distributions, but the assumptions cannot always be fitted to the facts. The performance of the proposed method is clearly better than that of the Bayesian method.

- (2) In both case studies, the proposed method provides better performance than the bootstrap method. Taking the case of the blast furnace as an example, the proposed method constructs PIs with PINAWs that are 2.80%, 3.08%, and 4.58% lower than those of the bootstrap method at the three confidence levels. This observation shows that the proposed method generates more informative PIs than the bootstrap method.
- (3) The PINAWs of the PIs constructed by the proposed method are narrower than those of the LUBE method, and the PINADs of the PIs constructed by the proposed method are also much lower than those of the LUBE method. For example, the proposed method constructs PIs with PINADs that are 2.80%, 3.08%, and 4.58% lower than those of the LUBE method in the case of the hot-rolling process at the three confidence levels. The reason is that the CWDC proposed in this study considers the deviation of the PIs.

According to the results of the analysis, all four methods can be used to construct PIs for byproduct gas generation and consumption forecasting. The PIs generated by the four methods in most cases have valid PICPs and reasonable PINAWs and PINADs. However, it is clear that the proposed approach constructs higher quality PIs than the other three conventional methods. Thus the proposed method is an effective way to improve the quality of PIs for byproduct gas flow forecasting applications.

6. Conclusions

To quantify the uncertainties of byproduct gas forecasting, a two-step approach based on a twin ELM is proposed to construct high quality PIs. The twin ELM is first pretrained using a pair of symmetric weighted objective functions. Then, the output weights are further adjusted by PSO. During the procedure, a novel objective function, which accounts for the PI coverage probability, width, and deviation, is constructed to obtain optimal PIs. The effectiveness of the proposed method has been successfully verified through tests and comparisons with well-established methods using both benchmark datasets and real-world byproduct gas datasets. Demonstrated results show that not only valid PICPs and narrow PINAWs are obtained, but also small PINADs are achieved. The results have illustrated that the proposed method constructs high quality PIs for byproduct gas forecasting application. The constructed PIs can be applied to gas system scheduling for making reliable decisions.

Although we have only verified the effectiveness of the proposed method in iron and steel plants, it could be easily

extended to other fields such as wind speed and electric load forecasting. Furthermore, as with other applications of ELM, the performance of the proposed method might be improved though the combination of the proposed method with structure selection strategy, for example, pruning strategy [32]. Some other novel evolutionary algorithms could be applied to enhance the quality of constructed PIs.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research is supported by the Key Research Program of the Chinese Academy of Sciences Grant no. KFZD-SW-415 and Science and Technology Service Network Initiative no. KTJ-SW-STS-159.

References

- [1] C. Sheinbaum-Pardo, "Decomposition analysis from demand services to material production: The case of CO₂ emissions from steel produced for automobiles in Mexico," *Applied Energy*, vol. 174, pp. 245–255, 2016.
- [2] B. Xu and B. Lin, "Assessing CO₂ emissions in China's iron and steel industry: A dynamic vector autoregression model," *Applied Energy*, vol. 161, pp. 375–386, 2016.
- [3] V. B. de Oliveira Junior, J. G. C. Pena, and J. L. F. Salles, "An improved plant-wide multiperiod optimization model of a byproduct gas supply system in the iron and steel-making process," *Applied Energy*, vol. 164, pp. 462–474, 2016.
- [4] X. Zhao, H. Bai, X. Lu, Q. Shi, and J. Han, "A MILP model concerning the optimisation of penalty factors for the short-term distribution of byproduct gases produced in the iron and steel making process," *Applied Energy*, vol. 148, pp. 142–158, 2015.
- [5] G. F. Porzio, V. Colla, N. Matarese et al., "Process integration in energy and carbon intensive industries: An example of exploitation of optimization techniques and decision support," *Applied Thermal Engineering*, vol. 70, no. 2, pp. 1148–1155, 2014.
- [6] Y. Liu, J. Zhao, W. Wang, Y.-P. Wu, and W.-C. Chen, "Improved echo state network based on data-driven and its application to prediction of blast furnace gas output," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 35, no. 6, pp. 731–738, 2009.
- [7] J. Zhao, Q. Liu, W. Wang, W. Pedrycz, and L. Cong, "Hybrid neural prediction and optimized adjustment for coke oven gas system in steel industry," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 3, pp. 439–450, 2012.
- [8] L. Zhang, C. Hua, Y. Tang, and X. Guan, "Ill-posed Echo State Network based on L-curve Method for Prediction of Blast Furnace Gas Flow," *Neural Processing Letters*, vol. 43, no. 1, pp. 97–113, 2016.
- [9] J. Zhao, W. Wang, Y. Liu, and W. Pedrycz, "A two-stage online prediction method for a blast furnace gas system and its application," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 3, pp. 507–520, 2011.
- [10] Z. Han, Y. Liu, J. Zhao, and W. Wang, "Real time prediction for converter gas tank levels based on multi-output least square support vector regressor," *Control Engineering Practice*, vol. 20, no. 12, pp. 1400–1409, 2012.
- [11] J. Zhao, Q. Liu, W. Pedrycz, and D. Li, "Effective noise estimation-based online prediction for byproduct gas system in steel industry," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 953–963, 2012.
- [12] Z. Lv, J. Zhao, Y. Liu, and W. Wang, "Use of a quantile regression based echo state network ensemble for construction of prediction Intervals of gas flow in a blast furnace," *Control Engineering Practice*, vol. 46, pp. 94–104, 2016.
- [13] A. J. Jones, D. Evans, and S. E. Kemp, "A note on the gamma test analysis of noisy input/output data and noisy time series," *Physica D. Nonlinear Phenomena*, vol. 229, no. 1, pp. 1–8, 2007.
- [14] P. Lauret, E. Fock, R. N. Randrianarivony, and J.-F. Manicom-Ramsamy, "Bayesian neural network approach to short time load forecasting," *Energy Conversion and Management*, vol. 49, no. 5, pp. 1156–1166, 2008.
- [15] W. Yao, Z. Zeng, and C. Lian, "Generating probabilistic predictions using mean-variance estimation and echo state network," *Neurocomputing*, vol. 219, pp. 536–547, 2017.
- [16] L. Pan and D. N. Politis, "Bootstrap prediction intervals for linear, nonlinear and nonparametric autoregressions," *Journal of Statistical Planning and Inference*, vol. 177, pp. 1–27, 2016.
- [17] Z. Shang and J. He, "Confidence-weighted extreme learning machine for regression problems," *Neurocomputing*, vol. 148, pp. 544–550, 2015.
- [18] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Comprehensive review of neural network-based prediction intervals and new advances," *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1341–1356, 2011.
- [19] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Lower upper bound estimation method for construction of neural network-based prediction intervals," *IEEE Transactions on Neural Networks*, vol. 22, no. 3, pp. 337–346, 2011.
- [20] E. Mazloumi, G. Rose, G. Currie, and S. Moridpour, "Prediction intervals to account for uncertainties in neural network predictions: Methodology and application in bus travel time prediction," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 3, pp. 534–542, 2011.
- [21] H. Quan, D. Srinivasan, and A. Khosravi, "Uncertainty handling using neural network-based prediction intervals for electrical load forecasting," *Energy*, vol. 73, pp. 916–925, 2014.
- [22] A. Antoniadis, X. Brossat, J. Cugliari, and J.-M. Poggi, "A prediction interval for a function-valued forecast model: Application to load forecasting," *International Journal of Forecasting*, vol. 32, no. 3, pp. 939–947, 2016.
- [23] C. Lian, Z. Zeng, W. Yao, H. Tang, and C. L. P. Chen, "Landslide Displacement Prediction With Uncertainty Based on Neural Networks With Random Hidden Weights," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 12, pp. 2683–2695, 2016.
- [24] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.
- [25] H. Liu, H.-Q. Tian, and Y.-F. Li, "Four wind speed multi-step forecasting models using extreme learning machines and signal decomposing algorithms," *Energy Conversion and Management*, vol. 100, pp. 16–22, 2015.
- [26] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

- [27] B. Liu, L. Wang, and Y. Jin, "An effective PSO-based memetic algorithm for flow shop scheduling," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 1, pp. 18–27, 2007.
- [28] Z. A. Bashir and M. E. El-Hawary, "Applying wavelets to short-term load forecasting using PSO-based neural networks," *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 20–27, 2009.
- [29] K. Ishaque, Z. Salam, M. Amjad, and S. Mekhilef, "An improved particle swarm optimization (PSO)-based MPPT for PV with reduced steady-state oscillation," *IEEE Transactions on Power Electronics*, vol. 27, no. 8, pp. 3627–3638, 2012.
- [30] P. Vlachos, StatLib Datasets Archive, 2010, <http://lib.stat.cmu.edu/datasets>.
- [31] A. Frank and A. Asuncion, UCI Machine Learning Repository, 2010, <http://archive.ics.uci.edu/ml>.
- [32] H. Saxen and F. Pettersson, "Method for the selection of inputs and structure of feedforward neural networks," *Computers & Chemical Engineering*, vol. 30, no. 6, pp. 1038–1045, 2006.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

