Information Sciences 486 (2019) 119–132

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Construction of prediction intervals for carbon residual of crude oil based on deep stochastic configuration networks☆

Jun Lu, Jinliang Ding*

State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, 110819, China

ARTICLE INFO

Article history:
Received 14 June 2018
Revised 15 February 2019
Accepted 17 February 2019
Available online 20 February 2019

Keywords:
Uncertain data modeling
Deep stochastic configuration networks
Prediction intervals
Nuclear magnetic resonance
Carbon residual content of crude oil

ABSTRACT

Complex components lead to large fluctuations in the physicochemical properties of crude oil which make accurate prediction more difficult. To quantify the potential uncertainty associated with prediction, this paper proposes a novel approach to construct prediction intervals (PIs) for the carbon residual content of crude oil based on the lower-upper bound estimation (LUBE) method and deep stochastic configuration networks (DSCNs). According to the principle of stochastic configuration networks, the input weights and biases of DSCN are randomly assigned with a supervisory mechanism and only the output weights need to be evaluated which can greatly reduce the number of parameters to be optimized. Then, combining the coverage width-based criterion and mean accumulated width deviation, a new cost function of DSCN for constructing PIs based on the LUBE method is proposed to center the mean values of the PIs as near as possible to the targets, hence the average of lower and upper bounds of the PI is calculated as the deterministic output, which can solve the problem that the PIs based on the original LUBE method cannot provide the deterministic prediction. Moreover, a modified backtracking search optimization algorithm, improving the population diversity and increasing the search capability while maintaining the convergence speed, is presented to obtain the optimal PIs. Finally, experiments using real-world data are carried out and the results demonstrate that the proposed approach can construct PIs with high quality.

1. Introduction

The fast evaluation of the physicochemical properties of crude oil plays a key role in the production and optimization of petrochemical processes [6]. The carbon residual (CONR) content is one of the important control indexes in the refining process of crude oil [23]. Traditional laboratory evaluation methods of physicochemical properties of crude oil are time-consuming and do not meet the practical requirements. In recent years, nuclear magnetic resonance (NMR) has been successfully applied in the evaluation of physicochemical properties of crude oil [22,37]. Since NMR belongs to the “secondary instrument” category which needs a calibration model, therefore, the establishment of a prediction model is a key issue in the application of NMR. Multiple linear regression (MLR) [4,21], partial least squares (PLS) [8,19] and back propagation neural networks (BPNNs) [28] have been used to model the physicochemical properties of crude oil.

☆ This paper belongs to the special issue RANN edited by Prof. W. Pedrycz.

* Corresponding author.
E-mail address: jlding@mail.neu.edu.cn (J. Ding).

https://doi.org/10.1016/j.ins.2019.02.042
0020-0255/

With the development of refineries, the capacity of crude oil refining continues to increase, and the geographical origins of crude oil become more complex [3]. While there are differences in components and physicochemical properties of crude oil because of different geographical origins, and even in the same geographical origin, there are also great differences in the components due to different oil formation layers. Frequent fluctuations and uncertain nature of components makes it quite a challenge to accurately predict the physicochemical properties of crude oil. The outputs given by deterministic methods such as MLR and PLS convey no information to evaluate the prediction results, so it is impossible to discriminate the reliability of the prediction results. In addition, the NMR spectra of crude oil are extremely similar to each other, hence it is difficult to distinguish one from another. Therefore, high-performance modeling methods are needed to effectively predict the corresponding physicochemical properties of different spectra.

Prediction intervals (PIs) can provide a better solution for uncertain data modeling [29]. PIs construct reasonable intervals which cover the actual targets with a certain confidence level. The PIs consist of a coverage probability and interval width which can provide more information on the targets for decision-makers [1]. Traditional methods such as delta [17], bootstrap [14], Bayesian [18] and mean-variance estimation (MVE) [20] are widely used for constructing PIs based on neural networks (NNs). The main drawbacks of these traditional methods are that the noise is assumed to be of Gaussian distribution and there is an expensive computational cost for a large dataset [9,12,13,30]. In these methods, once the assumption with respect to noise does not hold, it will lead to unreliable PIs [16]. The lower-upper bound estimation (LUBE) method proposed in [13] does not need to make any special assumption about data distribution, and the constructed PIs based on the LUBE method are more reliable [24]. But the shortcoming of the LUBE method is that it only provides PIs for the targets without giving deterministic prediction results. In the LUBE method, the PI-based cost function for training the NNs consists of the coverage probability and prediction interval width instead of the error-based function [10]. Since the PI-based cost function is nonlinear and complex, traditional gradient-based methods cannot be applied. Hence, intelligent optimization algorithms are adopted to train the NN, such as particle swarm optimization (PSO) [26]. PIs based on the LUBE method have been widely used in various industrial fields, e.g., wind power forecasting [25], but there are few examples in the field of prediction of the physicochemical properties of crude oil. Consequently, we propose PIs for the physicochemical properties of crude oil in this paper. This method can characterize the reliability of the prediction results and provide uncertain information associated with the prediction for decision-makers, and it is convenient for decision-makers to determine the prediction results in a reasonable manner.

In the original LUBE method, all parameters of NNs, including the input weights, biases and output weights, need to be turned, which leads to a slow training process. To address this issue, randomized methods for training networks have developed [2,27,31,32,34,36]. In this paper, the DSCN proposed in [33] is employed to implement the LUBE method. The input weights and biases of DSCN are generated randomly and selected using a data-dependent inequality supervisory mechanism reported in [33] which are kept fixed in the training process. Moreover, the scope setting of input weights and biases are determined adaptively. More details about the impact of scope setting on the performance of neural networks based on randomized methods can refer to [15]. The supervisory mechanism suggested in [33] can avoid producing junk nodes and improve the learning capability of DSCN. The deep structure of DSCN can increase its expressive power, and the random characteristic can reduce the number of parameters to be optimized and decrease the computational cost. In addition, the inequality supervisory mechanism of DSCN for the selection of random parameters can improve the prediction performance which can also improve the quality of the constructed PIs.

Therefore, from the application viewpoint, the PIs of carbon residual content based on the LUBE method are developed and the DSCN is adopted for the implementation of the LUBE method. A new cost function for constructing PIs based on the LUBE method is proposed. The constructed PIs using the proposed cost function can provide both the prediction intervals and deterministic prediction values. Due to the high nonlinearity and complexity of the proposed cost function, a modified backtracking search optimization algorithm (BSA) is employed to train the DSCN for constructing PIs. In the modified BSA (MBSA), a dynamic updating strategy of the evolutionary amplitude control factor and an adaptive mutation operation are proposed to guide the evolutionary direction and speed up the convergence. Meanwhile, *Levy flights* is introduced to obtain better population diversity and avoid being trapped in local optima. Experiments are conducted based on a data set collected from a refinery and the results demonstrate the effectiveness of the proposed method for constructing PIs.

The rest of this paper is organized as follows. In Section 2, the evaluation system of physicochemical properties of crude oil based on NMR is presented and the problem of PI construction for the physicochemical properties is described. The proposed method of constructing PIs and a modified backtracking search optimization algorithm (MBSA) are given in Section 3. In Section 4, experiments using practical data are carried out and the results are discussed. Lastly, our conclusion is given in Section 5.

2. Problem description

2.1. Problem description of prediction of the physicochemical properties of crude oil

The compounds in crude oil can be represented by combinations of different chemical shifts of the NMR spectrum and the content can be calculated by the absorption peak intensity at the corresponding chemical shift. However, a majority of the physicochemical properties of crude oil are influenced by different compounds, which leads to strong coupling between chemical shifts of different physicochemical properties. Hence, the absorption peak intensity is the superposition of

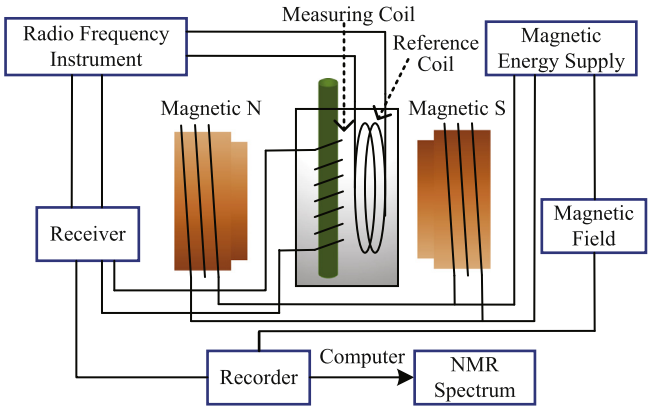

Fig. 1. Schematic diagram of NMR.

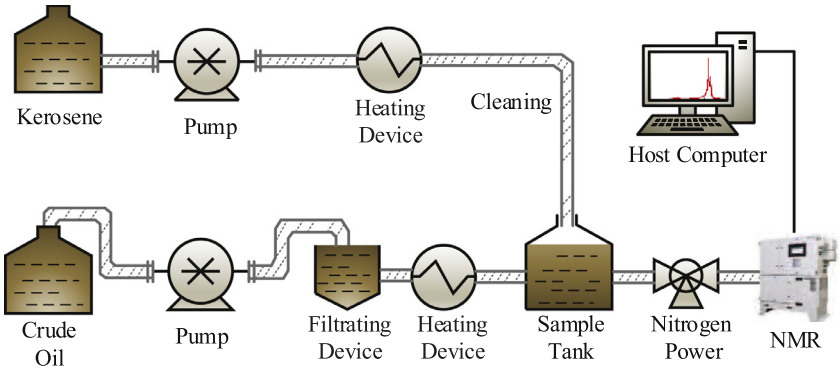

Fig. 2. Structure of fast evaluation system of crude oil based on NMR.

absorption peaks of different physicochemical properties at the corresponding chemical shift. Therefore, it is difficult to find the relationship between physicochemical properties and the NMR spectrum based on the first principles. In addition, the complex components of crude oil lead to large fluctuations in the physicochemical properties. Consequently, data-based PIs of the physicochemical properties of crude oil are proposed in this paper.

2.2. Brief introduction of a fast evaluation system of crude oil based on NMR

Nuclear magnetic resonance (NMR) is a physical phenomenon of the interaction between the alternating magnetic field and the substance. In this process, the resonance frequency and corresponding energy level transition of the nucleus are different from each other, and the chemical shift and absorption peak intensity reflected in the NMR spectrum are also different. According to these differences, it is possible to determine the specific elements contained in the substance, as well as the concentration and molecular structure of these elements. Then, we can undertake a qualitative and quantitative analysis of the compositions in the substance. The schematic diagram of NMR is shown in Fig. 1.

The structure of a fast evaluation system of crude oil based on NMR in a refinery in south China is shown in Fig. 2. The system mainly consists of filtrating devices, heating devices, a NMR and a host computer. First, the kerosene is used to purge the pipeline where the crude oil samples are imported, so as to ensure that there are no impurities in the pipeline. Then, crude oil samples are simply filtered and heated to 70 centigrade. The kerosene inside the pipeline is discharged by the power provided by the nitrogen valve, and the crude oil samples are pressed into the pipeline and NMR. The analysis results obtained from NMR are uploaded to the host computer through the communication system, and then the NMR hydrogen spectrum is transformed into sequential data with 700 dimensions.

The online Aspect Imaging AI-60 NMR is adopted in this system, the operating frequency is 60 MHz and the chemical shift reference is tetramethylsilane (TMS). The typical NMR spectrum of crude oil is shown in Fig. 3 where the abscissa represents the relative chemical shift, and the ordinate indicates the intensity of the hydrogen absorption peak at the corresponding relative chemical shift.

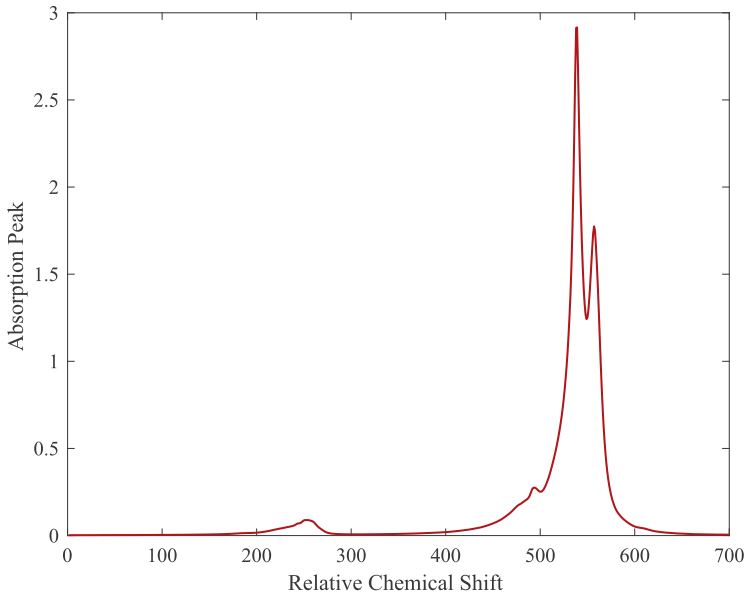

Fig. 3. The NMR spectrum of crude oil.

3. PI-based estimation of carbon residual content

From the above analysis, it is known that PLS, MLR and BPNN are applied to estimate the physicochemical properties of crude oil. But these methods cannot quantify the uncertainty associated with the prediction. Therefore, we implement prediction intervals to model the carbon residual of crude oil and the details are given as follows.

3.1. Input and output selection of PIs

According to the fast evaluation system of the physicochemical properties of crude oil based on NMR and the modelling objective, the input vector of PIs is selected as the NMR hydrogen spectrum data $\boldsymbol{x} \in R^{700}$. Principal component analysis (PCA) is implemented to reduce the dimension of the NMR hydrogen spectrum data, and the principal components with 95% cumulative percent variance rate are selected. The outputs of PIs are the carbon residual content of crude oil $y \in R$ and its lower and upper bounds.

3.2. Definition of PIs

Given a data set $(\boldsymbol{x}_i, y_i) \in R^d \times R,\ i = 1, \ldots, N$, $\boldsymbol{x}_i$ denotes the NMR hydrogen spectrum data and y_i denotes the carbon residual in this paper. The PI of y_i with the confidence level $100(1-\alpha)\%$ is defined as [11]:

$$\mathrm{PI}(\boldsymbol{x}_i) = [L(\boldsymbol{x}_i), U(\boldsymbol{x}_i)]. \tag{1}$$

The prediction interval $\mathrm{PI}(\boldsymbol{x}_i)$ covers the target with probability $100(1-\alpha)\%$

$$Pr(L(\boldsymbol{x}_i) \le y_i \le U(\boldsymbol{x}_i)) = 100(1-\alpha)\%, \tag{2}$$

where $Pr(\cdot)$ denotes the probability, $L(\boldsymbol{x}_i)$ and $U(\boldsymbol{x}_i)$ are the lower and upper bounds of $\mathrm{PI}(\boldsymbol{x}_i)$, respectively.

3.3. Evaluation indices of PIs

The PI coverage probability (PICP) refers to the probability that the target is covered by the constructed PI. PICP is one of the most important indices of the constructed PI, and it can be defined as [13]:

$$\mathrm{PICP} = \frac{1}{N}\sum_{i=1}^{N} B_i, \tag{3}$$

where N is the number of samples and B is a Boolean variable:

$$B_i = \begin{cases} 1 & \text{if } \ y_i \in [L(\boldsymbol{x}_i), U(\boldsymbol{x}_i)] \\ 0 & \text{if } \ y_i \notin [L(\boldsymbol{x}_i), U(\boldsymbol{x}_i)]. \end{cases} \tag{4}$$

PICP is used to evaluate the reliability of the constructed PIs. If PICP is much less than $100(1-\alpha)\%$, this means that the PIs are invalid and should be reconstructed.

When the constructed PIs have extremely large interval width, for example, the minimum and maximum values of targets are chosen as the lower and upper bounds, we can achieve a high PICP. But these constructed PIs reflect no information about the targets and cannot be used for decision making. Therefore, another index called the mean prediction interval width (MPIW) is introduced to quantify the effectiveness of the constructed PIs. And the MPIW is defined as [13]:

$$\text{MPIW} = \frac{1}{N}\sum_{i=1}^{N}(U(\boldsymbol{x}_i) - L(\boldsymbol{x}_i)). \tag{5}$$

The normalization of MPIW can objectively compare PIs that are constructed by different techniques. The normalized MPIW (NMPIW) is given as follows:

$$\text{NMPIW} = \frac{1}{R}\left(\frac{1}{N}\sum_{i=1}^{N}(U(\boldsymbol{x}_i) - L(\boldsymbol{x}_i))\right), \tag{6}$$

where R is the range of targets and $R = y_{\max} - y_{\min}$. There is a direct relationship between PICP and NMPIW. In general, a wider prediction interval can result in higher coverage probability. But in practical applications, PICP and NMPIW are two conflicting indices. The optimal PIs should have a small NMPIW while maintaining a high PICP, therefore, the multi-objective problem is to maximize PICP and minimize NMPIW simultaneously, and the constraints are $0 \le \text{PICP} \le 100\%$ and $\text{NMPIW} > 0$.

3.4. Optimization objective function

PICP and NMPIW only appraise one aspect of the PIs. In order to measure the impacts of PICP and NMPIW simultaneously, a comprehensive objective function combined PICP and NMPIW based on the modified coverage width-based criterion (CWC) is proposed in [16]. Then the multi-objective optimization problem can be formulated as a single objective problem:

$$\begin{aligned} \min\ & \text{CWC}_\text{M} = (\text{NMPIW} + \psi)\cdot \mathrm{e}^{(\text{PICP}-\mu)^2/(2\sigma^2)} \\ s.t.\ & 0 \le \text{PICP} \le 100\% \\ & \text{NMPIW} > 0, \end{aligned} \tag{7}$$

where ψ is a positive number which is much smaller than NMPIW, $\mu = 100(1-\alpha)\%$ is the predefined confidence level, σ is a positive number which is smaller than 1.

The aforementioned PI indices take no consideration of targets that lie outside the constructed PIs. When the targets are not covered by the PIs, we hope that the lower or upper bounds of PIs will approach the targets and finally cover the targets. In addition, in the decision-making process of crude oil refining, it is not only necessary to consider the uncertainty associated with the prediction of physicochemical properties, but also a deterministic analysis should be given. Therefore, for the targets that fall within the constructed PIs, we hope to make the center of the prediction interval as near as possible to the target value, so that the average values of the lower and upper bounds of the PIs can be used as the deterministic prediction results. Consequently, the mean accumulated width deviation (MAWD) is proposed and it can be defined as:

$$\text{MAWD} = \frac{1}{N}\sum_{i=1}^{N} WD_i, \tag{8}$$

where the width deviation (WD) is given by:

$$WD_i = \begin{cases} \frac{L(\boldsymbol{x}_i)-y_i}{U(\boldsymbol{x}_i)-L(\boldsymbol{x}_i)}, & \text{if } y_i < L(\boldsymbol{x}_i) \\ \frac{|y_i-\hat{y}_i|}{U(\boldsymbol{x}_i)-L(\boldsymbol{x}_i)} & \text{if } y_i \in [L(\boldsymbol{x}_i), U(\boldsymbol{x}_i)] \\ \frac{y_i-U(\boldsymbol{x}_i)}{U(\boldsymbol{x}_i)-L(\boldsymbol{x}_i)}, & \text{if } y_i > U(\boldsymbol{x}_i), \end{cases} \tag{9}$$

where $\hat{y}_i = 0.5 \times (U(\boldsymbol{x}_i) + L(\boldsymbol{x}_i))$ is used as the deterministic prediction.

According to the above analysis, one can see that the optimal PIs should have higher PICP, smaller NMPIW and MAWD. Therefore, the comprehensive objective function of constructing PIs using the LUBE method is formulated as:

$$\begin{aligned} \min\ & \text{F} = \lambda\cdot\text{CWC}_\text{M} + \gamma\cdot\text{MAWD} \\ s.t.\ & 0 \le \text{PICP} \le 100\% \\ & \text{NMPIW} > 0, \end{aligned} \tag{10}$$

where λ and γ denote the weight of CWC_M and $MAWD$, respectively.

Remark 1. The influence of CWC_M and $MAWD$ criteria on the optimization results can be regulated by adjusting the weight coefficients λ and γ. In practice, we generally keep the coefficient λ fixed and tune the coefficient γ or vice versa, and the CWC_M and $MAWD$ should have the same order of magnitude by tuning λ and γ.

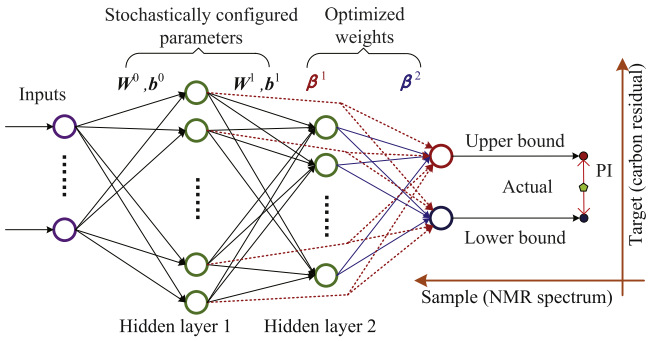

Fig. 4. Schematic diagram of the DSCN (two hidden layers) for the implementation of LUBE method.

3.5. LUBE method based on deep stochastic configuration networks

3.5.1. Deep stochastic configuration networks

The deep stochastic configuration network is built in a constructive way based on the stochastic configuration algorithm proposed in [35]. Different from the traditional multi-layer perceptron, each hidden node of DSCN is connected to the output layer [33]. Given a data set $\{\boldsymbol{x}, \boldsymbol{y}\} = (\boldsymbol{x}_i, y_i) \in R^d \times R,\ i = 1, \ldots, N$, for a DSCN with n layers and $(L_n - 1)$ hidden nodes within each layer, $\boldsymbol{\varepsilon}^n_{L_n-1}(\boldsymbol{x}) = [\varepsilon^n_{L_n-1}(\boldsymbol{x}_1), \varepsilon^n_{L_n-1}(\boldsymbol{x}_2), \ldots, \varepsilon^n_{L_n-1}(\boldsymbol{x}_N)]^T$ denotes the residual error vector. If the error does not meet an expected tolerance, then an L_nth hidden node is added and let:

$$h^n_{L_n}(\boldsymbol{w}^{n-1}_{L_n}, b^{n-1}_{L_n}) = \left\{ G\left[\left(\boldsymbol{w}^{n-1}_{L_n}\right)^T \boldsymbol{x}^{n-1}_1 + b^{n-1}_{L_n} \right], \ldots, G\left[(\boldsymbol{w}^{n-1}_{L_n})^T \boldsymbol{x}^{n-1}_N + b^{n-1}_{L_n} \right] \right\}^T, \tag{11}$$

where $\boldsymbol{w}^{n-1}_{L_n}$, $b^{n-1}_{L_n}$ and $\boldsymbol{h}^n_{L_n}(\cdot)$ denote the input weights, biases and output matrix of the L_nth hidden node within the nth layer, $G(\cdot)$ is the activation function, $\boldsymbol{x}^0_i = \boldsymbol{x}_i$ and $\boldsymbol{x}^{n-1}_i = [G(\boldsymbol{w}^{n-1}_1 \cdot \boldsymbol{x}^{n-2}_i + b^{n-1}_1), \ldots, G(\boldsymbol{w}^{n-1}_{L_n} \cdot \boldsymbol{x}^{n-2}_i + b^{n-1}_{L_n})]$ for $n \geq 2$.

The randomly generated parameters ($\boldsymbol{w}^{n-1}_{L_n}$ and $b^{n-1}_{L_n}$) should meet the following supervisory mechanism [33]:

$$\xi = \frac{\left\langle \boldsymbol{\varepsilon}^n_{L_n-1}(\boldsymbol{x}), \boldsymbol{h}^n_{L_n}(\boldsymbol{w}^{n-1}_{L_n}, b^{n-1}_{L_n}) \right\rangle^2}{\left\langle \boldsymbol{h}^n_{L_n}(\boldsymbol{w}^{n-1}_{L_n}, b^{n-1}_{L_n}), \boldsymbol{h}^n_{L_n}(\boldsymbol{w}^{n-1}_{L_n}, b^{n-1}_{L_n}) \right\rangle} - (1 - r - \mu_L) \times \left\langle \boldsymbol{\varepsilon}^n_{L_n-1}(\boldsymbol{x}), \boldsymbol{\varepsilon}^n_{L_n-1}(\boldsymbol{x}) \right\rangle > 0, \tag{12}$$

where $0 < r < 1$, $0 < \mu_L < 1 - r$ and $\lim_{L\to\infty} \mu_L = 0$, $\langle \cdot, \cdot \rangle$ denotes the scalar product.

Remark 2. As described in [33], we know that $T_{\max}$ hidden nodes are produced and their random parameters are generated in the varying range $[-\omega_k, \omega_k]$, $k = 1, 2, \ldots, K$, the hidden nodes which meet the supervisory mechanism $\xi > 0$ are selected as the candidates, and finally the node with the largest ξ is chosen as the newly added one.

3.5.2. DSCN for LUBE method

The LUBE method was proposed to construct PIs in [13]. The key idea of the LUBE method is to build a NN with two outputs which correspond to the lower and upper bounds of the PIs. The traditional multilayer feedforward neural networks are always adopted in the LUBE method, but the input weights, biases and output weights all need to be optimized which leads to excessive decision variables in the optimization process. Therefore, the DSCN is employed to implement the LUBE method for constructing PIs, and the structure (with two hidden layers) is shown in Fig. 4. The input weights and biases of DSCN are randomly produced and then assigned based on an inequality supervisory mechanism, only the output weights need to be trained which can reduce the number of parameters to be optimized.

Given a data set $\{\boldsymbol{x}, \boldsymbol{y}\} = (\boldsymbol{x}_i, y_i) \in R^d \times R,\ i = 1, \ldots, N$, the DSCN for constructing PIs based on LUBE method can be formulated as:

$$O_\kappa(\boldsymbol{x}_i) = \sum_{k=1}^{n} \sum_{j=1}^{L_n} \beta_{(\kappa)}{}^k_j G\left(\boldsymbol{w}^{k-1}_j \cdot \boldsymbol{x}^{k-1}_i + b^{k-1}_j\right), \tag{13}$$

where $\kappa = 1, 2$, O_1 and O_2 denote the upper and lower bounds of the PIs, respectively. $\boldsymbol{w}^{k-1}_j$, b^{k-1}_j and $\beta_{(\kappa)}{}^k_j$ are the input weights, biases and output weights of the jth node within the kth layer.

The objective function of DSCN for constructing PIs based on the LUBE method is designed as follows:

$$\begin{aligned} \min_{\beta} \quad & F_{\mathrm{DSCN}} = \lambda \cdot CWC_M + \gamma \cdot \mathrm{MAWD} \\ s.t. \quad & U(x_i) = O_1(x_i) \\ & L(x_i) = O_2(x_i) \\ & 0 \le \mathrm{PICP} \le 100\% \\ & \mathrm{NMPIW} > 0. \end{aligned} \tag{14}$$

The objective function Eq. (14) is used to train DSCN for constructing PIs. The parameters (input weights, biases, the number of hidden nodes and hidden layers) of DSCN should be optimally determined. Hence, before constructing the PIs, the DSCN is built based on the inequality supervisory mechanism Eq. (12) to find the optimal parameters. Then the obtained parameters are used as the initial parameters of DSCN for constructing PIs based on the LUBE method. For the DSCN implemented in the LUBE method, we name the training process of DSCN based on the inequality supervisory mechanism as pre-training. Therefore, the implementation of DSCN for constructing PIs is summarized in the following steps:

(1) The DSCN is trained based on the supervisory mechanism and the cost function is the error-based function (pre-training).
(2) The parameters (input weights, biases, the number of hidden nodes and hidden layers) obtained from the pre-training are used as the initial parameters of DSCN for constructing PIs and keep fixed in the training process. Only the weights of the two output nodes need to be tuned using cost function Eq. (14), and the two output nodes represent the upper and lower bounds of the PIs.

3.6. Modified backtracking search optimization algorithm

Since the objective function Eq. (14) of DSCN for constructing PIs is complex and nonlinear, the classical derivative-based methods cannot be applied. Therefore, a modified backtracking search optimization algorithm (MBSA) is proposed to optimize the output weights of DSCN to construct optimal PIs. Each individual of MBSA stands for a set of candidate output weights of DSCN. So, the output weights of DSCN are coded in the following way:

$$\left(\underbrace{\beta_{(1)1}^{1}, \beta_{(1)2}^{1}, \beta_{(1)3}^{1}, \ldots, \beta_{(1)n\times L_n}^{n}}_{\text{output weights of } O_1 \text{ (upper bound)}}, \underbrace{\beta_{(2)1}^{1}, \beta_{(2)2}^{1}, \beta_{(2)3}^{1}, \ldots, \beta_{(2)n\times L_n}^{n}}_{\text{output weights of } O_2 \text{ (lower bound)}} \right). \tag{15}$$

The proposed modified backtracking search optimization algorithm is explained in detail as follows.

3.6.1. Original backtracking search optimization algorithm

BSA [7] is a population-based algorithm for solving optimization problems. The main advantage of BSA is that it has fewer control parameters and the historical population is retained to provide a good evolutionary direction. The process of BSA mainly consists of population initialization, historical population selection, mutation, crossover and new population selection.

The initialization of BSA is the same as the differential evolution (DE) which generates the initial population randomly. The difference is that the initialization of BSA includes the current population P and the historical population $oldP$:

$$\begin{cases} P_{i,j} \sim \mathrm{U}(low_j, up_j) \\ oldP_{i,j} \sim \mathrm{U}(low_j, up_j), \end{cases} \tag{16}$$

where $i = 1, 2, \ldots, N$, N is the population size, $j = 1, 2, \ldots, D$, D is the dimension of decision variable, $\mathrm{U}(\cdot)$ denotes the uniform distribution, low and up represent the lower and upper scopes of decision variables, respectively.

The historical population is updated randomly, and then the individual of $oldP$ are ranked randomly:

$$\begin{cases} oldP = P, & \text{if } a < b \\ oldP = permuting(oldP), \end{cases} \tag{17}$$

where a and b are uniformly distributed random numbers in [0, 1], $permuting(\cdot)$ denotes the random shuffling operator.

The mutation of BSA is defined as:

$$M = P + F_a \cdot (oldP - P), \tag{18}$$

where F_a denotes the amplitude control factor of the evolution.

The crossover of BSA generates the trial population T and the process is divided into two parts. In the first crossover strategy, an NP^*D matrix denoted by Map is produced which only contains elements 0 and 1. This Map matrix controls which dimension of the individuals are chosen to perform the crossover. In the second crossover strategy, only one randomly

chosen dimension of each individual participates into the crossover. Details on the crossover process can refer to [7]. And the crossover process is expressed as follows:

$$T_{i,j} = \begin{cases} P_{i,j} & \text{if } Map_{i,j} = 1 \\ M_{i,j} & \text{otherwise.} \end{cases} \tag{19}$$

In the selection stage of BSA, the new population is updated according to the fitness value using the greedy selection strategy, and the global minimizer is updated based on the best individual:

$$if \quad fit(T_i) < fit(P_i) \Rightarrow \begin{cases} P_i = T_i \\ fit(P_i) = fit(T_i), \end{cases} \tag{20}$$

where $fit(\cdot)$ denotes the fitness function value. The above process is repeated until some termination conditions are satisfied, and the optimal solutions are obtained.

3.6.2. Modification of backtracking search optimization algorithm

In the original BSA, the amplitude control factor F_a is equal to a random number obeying the standard normal distribution. To improve the convergence characteristic of BSA, in the modified BSA (MBSA), we propose a dynamic updating strategy for the amplitude control factor:

$$F_a = (3 \cdot s)^{\frac{1}{s}} \cdot r_n, \tag{21}$$

where s denotes the current iteration number and r_n denotes a random number obeying the standard normal distribution.

In addition, the mutation operation is the most important part of BSA. In the mutation operation of BSA, the historical population is used to guide the evolution direction. But this simple mutation may slow the convergence speed of BSA and it may not find the best solution. Inspired by the social learning PSO [5], a new adaptive mutation operation is proposed:

$$M = \begin{cases} P + F_a \cdot (oldP - P), & \text{if } c < d \\ P + F_a \cdot (oldPb - P), & \text{otherwise.} \end{cases} \tag{22}$$

where c and d are uniformly distributed random numbers in [0, 1].

The *oldPb* is generated as follows. First, the individuals of the historical population are sorted in descending order according to their fitness values, i.e., the individual with the largest fitness is placed firstly while the individual with the smallest fitness is placed at the end. Then, a given individual i ($1 \le i < N$) is randomly learning from a better (with a smaller fitness value) individual k ($i < k \le N$):

$$oldPb_i = oldP_i + \alpha \cdot (oldP_k - oldP_i), \tag{23}$$

where $0 < \alpha < 1$. In this paper, α is set as 0.3. By using the new adaptive mutation operation, both the individual and the better individual of the historical population are used to guide the evolution of the population with equal probability, which can avoid the blindness in the evolutionary process.

Due to the high dimension of parameters to be optimized in the construction of PIs, the diversity of the population is extremely important. In the optimization problem, *Levy flights* is an effective way to improve the diversity of a population. *Levy flights* obeys a heavy-tailed distribution. In this paper, *Levy flights* is adopted to produce another trial population which is defined as follows:

$$\begin{cases} Levy(v) \sim \tau = s^{-v}, \quad 1 < v \le 3 \\ Popt_i = P_i + \alpha_1 \oplus Levy(v), \end{cases} \tag{24}$$

where $\alpha_1 > 0$ and $\oplus$ is the entry-wise multiplication. The best individual is retained according to the fitness function values of P and the two trial populations *T, Popt*.

3.7. Procedure of constructing PIs based on the proposed method

According to the above discussion, the detailed implementation procedure of the proposed method for constructing PIs is summarized as follows:

Step 1: The collected data set is split into three parts: training data set, validation data set and testing data set.
Step 2: Initialization of structure parameters (the maximum number of hidden layers and hidden nodes of each layer) and the varying scope $\Omega = [\omega_{\min} : \omega : \omega_{\max}]$ of random parameters (input weights and biases).
Step 3: In light of the inequality supervisory mechanism Eq. (12) and the early stopping method, the optimal structure parameters are determined based on the results of the validation data set. Then the DSCN can be built using the optimal structure parameters.
Step 4: The structure parameters and random parameters obtained from *step 3* are set as the initial parameters of DSCN for constructing PIs.
Step 5: The initial output weights are generated randomly and are coded in way of Eq. (15).

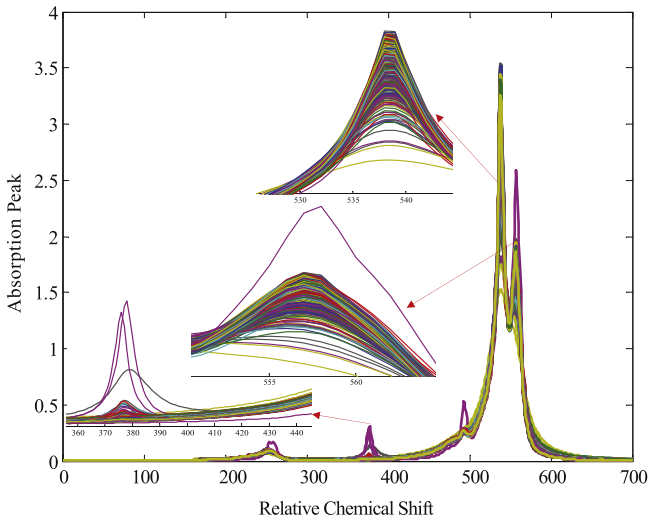

Fig. 5. The 776 groups of NMR hydrogen spectra.

Step 6: Based on the cost function Eq. (14), the proposed MBSA is applied to optimize the output weights of DSCN for constructing PIs.
Step 7: Given the testing data and output the prediction result.

Based on the above implementation procedure, the pseudo code of the proposed method is summarized as Algorithm 1.

Algorithm 1 The proposed method for constructing PIs.

Input: the training and validation data (NMR spectra and carbon residual), the predefined confidence level μ.

Pre-training of DSCN for constructing PIs: parameters (input weights, biases, the number of hidden layers and hidden nodes) of DSCN are adjusted using supervisory mechanism Eq. (12) on the training and validation data;
Initialization of DSCN for constructing PIs: the parameters obtained from *step 1* are set as the initial parameters of DSCN for constructing PIs, assign output weights randomly and code them in way of Eq. (15);
Initialization of MBSA: the population size *NP*, decision variable dimension *D*, maximum number of iterations *max_s*, and other parameters listed in Table I, initialize the *P* and *oldP* using Eq. (16);
for $s = 1$ to max_s **do**
    oldP selection using Eq. (16);
    for $i = 1$ to *NP* **do**
        Calculate the outputs of DSCN, PICP, NMPIW and MAWD using Eq. (13), Eq. (3), Eq. (6) and Eq. (8);
        Calculate the fitness of *oldP* using the objective function Eq. (14);
    end for
    Sort the individual of *oldP* according to fitness;
    Construct the *oldPb* based on Eq. (23) and conduct mutation using Eq. (18);
    Generate populations *T*, *Popt*, new *P* and update the global minimizer using Eq. (22), Eq. (24) and Eq. (20);
end for
Output the final solutions as the optimal output weights of DSCN;
Given the new NMR spectrum and calculate the outputs of DSCN using Eq. (13) and construct PIs;

4. Experimental results

4.1. Data set

In this paper, 776 groups of NMR hydrogen spectra as shown in Fig. 5 and the corresponding carbon residual content of crude oil were collected from a refinery in south China between May 2016 and March 2017.

The collected NMR hydrogen spectra are obtained from the Aspect Imaging AI-60 online NMR. The hydrogen spectrum has been simply tuned by the NMR. In industrial processes, the data usually arrive in time sequence. Therefore, in view of practical application, the data set is split into three parts in the order of acquisition time, of which 646 groups are used as the training data, 65 groups are used as the validation data and the remaining 65 groups are used for evaluating the constructed PIs. The validation dataset is applied to determine the hyperparameters of the proposed method.

Table 1
Parameters of the proposed algorithm.

Target	Parameter	Numerical value
Objective function F_{DSCN}	ψ	0.01%
	σ	0.15
	λ	1
	γ	0.05
MBSA	α_1	0.8

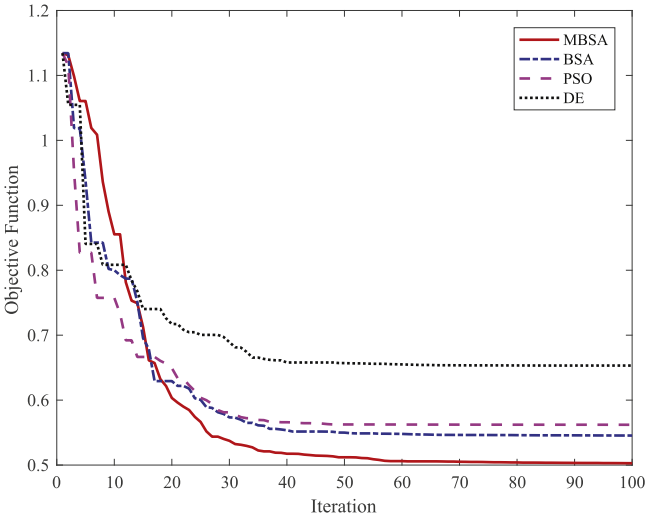

Fig. 6. Convergences of MBSA, BSA, PSO and DE for objective function optimization on the training data.

4.2. Parameter selection

In this paper, the predefined confidence level of the PIs is set to $\mu = 95\%$, the parameters of the proposed method are determined according to results of the validation data set. The sigmoid function is selected as the activation function of DSCN. The number of hidden layers and the number of hidden nodes of DSCN are selected according to the results of the validation data set. The input weights and hidden biases of DSCN are determined in the pre-training process described in Section 3.5. In the pre-training process of DSCN for constructing PIs based on the LUBE method, the scope of the random parameters is automatically selected in the set $[-\omega, \omega], \omega = 1, 2, 4, 8, 16$, and the initial value of r is set to 0.9, the maximum configuration time $T_{\max}$ of the newly added candidate nodes is set to 200, the number of hidden layers is set to $n = 2$ and the number of nodes within each hidden layer is set to $L_n = 8$. The population size (NP), the dimension of decision variables (D) and the maximum iteration (max_s) of the MBSA are set to $NP = 300$, $D = 2 \times 2 \times L_k = 32$ and max_$s = 100$, respectively. The other parameters are reported in Table 1. All the algorithms in the experiments are programmed in MATLAB and run on a computer with 3.4 GHz CPU.

4.3. Comparison and discussion

All the experiments in this paper are repeated thirty times, and the average value of the thirty experiments' results are reported. To illustrate the performance of the proposed MBSA, experiments are carried out to compare the MBSA with the original BSA and two typical optimization algorithms: PSO and differential evolution algorithm (DE). For a fair comparison, the four optimization algorithms are considered to have the same population size, maximum iteration and the same initial populations. The convergence characteristics of MBSA, BSA, PSO and DE for optimizing the objective function F_{DSCN} are shown in Fig. 6. It can be observed that compared with BSA, PSO and DE, the proposed MBSA can obtain the optimal value of the objective function and the convergence speed is also guaranteed. This result demonstrates that the MBSA can obtain better solutions of the decision variables than the other algorithms.

The variations of PICP, NMPIW and the objective function value with respect to the iteration of MBSA are shown in Fig. 7. It can be seen that in the initial stage of the MBSA iteration, to minimize the objective function F_{DSCN} and improve the PICP, the NMPIW will increase. After a few iterations, the PICP is close to the predefined confidence level μ, then the NMPIW and the objective function F_{DSCN} will decrease with respect to the MBSA iteration. With the increase of the iteration, the PICP and NMPIW iterate in such way, and the objective function F_{DSCN} decreases continuously until the PICP and NMPIW come to a relatively balanced stable point.

Table 2
Results of different optimization algorithms on the testing data.

Algorithm	PICP (%)	NMPIW
DE	93.9628	0.3611
PSO	95.8204	0.3369
BSA	96.7492	0.3262
MBSA	98.4615	0.3096

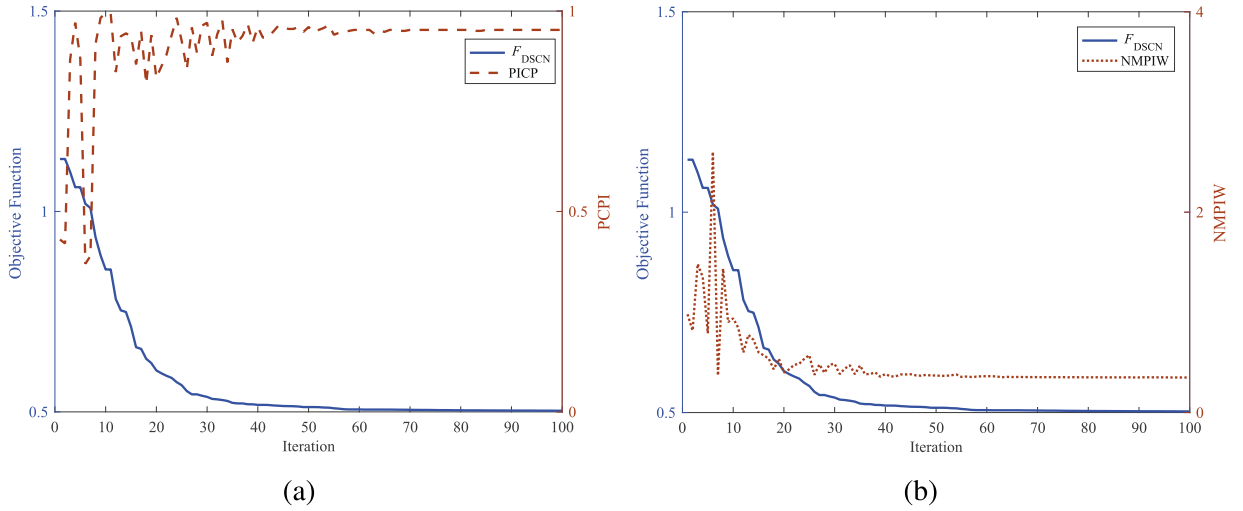

Fig. 7. Variations of PICP (a), NMPIW (b) and F_{DSCN} with respect to MBSA iteration on the training data.

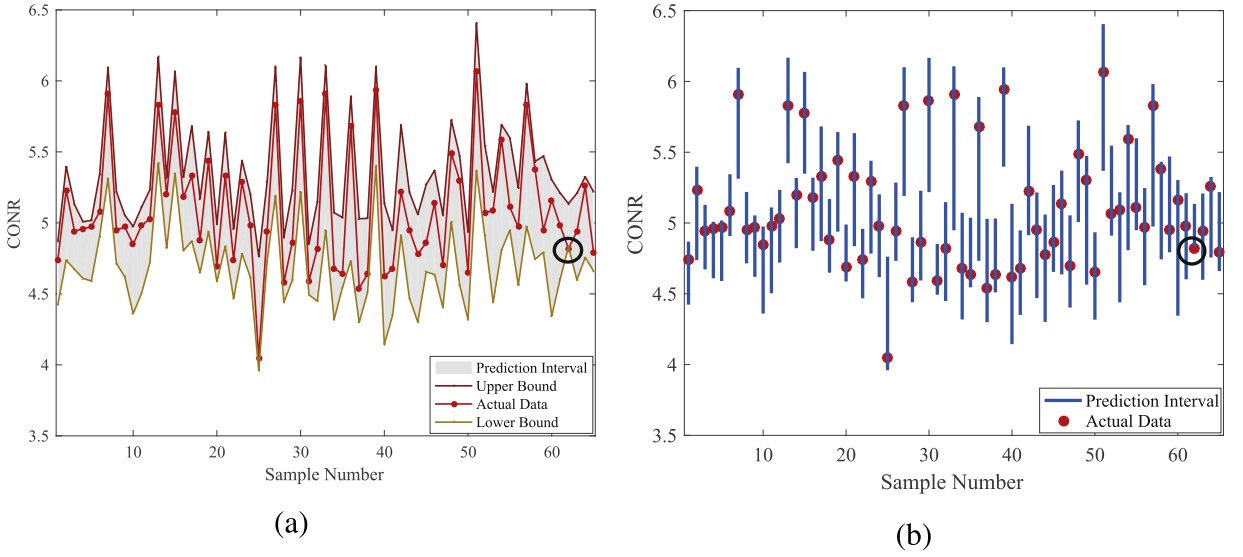

Fig. 8. The proposed PIs for CONR on the testing data.

The PICPs and NMPIWs of MBSA, BSA, PSO and DE for the optimization task are given in Table 2. According to the results in Table 2, one can see that the proposed MBSA can increase the coverage probability while decreasing the NMPIW and improving the sharpness of the constructed PIs.

The constructed PIs for the CONR using the proposed method are shown in Fig. 8 where Fig. 8(b) is better for visualization. As shown in Fig. 8, the proposed PIs can cover almost all the targets except the circled one. These results demonstrate that the proposed method can construct PIs with high quality.

Fig. 9 shows the deterministic prediction results and prediction errors provided by the proposed PIs. As shown in Fig. 9, we can see that the prediction outputs which are computed by averaging the lower and upper bounds of the PIs can fit

Table 3
Evaluation indices of PIs using different methods on the testing data.

Method	PICP (%)	NMPIW
O-LUBE	93.8461	0.3269
M-LUBE	95.3846	0.3358
The proposed method	98.4615	0.3096

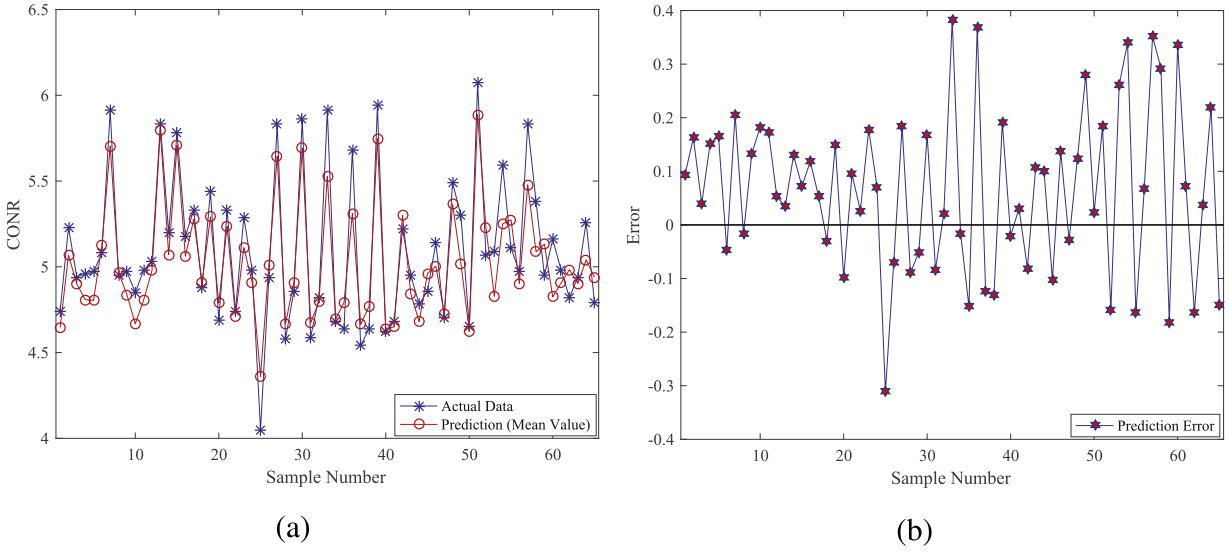

Fig. 9. Deterministic prediction results (a) and prediction errors (b) of the proposed PIs for CONR on the testing data.

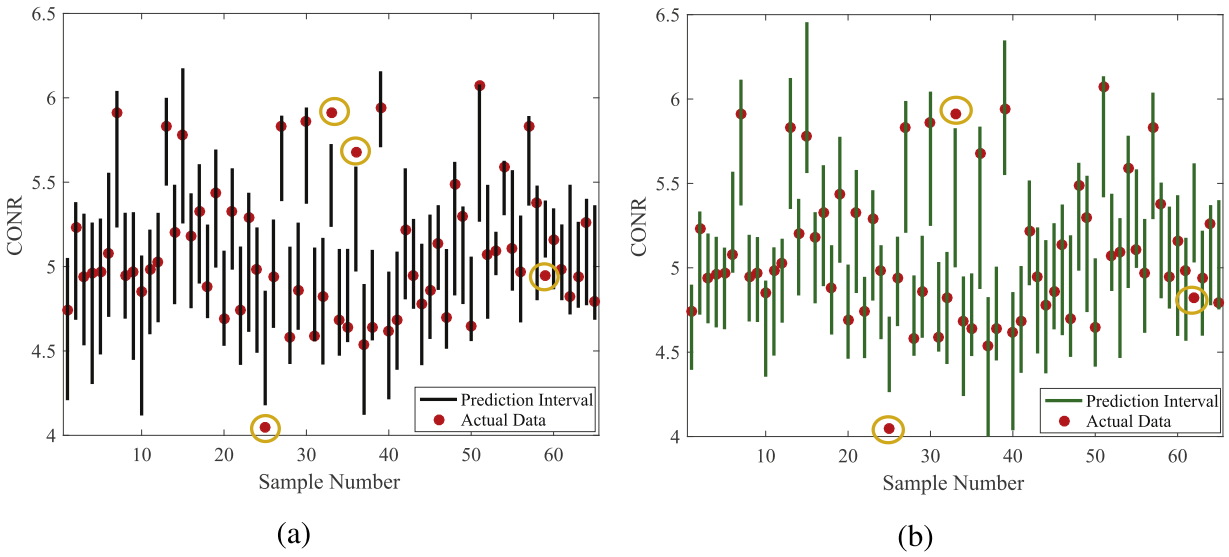

Fig. 10. The constructed PIs based on the O-LUBE (a) and M-LUBE (b) on the testing data.

the targets better with small errors. The results of the experiment illustrate that the constructed PIs based on the proposed objective function can provide effective deterministic prediction results to some extent.

To further examine the effectiveness of the proposed PIs, the PIs for the CONR are constructed based on the original LUBE method (O-LUBE) [13] and the modified LUBE method (M-LUBE) [16] using the same data set. Fig. 10 shows the PIs which are constructed based on the O-LUBE and M-LUBE. The PICPs and NMPIWs of the proposed method and two comparative methods are given in Table 3. According to Fig. 10 as well as Table 3, it can be seen that the NMPIW of the proposed method is smaller than that of the O-LUBE and M-LUBE, therefore the constructed PIs based on the proposed method have

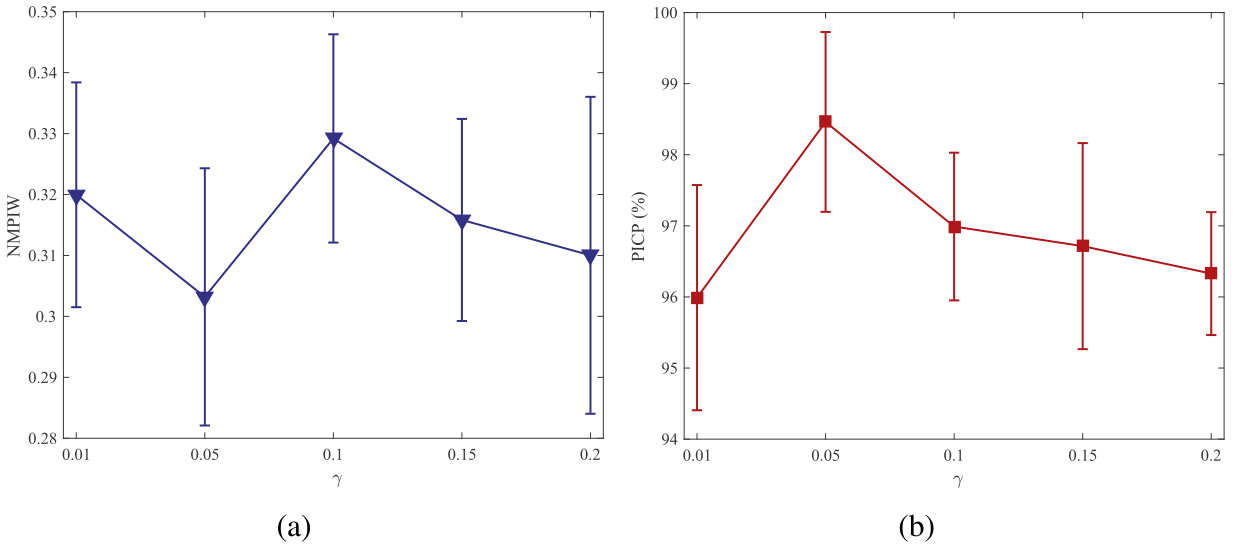

Fig. 11. Robustness analysis of results (NMPIW (a), PICP (b)) of the proposed PIs with different γ on the testing data.

a narrower shape. The PICP of the proposed method is larger than that of the two comparative methods because of the proposed objective function which the MAWD introduced.

4.4. Robustness analysis for parameters λ and γ

To examine the reliability of the proposed PIs, the robustness analysis for parameters λ and γ in the cost function F_{DSCN} are reported. As mentioned in Remark 1, the coefficient λ keeps fixed and is set to 1, only the coefficient γ is tuned to control the influences of CWC_M and $MAWD$ on the cost function F_{DSCN}. Fig. 11 exhibits the variations of mean values and standard deviations of the NMPIW and PICP of 30 experiments with respect to $\gamma = [0.01, 0.05, 0.1, 0.15, 0.2]$. The results indicate that the proposed PIs can keep sound robustness with respect to a certain range of γ.

5. Conclusion

This paper proposes a novel method of constructing PIs based on the LUBE method to quantify the uncertainty associated with the prediction of carbon residual content of crude oil. The DSCN is implemented in the LUBE method where the input weights and biases are generated randomly, therefore the number of parameters which need to be adjusted is reduced effectively. A new cost function for constructing PIs based on the LUBE method is proposed, which can improve the quality of the PIs and also provide a deterministic prediction value. In addition, a MBSA is proposed to optimize the output weights of DSCN for constructing PIs. Furthermore, experiments are carried out on a practical data set and the results indicate that the proposed method can construct PIs with a high coverage probability and narrow interval, and it can provide useful information for the decision-making process in crude oil refining.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61590922 and Grant 61525302, the Project of Ministry of Industry and Information Technology of China under Grant 20171122-6, the Fundamental Research Funds for the Central Universities under Grant N160801001 and Grant N161608001.

References

[1] R. Ak, V. Vitelli, E. Zio, An interval-valued neural network approach for uncertainty quantification in short-term wind speed prediction, IEEE Trans. Neural Netw. Learn. Syst. 26 (11) (2015) 2787–2800.
[2] M. Alhamdoosh, D. Wang, Fast decorrelated neural network ensembles with random weights, Inf. Sci. 264 (2014) 104–117.
[3] L. Bai, Y. Jiang, D. Huang, X. Liu, A novel scheduling strategy for crude oil blending, Chin. J. Chem. Eng. 18 (5) (2010) 777–786.
[4] L.L. Barbosa, F.V.C. Kock, R.C. Silva, J.C.C. Freitas, V.L. Junior, E.V.R. De Castro, Application of low-field NMR for the determination of physical properties of petroleum fractions, Energy Fuel 27 (2) (2013) 673–679.
[5] R. Cheng, Y. Jin, A social learning particle swarm optimization algorithm for scalable optimization, Inf. Sci. 291 (2015) 43–60.
[6] X. Chu, Y. Xu, S. Tian, J. Wang, W. Lu, Rapid identification and assay of crude oils based on moving-window correlation coefficient and near infrared spectral library, Chemometr. Intell. Lab. Syst. 107 (1) (2011) 44–49.
[7] P. Civicioglu, Backtracking search optimization algorithm for numerical optimization problems, Appl. Math. Comput. 219 (15) (2013) 8121–8144.

[8] T.I. Dearing, W.J. Thompson, C.E. Rechsteiner, B.J. Marquardt, Characterization of crude oil products using data fusion of process raman, infrared, and nuclear magnetic resonance (NMR) spectra, Appl. Spectrosc. 62 (2) (2011) 181–186.
[9] A. Ding, X. He, Backpropagation of pseudo-errors: neural networks that are adaptive to heterogeneous noise, IEEE Trans. Neural Netw. 14 (2) (2003) 253–262.
[10] M.A. Hosen, A. Khosravi, S. Nahavandi, D. Creighton, Improving the quality of prediction intervals through optimal aggregation, IEEE Trans. Ind. Electron. 62 (7) (2015) 4420–4429.
[11] A. Khosravi, S. Nahavandi, D. Creighton, Construction of optimal prediction intervals for load forecasting problems, IEEE Trans. Power Syst. 25 (3) (2010) 1496–1503.
[12] A. Khosravi, S. Nahavandi, D. Creighton, A.F. Atiya, Comprehensive review of neural network-based prediction intervals and new advances, IEEE Trans. Neural Netw. 22 (9) (2011) 1341–1356.
[13] A. Khosravi, S. Nahavandi, D. Creighton, A.F. Atiya, Lower upper bound estimation method for construction of neural network-based prediction intervals, IEEE Trans. Neural Netw. 22 (3) (2011) 337–346.
[14] A. Khosravi, S. Nahavandi, D. Srinivasan, R. Khosravi, Constructing optimal prediction intervals by using neural networks and bootstrap method, IEEE Trans. Neural Netw. Learn. Syst. 26 (8) (2015) 1810–1815.
[15] M. Li, D. Wang, Insights into randomized algorithms for neural networks: practical issues and common pitfalls, Inf. Sci. 382–383 (2017) 170–178.
[16] C. Lian, Z.G. Zeng, W. Yao, H.M. Tang, C.L.P. Chen, Landslide displacement prediction with uncertainty based on neural networks with random hidden weights, IEEE Trans. Neural Netw. Learn. Syst. 27 (12) (2016) 2683–2695.
[17] T. Lu, M. Viljanen, Prediction of indoor temperature and relative humidity using neural network models: model comparison, Neural Comput. Appl. 18 (4) (2009) 345–357.
[18] D.J.C. MacKay, The evidence framework applied to classification networks, Neural Comput. 4 (5) (1992) 720–736.
[19] A. Masili, S. Puligheddu, L. Sassu, P. Scano, A. Lai, Prediction of physical-chemical properties of crude oils by ^{1}H NMR analysis of neat samples and chemometrics, Magn. Reson. Chem. 50 (11) (2012) 729–738.
[20] E. Mazloumi, G. Ros, G. Currie, S. Moridpour, Prediction intervals to account for uncertainties in neural network predictions: methodology and application in bus travel time prediction, Eng. Appl. Artif. Intell. 24 (3) (2011) 534–542.
[21] D. Molina, U.N. Uribe, J. Murgich, Correlations between SARA fractions and physicochemical properties with ^{1}H NMR spectra of vacuum residues from colombian crude oils, Fuel 89 (1) (2010) 185–192.
[22] V.G. Morgan, L.L. Barbosa, V. Lacerda, E.V.R. de Castro, Evaluation of physicochemical properties of the post-salt crude oil for low-field NMR, Ind. Eng. Chem. Res. 53 (21) (2014) 8881–8889.
[23] P. Peinder, T. Visser, D. Petrauskas, F. Salvatori, F. Soulimani, B.M. Weckhuysen, Prediction of long-residue properties of potential blends from mathematically mixed infrared spectra of pure crude oils by partial least-squares regression models, Energy Fuel 23 (4) (2009) 2164–2168.
[24] H. Quan, D. Srinivasan, A. Khosravi, Particle swarm optimization for construction of neural network-based prediction intervals, Neurocomputing 127 (2014) 172–180.
[25] H. Quan, D. Srinivasan, A. Khosravi, Short-term load and wind power forecasting using neural network-based prediction intervals, IEEE Trans. Neural Netw. Learn. Syst. 25 (2) (2014) 303–315.
[26] H. Quan, D. Srinivasan, A. Khosravi, Uncertainty handling using neural network-based prediction intervals for electrical load forecasting, Energy 73 (2014) 916–925.
[27] S. Scardapane, D. Wang, Randomness in neural networks: an overview, Wiley Interdiscip. Rev. 7 (2) (2017) e1200, doi:10.1002/widm.1200.
[28] T.M. Shea, S. Gunsel, Modeling base oil properties using NMR spectroscopy and neural networks, Tribol. Trans. 46 (3) (2003) 296–302.
[29] D.L. Shrestha, D.P. Solomatine, Machine learning approaches for estimation of prediction interval for the model output, Neural Netw. 19 (2) (2006) 225–235.
[30] N.A. Shrivastava, A. Khosravi, B.K. Panigrahi, Prediction interval estimation of electricity prices using PSO-tuned support vector machines, IEEE Trans. Ind. Inf. 11 (2) (2015) 322–331.
[31] D. Wang, Editorial: randomized algorithms for training neural networks, Inf. Sci. 364–365 (2016) 126–128.
[32] D. Wang, C. Cui, Stochastic configuration networks ensemble with heterogeneous features for large-scale data analytics, Inf. Sci. 417 (2017) 55–71.
[33] D. Wang, M. Li, Deep stochastic configuration networks with universal approximation property, in: A Updated Version of this Paper has been Published in the Proceedings of 2018 International Joint Conference on Neural Networks, Rio de Janeiro, Brazil, 2017. ArXiv:1702.0563918 July 8–13,2018.
[34] D. Wang, M. Li, Robust stochastic configuration networks with kernel density estimation for uncertain data regression, Inf. Sci. 412–413 (2017) 210–222.
[35] D. Wang, M. Li, Stochastic configuration networks: fundamentals and algorithms, IEEE Trans. Cybern. 47 (10) (2017) 3466–3479.
[36] L. Zhao, B. Li, Y. Wang, B. Chen, W. Wang, Mill load parameter model using fast decorrelated neural network ensemble, Control Eng. China 24 (9) (2017) 1952–1957.
[37] N. Zheng, J. Ding, Regression GAN based prediction for physical properties of total hydrogen in crude oil, Acta Autom. Sinica 44 (5) (2018) 915–921.