Intelligent Partition of Operating Condition-Based Multi-Model Control in Flue Gas Desulfurization

XIAOLI LI\textsuperscript{1}, QUANBO LIU\textsuperscript{1}, KANG WANG\textsuperscript{1}, FUQIANG WANG\textsuperscript{2}, GUIMEI CUI\textsuperscript{3}, AND YANG LI\textsuperscript{4}

\textsuperscript{1}Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China
\textsuperscript{2}Shenhua Guohua (Beijing) Electric Power Research Institute Corporation, Beijing 100018, China
\textsuperscript{3}School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou 014010, China
\textsuperscript{4}School of International Studies, Communication University of China (CUC), Beijing 100024, China

Corresponding author: Quanbo Liu (liuquanbobjut@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61873006 and Grant 61673053, and in part by the National Key Research and Development Project under Grant 2018YFC1602704 and Grant 2018YFB1702704.

\textbf{ABSTRACT} Flue gas emission is an inevitable procedure in the course of electricity generation, which would pose a severe threat to human health, and has an adverse effect on our environment. Due to the fact that the environment in practical flue gas desulfurization system fluctuates frequently, system parameters tend to vary constantly during the operating process, thus control performance with traditional strategies tends to be suboptimal in most cases. To address this problem, some insight into operating conditions must be gained prior to taking proper control strategy. Therefore, in this article, based on actual measurements in 1000 MW Unit Wet Limestone FGD System for a coal-fired power plant, a kind of intelligent operating condition partition method is combined with the multi-model adaptive control strategy. Specifically, analysis and partition of operating condition is carried out in the first place, then adaptive multi-model control is implemented with the combination of parallel dynamic neural network and partition results. Additionally, the applicability of proposed control mode is investigated through different simulation examples. At the same time, to further enhance the flexibility of multi-model control structure, some possible improvements on it is also discussed.

\textbf{INDEX TERMS} Clustering, feature selection, flue gas desulfurization, multiple models, neurocontrol.

\section{I. INTRODUCTION}

In recent years, individuals in expanding numbers begin to focus their attention on the exacerbation of environment. The reason for the environmental deterioration is multifaceted, among which the high concentration of sulphur dioxide in flue gas plays a dominant role. In fact, coal-fired power stations are one of the primary sources for the flue gas emission [1], [2]. Therefore, more stringent regulations are implemented to impose restrictions on the \(SO_2\) emission for coal-fired power stations. As stated in the regulation implemented in 2014, the \(SO_2\) emission was limited to 35 mg/m\(^3\) [3]. As far as desulfurization technique is concerned, distinct methods such as dry, semi-dry and wet flue gas desulfurization can be utilized, among which limestone wet flue gas desulfurization (WFGD) is sufficiently mature and has become the most commonly used technology in practical scenarios [4]–[6]. The outlet \(SO_2\) concentration and pH value of slurry in absorber are two vital parameters, both of which can reflect the effectiveness of desulfurization in some sense. Accordingly, both of them need to be controlled to certain values such that a desirable desulfurization outcome can be guaranteed. Specifically, the pH value should be set to an optimal point, based on the trade-off between desulfurization efficiency and system stability. Meanwhile, the outlet \(SO_2\) concentration should be lower than \(SO_2\) emission limit by a certain margin, which could diminish the possibility of the surfeit of \(SO_2\) emission due to the instability in desulfurization system.

Over the past few years, considerable research was carried out on control problems in WFGD, and a wide variety of control strategies have been applied. A decentralized control
strategy on the basis of controllability analysis is applied to
the control of outlet SO$_2$ concentration in [7]. Li et al. in [8]
presented two kinds of control strategies, namely single-
loop feedforward-feedback control and cascade feedforward-
feedback control, for slurry pH value in the absorption tower.
The predictive control based upon dynamic matrix control
algorithm is employed to control the change of pH value in
[9]. Furthermore, neural network control has been widely
used in the context of WFGD and yields satisfactory control
performance [10], [11]. Despite great progress has been
made in works mentioned above, many issues remain to be
considered and addressed. In contrast to most publications
that largely deal with theoretical analysis on the control
method presented in WFGD, the emphasis in this work is
placed more on the integration of practical operating condi-
tions and control. Given the fact that the system works
in practical industrial context is always subject to the influ-
ence of varying environments, it is not possible for only a
global model to describe all types of operating conditions.
In such a case, it is preferable to build a set of local models and
make each model correspond to an operating condition
uniquely. Evidently, multi-model control is well-suited to be
applied in WFGD system, since different environments may
arise during the system operation. However, the validity of
multi-model control implemented in WFGD depends mainly
on two points: 1) exhaustive analysis on practical operating
conditions; 2) accurate identification of resulting sub oper-
ating conditions. The first issue is handled by us through a
sequence of machine learning techniques which will be made
clear in the following sections. Further, parallel dynamic
neural network (PDNN), which has a superior approxima-
cation capability compared with the static one [12], is chosen
such that more accurate identification result could be
yielded.

The rest of this article is organized as follows: in the next
section, the mechanism of WFGD and concept of operat-
ing condition based modelling are introduced briefly. The
detailed procedure for partition of operating conditions is
given in Section 3. Then in Section 4, identification model for
each operating condition is established firstly. Then control
system for WFGD is built via multiple models. Moreover,
in Section 5, a series of experiments are designed to com-
pare and improve the proposed control system. The paper is
concluded in Section 6.

II. PROCESS OF WET FLUE GAS DESULFURIZATION AND
MODELLING

A. MECHANISM AND PROCESS OF WET FLUE GAS
DESULFURIZATION

The principle of WFGD is the reaction between SO$_2$ in flue
gas and alkaline sorbent, limestone slurry in this work. The
overall chemical reaction is as follows [13]:

\[
\text{SO}_2 + \text{CaCO}_3(s) + \frac{1}{2}\text{O}_2 + 2\text{H}_2\text{O} = \text{CaSO}_4 \cdot 2\text{H}_2\text{O}(s) + \text{CO}_2
\]

(1)

Absorber is the primary equipment in WFGD (the one researched in our case study is presented in Fig. 2), for the whole desulfurization process takes place there. The flue gas generated by coal-fired boiler would first be passed into gas-gas heater after dedust, which aims at dropping tempera-
ture in flue gas. Then flue gas enters into the absorber
through the inlet duct and flows from the bottom up. Mean-
while, the limestone slurry is sprayed downwards from a set
of fixed nozzles and would meet the rising flue gas. The
countercurrent contact between SO$_2$ in flue gas and slurry can
increase the contact area of the two and hence improve SO$_2$
absorption efficiency. Additionally, oxygen is sparged to the
slurry tank in absorber, in order to stimulate the oxidation of
calcium sulphite and bisulphite. Moreover, the final product
gypsum is obtained after the process of sludge removal and
dehydration [14], which can be recycled for construction and
agriculture uses. A simplified representation of WFGD
system is shown in the figure below.

As depicted in FIGURE 1, SO$_2$ concentration in desul-
furized flue gas and pH value in slurry tank are measured
by SO$_2$ concentration detector and pH meter respectively,
both of which are key parameters in a WFGD system. To be
specific, pH value in slurry tank closely correlates with the
desulfurization efficiency [15] and outlet SO$_2$ concentration
can best reflect the effectiveness of desulfurization. Thus,
we select two of them as controlled variables in our work.
Meanwhile, the densimeter and flowmeter are installed on the
ducts, through which limestone slurry is fed to the absorber.
Thus slurry flow flux together with slurry density can be mea-
sured simultaneously. Moreover, the flow flux of slurry can be
adjusted via slurry pump, while slurry density is determined
by the feed quantity of limestone. Consequently, the flux and
density of limestone slurry are selected as manipulated inputs
in our work.

B. THE PRINCIPLE OF OPERATING CONDITION BASED
MODELLING

In the domain of engineering, system modelling can be basic-
ally classified as mechanical and black-box approaches [16].
The former is based solely on thorough understanding of
inherent mechanism of system, while black-box model is
derived from observed data of the system. Considering
chemistry takes place in the process of flue gas desulphurization is rather complicated, black-box modelling is chosen in our case study. As the measured process data play a dominant role in modelling method of this sort, the accuracy and efficacy of modelling will greatly depend on analysis and mining of the data. However, in most cases, a model only works well for a particular range. That is, the model we built may not valid in the whole operation and it is merely effective locally in a certain environment. In this sense, an attempt is made to construct valid models for multiple local environments, instead of using a global model to characterize the whole process. The local environment for which each model represents is termed as operating condition in our case study. Each operating condition may consist of several segments of measured process data. Mathematically, it could be viewed as a set with respect to operating points that belong to the same pattern. Suppose set $X = \{x(T), x(2T), \ldots, x(nT)\}$ is a time sequence with $n$ operating points, where $x \in \mathbb{R}^d$ and $T$ is the sampling time. Let $S$ be a particular similarity measure function, which quantifies the similarity between two operating points. Based on $S$, the overall operating condition $C$ is partitioned as $N$ different sub conditions $\{C_1, \ldots, C_N\}$ ($N \leq n$), such that

1) $C_i \neq \emptyset$, $i = 1, \ldots, N$;
2) $C = \bigcup_{i=1}^{N} C_i$;
3) $C_i \cap C_j = \emptyset$, $i, j = 1, \ldots, N$ and $i \neq j$.

Note that any operating point in $X$ is a $d$-variate vector, but $d$ may not refer to the total number of variables in dataset. Selected $d$ variables should be able to fully describe the behavior of system. To better illustrate the relation between operating conditions and operating points, we use the following figure to display operating conditions partitioned by three-dimensional operating points, where points with the same color stand for one type of operating condition.

The operating conditions essentially reflect varied operation modes in system. Once operating conditions are determined explicitly, local models can be established accordingly. Referring to the notion of operating condition stated earlier, the procedure of operating condition based modelling can be summarized as:

1) Determine dimensionality of operating space constituted by operating point vectors.
2) Utilizing mapping $F$ such that the operating space can be partitioned into several non-overlapping regions.\[3) Establishing model for each region (operating condition)\]

On this base, the specific process of operating condition partition (i.e., the above former two steps) in our case study will be elaborated in the next section. And the modelling part is discussed in Section 4.

III. PARTITION OF OPERATING CONDITIONS IN WFGD

In this section we will give details on how we perform operating condition partition in WFGD system. Prior to working on it, it is of necessity to determine the dimension of operating space, i.e., variables that are able to describe the operating condition of system. Typically, prior knowledge as well as practical engineering experience plays a significant role in
this process. In our case study, the measured process data is obtained from 1000 MW unit wet limestone FGD system for a coal-fired power plant, which spans over nearly six hours and sampled every one second. Since the ultimate goal in our case study is to control slurry pH value and outlet SO$_2$ concentration, thus the initial choice of variables affecting operating conditions should be based upon above two controlled variables. Empirically, the operating condition in our case study is primarily influenced by the following four variables: 1) load for power generating units; 2) SO$_2$ concentration in raw flue gas; 3) O$_2$ concentration in raw flue gas; 4) temperature of raw flue gas and 5) fly ash concentration. In what follows, to avoid curse of dimensionality [17] in operating space, we will select ones that can better reflect operating conditions among five above variables. In the following, random forest regression algorithm would be performed to select proper features from above variables.

As an ensemble learning algorithm, random forest can be used for both classification and regression [18]. The rationale for random forest is to combine results of a number of decision trees for classification or regression purpose. For classification issues, it uses majority voting mechanism to determine the output category. While for regression ones, output value is calculated as the average for outputs of total decision trees. Additionally, it is characterized by strong robustness, that is, outliers and surplus data have minor impact on the final results [19]. The main steps of random forest regression are shown in the form of flow diagram as below (where sum squared error is represented as SSE):

Here we treat five aforementioned variables (i.e. power unit load, SO$_2$ concentration in flue gas, O$_2$ concentration in flue gas, flue gas temperature and fly ash concentration) as input features, and pH value of slurry and outlet SO$_2$ concentration as outputs. The process of feature selection is built upon random forest regression, where bootstrapped samples are also utilized to construct each regression tree. Here, mean decrease accuracy (a measure that will be defined later on) is exploited to quantify the importance of five selected features in the WFGD system. In this method, the sequence of values in a specified feature is reordered randomly each time, while that of remaining features keep unchanged. Generally speaking, for a relatively important feature, permutation of its values would decrease the correlation between this feature and model outputs, thus lead to a major impact on accuracy of random forest regression model. For this reason, mean decrease accuracy for prior features would be greater than those of less important ones. In addition, coefficient of determination $R^2$ is chosen to measure the accuracy of model regression. Then, the mean decrease accuracy of $i$th feature in random forest is given by

$$R^2_{\text{avg}_i} = \frac{1}{N} \sum_{n=1}^{N} R^2_{\text{avg}_i}(T_n)$$

(2)

where $N$ denotes the number of trees in random forest. $R^2_{\text{avg}_i}(T_n)$ represents relative variation of $R^2$ value in the case of values permutation performed on $i$th feature for regression tree $T_n$, which can be calculated as:

$$R^2_{\text{avg}_i}(T_n) = \frac{R^2(T_n) - R^2_i(T_n)}{R^2(T_n)}$$

(3)

with $R^2(T_n)$ and $R^2_i(T_n)$ in (3) representing two $R^2$ values calculated via $n$th tree, which are obtained before and after the permutation of values in $i$th feature respectively. Defining $\hat{y}_i$ and $y_i$ as the estimate and real value for $i$th sample, $\bar{y}$ is the mean value of $m$ outputs, the $R^2$ has the form

$$R^2 = \frac{\sum_{i=1}^{m} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{m} (y_i - \bar{y})^2}$$

(4)

By (2)-(4), mean decrease accuracy of five variables, which quantifies to what degree each variable affects the overall operating condition in WFGD system, can be calculated respectively via on-site data described earlier. The results with $N = 100$ are presented as below. For simplicity, features are labelled by parts of their names.

It follows from FIGURE 5 that mean decrease accuracy values of SO$_2$ and O$_2$ concentration in raw flue gas, plus power generating units load far exceed those of the other two features’, which implies that they have greater impacts on operating conditions in the context of WFGD. As a result, we would select them as variables to perform the task of operating conditions partition. As for variables chosen above, hierarchical clustering method is applied to determine the number and components of operating conditions. Hierarchical clustering can be classified into two categories, i.e., agglomerative and divisive types, and the former is more widely used [20]. Furthermore, hierarchical clustering is characterized by little dependence on input parameters, no restrictions of clustering shapes and convenience for visualization, which makes it applicable in various practical scenarios [21]. The agglomerative hierarchical clustering is a kind of bottom-up clustering method. Let us define the dataset $D = \{x_1,x_2,...,x_N\}$, and cluster set $C = \{C_1,C_2,...,C_K\}$, where $K$ is the specified number of cluster. The mechanism of clustering can be summarized as follows:

1) Regard each sample as a separate cluster, i.e., $x_i = C_i$, $i = \{1, \cdots, N\}$. Then calculate the similarity between each pair of cluster. The similarity function below is given to measure the similarity between clusters $C_i$ and $C_j$.

$$s(C_i, C_j) = \frac{\sum_{x \in C_i, y \in C_j, i \neq j} d(x, y)}{n_in_j}$$

(5)

2) The pair with the highest degree of similarity would merge into a new cluster. As a result, the number of clusters in the set would decrease by one.

3) Repeat the above two steps, (i.e., for new cluster set, calculate the similarity between two clusters in turn, and combine the pair with the greatest similarity each time) until the prescribed cluster number $K$ is reached.
Unlike clustering conducted on some specified datasets, ground true labels are not existent in practical industrial scenarios, which brings difficulties to the evaluation of clustering results and therefore the determination of operating conditions in our case study. In response to this problem, we select Silhouette index (Si) and Calinski-Harabasz Index (CH) to measure clustering performance such that the optimal number of operating conditions is obtained. As both of measures have been verified to be one of the most accurate clustering validation methods in [22], it is believed they can provide effective information on goodness of clustering results. Due to the space limitation, above two indices are described briefly. Silhouette index is an indicator which utilizes intra-class distance (cohesion) as well as inter-class distance (separation) to evaluate the performance of clustering. The detailed procedure for calculation and introduction can be found in [23] and [24]. To evaluate the validity of a clustering result, we use average value of all samples’ Si as the final score for tested clustering. The range of clustering Si varies between −1 and 1, and the closer it is to 1, the more appropriate the clustering is. On the other hand, CH is able to measure the dispersion degree for clusters, and is defined as the ratio between inter-cluster variance and intra-cluster variance [25]. Moreover, the CH is higher in the case of well-separated and dense clustering. For the purpose of finding the optimal partition number of operating conditions under hierarchical clustering method, the number of clustering ranging from 2 to 12 is tested. And the corresponding Si and CH for each case is listed in the table below.

| Number | Silhouette index | Calinski-Harabasz Index |
|--------|------------------|-------------------------|
| 2      | 0.547            | 2025.706                |
| 3      | 0.578            | 2446.220                |
| 4      | 0.622            | 2940.341                |
| 5      | 0.685            | 3329.327                |
| 6      | 0.637            | 3044.369                |
| 7      | 0.624            | 2981.523                |
| 8      | 0.629            | 3001.615                |
| 9      | 0.598            | 2733.238                |
| 10     | 0.563            | 2251.306                |
| 11     | 0.531            | 1926.167                |
| 12     | 0.517            | 1703.021                |

It follows from table above that both Si and CH can achieve its greatest value when clustering number is set as 5. As a result, the overall operating condition will be partitioned into five parts in our case study. To ensure the accuracy of clustering, z-score standardization is applied to data set of three selected features before clustering. And the partition result with the use of hierarchical clustering is shown as follows.

As seen from FIGURE 6, operating points with a certain kind of color represent a class of operating condition in our system. Furthermore, the proportion of each operating condition arising in the whole process can also be shown in an intuitive way. By far, the task for operating space partition has been accomplished. And partitioned operating conditions would serve the control purpose in next section.

**IV. PARALLEL DYNAMIC NERUAL NETWORK-BASED MULTI-MODEL CONTROL**

In this section, the multi-model control on the basis of PDNN is introduced. Following the work done in previous section, this section involves two aspects, i.e., 1) construction of local models for operating conditions partitioned before 2) combination of those local models for control purpose with the use of PDNN. Neural network can be classified as static and dynamic net based upon whether feedback is contained in its configuration. As feedback is incorporated in the structure, the dynamic neural network has superior approximation capability and can still be used in the presence of model uncertainty. PDNN, which is of interest in this work, is one type of dynamic neural networks, and the other type is known as series-parallel neural network, which was discussed by the authors in another paper [26].

As pointed out in the introductory section, there exists strong nonlinearity in the whole WFGD process. Meanwhile, varying industrial environment may bring about abrupt variations and uncertainties in system dynamics. To achieve desirable control performance in such situation, the strategy of divide and conquer is taken by us. To be specific, multiple local models rather than a global one is utilized to characterize the overall process. In this way, the complicated process is decomposed into several sub processes, which simplifies both identification process and control design. Then
corresponding controller is designed for each local model. Further, the original black-box model becomes well interpreted when established by operating points in each operating condition, which is of great significance in engineering practice. Additionally, the major benefit of this control mode lies in its powerful flexibility. For instance, there is no need to design local models with identical type and structure, for the choice of local models and control strategies are critically dependent on their corresponding operating conditions. The flexibility property of this kind of control mode is definitely more appealing to practical industrial scenarios. In what follows, partitioned operating conditions would be modelled by PDNN first, after that the principle of controller design for setpoint tracking problem in WFGD is formulated.

A. MODEL IDENTIFICATION

Local models are constructed to approximate system behavior in different operating environments. Model accuracy is the determinant for control performance in a specified operating range. The model to be identified is double input-double output, where the model inputs are flow and flux density of limestone slurry, and outlet SO₂ concentration along with pH value in slurry serves as output variables. According to operating points in partitioned operating conditions (as shown in FIGURE 6), the input and output variables for each operating condition can be easily obtained. Let a double input-double output continuous-time nonlinear system represent any one of the processes:

\[ \dot{x}_t = f(x_t, u_t, t) \]  

where \( x_t \) and \( u_t \) are the system state and input vectors, respectively, \( t \) is the system operating time. The following parallel dynamic neural network in [27] is used to identify the system above:

\[ \hat{x}_t = A\hat{x}_t + W_{1,t}\sigma(V_{1,t}\hat{x}_t) + W_{2,t}\phi(V_{2,t}\hat{x}_t)u_t \]  

To simplify problem analysis, let 2 × 2 matrix \( V_{1,t} \) and \( V_{2,t} \) and \( \phi(\cdot) \) be an identity matrix, which implies that there is no hidden layer in the neural network. Then (7) becomes

\[ \dot{\hat{x}}_t = A\hat{x}_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}\phi(\hat{x}_t)u_t \]  

where in our case study, \( A \in \mathbb{R}^{2 \times 2} \) is a Hurwitz matrix. \( A \) in (8) is the output of neural network. \( W_{1,t} \in \mathbb{R}^{2 \times 2} \) and \( W_{2,t} \in \mathbb{R}^{2 \times 2} \) are weight matrices connecting input and output layer. \( \phi(\cdot) \) is a differentiable input function, and it is defined as \( \phi(u_t) = u_t \), \( \sigma(\cdot) \) is a two-dimensional activation function whose major function is nonlinear states feedback. Either sigmoid or hyperbolic tangent function can be chosen due to their approximate identification performance in our case study. Here, sigmoid function is used and has the form:

\[ \sigma_i(x) = \frac{a_i}{1 + e^{-b_i x}} + c_i \quad i = 1, 2 \]  

where \( a_i, b_i \) determines bound and slope of the sigmoid function, \( c_i \) represents the quantity of vertical shift. Taking unmodelled dynamics into consideration, that is, the neural network (7) fails to describe nonlinear system (6) exactly, there exists an identification error \( \Delta_t \) which is denoted as:

\[ \Delta_t = \hat{x}_t - x_t \]  

It is assumed that nonlinear system (6) satisfies all assumptions we proposed in [12], in this case the weight updating law can be formulated as (11) and (12).

\[ \dot{W}_{1,t} = -s_tK_1P\Delta_t\gamma^T(\hat{x}_t) \]  

\[ \dot{W}_{2,t} = -s_tK_2P\Delta_t\gamma^T(u_t) \]  

Here \( K_1 \) and \( K_2 \) are positive matrices, \( P \) denotes the solution to matrix Riccati equation shown as below:

\[ A^TP + PA + PRP + Q = 0 \]  

\( s_t \) is defined as:

\[ s_t : = [1 - \sqrt{\frac{P}{P_1/2}}]_+, [z]_+ = \begin{cases} z & 0 \leq z < 0 \\ z & 0 \leq z < 0 \end{cases} \]  

with

\[ \mu : = \frac{\tilde{\eta}}{\lambda_{\min}(P^{-1/2}Q_0P^{-1/2})} \]  

Provided that (11)-(15) is utilized to carry out the identification task for each operating condition, then conclusions in Theorem 1 can be drawn accordingly.

**Theorem 1:**

1. The processes \( \Delta_t, W_{1,t} \) and \( W_{2,t} \) are bounded, i.e., \( \{\Delta_t, W_{1,t}, W_{2,t}\} \subset L_{\infty} \).

2. The identification error \( \Delta_t \) satisfies the following performance, i.e., \( \lim_{t \rightarrow \infty} \sup \|P^{1/2}\Delta_t\| = \sqrt{\mu} \).

The detailed proof can be found in our previous work [12].

In the following, due to the space limitation, we only detail the identification process of one operating condition as an example. The following parameters are chosen in this case, and it is a trial-and-error process.

\[ A = \begin{pmatrix} -15 & 0 \\ 0 & -10 \end{pmatrix}, R = \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix}, Q = \begin{pmatrix} 20 & 0 \\ 0 & 10 \end{pmatrix}, Q_0 = I, \tilde{\eta} = 0.2, K_1 = K_2 = \begin{pmatrix} 30 & 0 \\ 0 & 30 \end{pmatrix} \]

Then solve the matrix Riccati equation in (13) for \( P = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \). Sigmoid function in (9) takes the form of \( a = 1.5, b = 1, c = -0.5 \). All elements in initial weight matrices \( W_{1,0} \) and \( W_{2,0} \) are specified as zeros. To speed up the training process of neural network, process data are scaled between \([-1,1]\), and initial states are set as \( \hat{x}_0 = (0, 0)^T \). Identification results for the chosen operating condition are presented as below.

And it is seen that identified states yielded by established model are in good agreement with actual ones, which implies the local model is able to characterize the specified operating condition accurately. Similarly, remaining four operating conditions can be identified by adjusting some parameters above if necessary, and the objective is to obtain the optimal
identification results. The table below summarizes the mean squared error in different cases, where 1st identification results. The table below summarizes the mean squared error for different local models.

**TABLE 2.** Mean squared error for different local models.

| Model number | 1st state mse | 2nd state mse |
|--------------|--------------|---------------|
| 1            | 3.2512e-05   | 6.3859e-06    |
| 2            | 5.0875e-03   | 5.8105e-05    |
| 3            | 1.2270e-03   | 1.1897e-05    |
| 4            | 9.6505e-04   | 3.5623e-06    |
| 5            | 3.6372e-04   | 6.7904e-06    |

**FIGURE 7.** Identification result for one of operating conditions.

The dynamics of tracking error can be derived by differentiating $\Delta_t^*$ with respect to time, it can be derived from subtracting the derivative of constant reference signal $x_t^*$ from the expression for $\dot{x}_t$ in (8):

$$\dot{\Delta}_t^* = \dot{x}_t^* - x_t^* = A\dot{x}_t + W_{1,t}\sigma(\dot{x}_t) + W_{2,t}u_t$$  \hspace{1cm} (17)

The control input $u_t$ takes the form:

$$u_t = -W_{2,t}^{-1} [A\dot{x}_t^* + W_{1,t}\sigma(\dot{x}_t)]$$  \hspace{1cm} (18)

Then, substitution of $u_t$ into (17) yields

$$\dot{\Delta}_t^* = A(\dot{x}_t - x_t^*) = A\Delta_t^*$$  \hspace{1cm} (19)

As matrix $A$ in (19) is stable, the error is asymptotically stable and will decay to zero over time. Further, for the purpose of ensuring the existence of $W_{2,t}^{-1}$ in control input, projection algorithm in [29] can be applied to delimit the scope of each entry in $W_{2,t}$.

In this way, the controller of each local model can be designed accordingly. Considering it is not possible to model all operating conditions in WFGD system, adaptive control should also be contained to guarantee the control accuracy of unknown environment. In this sense, switching and tuning control strategy which has been researched broadly in many works (e.g., [30], [31]) is applied in our case study. Explicitly, local models we built before are used for control purposes in anticipated operating conditions, and adaptive one plays the role of compensating for model uncertainties in local models and controlling in unknown environments. Let $n$ be the number of models we utilize, then $\{I_i, C_i\}_{i=1}^n$ represents $n$ sets of model/controller pairs. As for fixed models, namely local models in our work, parameters in both models and corresponding controllers would keep constant all the time. Rather, parameters in adaptive pairs vary with different operating conditions such that control error can be further eliminated. In theory, high-performance control can be achieved only if appropriate local models and adaptive models are selected. At this stage, the key problem we are facing with is how to combine those local models together. Here, it is assumed that only one controller is acted on the plant at each instant, and the determination of controller is dependent upon identification performance of different models. Based on the identification error, the following criterion is established to measure the performance of each model at every instant.

$$J_i(t) = \alpha \|e_i(t)\|^2 + \beta \int_0^t \|e_i(\tau)\|^2 d\tau$$  \hspace{1cm} (20)

in which the first term represents transient error at an instant $t$, while the other term is the accumulated error of ith model. Provided that the performance criterion only contains the first part, in this case, the switching between models would become frequent, which may decrease the stability of control system. Therefore, the second integral term is added. And the incorporation of such two types of error is capable of evaluating the identification performance of certain model in a more comprehensive way. At each instant, the model...
indexed with arg min_{i∈γ} J_i(t) can best describe the current operating condition and thus should be selected. In our case study, the two weighting coefficients in (20) are determined to be \( α = 0.8, β = 0.2 \) by trial and error. Further, five local models established in the preceding subsection, plus one adaptive model constitute the multi-model control system in the context of WFGD. The control system has the structure depicted in FIGURE 8 as below.

In what follows, the stability analysis of control input and actual system state is discussed and proved in sequence.

**Theorem 2:**

Based on model control (18), Index function (20) and switching mechanism shown in FIGURE 8, a multiple model adaptive controller (MMAC) can be given. By taking this kind of MMAC, it can be proved

1. The control law \( u_t \) is bounded.
2. The system state \( x_t \) is bounded and the tracking error \( x_t - x^*_t \) would also be located in a given scope.

The proof of **Theorem 2** is given below.

**Proof of Theorem 2:**

1. From **Theorem 1**, we know that both \( W_{1,t} \) and \( W_{2,t} \) are bounded. Besides, setpoint \( x^*_t \) and sigmoid function value \( σ(\hat{x}_t) \) are also bounded. From (18), it can be concluded that the derived control input \( u_t \) is also bounded. (Here the case that matrix \( W_{2,t} \) is singular is not considered, the case can be avoided by projection algorithm or other modified method.)

2. From (16) (17) (18) (19), it can be seen that the tracking error \( Δ_t^* \) can satisfy the equation (19), if stable \( A \) is selected, the error would decay to zero, i.e., \( \lim_{t→∞} (\hat{x}_t - x^*_t) = 0 \). From (10) \( Δ_t = \hat{x}_t - x_t \), we have

\[
\lim_{t→∞} (x_t + Δ_t) - x^*_t = 0 \tag{21}
\]

From (21)

\[
\lim_{t→∞} (x_t - x^*_t) = \lim_{t→∞} -Δ_t \tag{22}
\]

then,

\[
\lim_{t→∞} \|x_t - x^*_t\| = \lim_{t→∞} -\|Δ_t\| = \|Δ_t^*\| \tag{23}
\]

From **Theorem 1** we have

\[
\lim_{t→∞} \sup_{t>0} \|p^{1/2}Δ_t\| = \sqrt{μ} \tag{24}
\]

From (23) (24), considering \( μ \) is a bounded positive number, we have \( \|x_t - x^*_t\| \) would be located in a bounded scope, \( x_t \) would also be bounded due to the bounded \( x^*_t \).

**Remark:**

From Theorem 2, the stability of the system can be guaranteed. Due to the existence of multiple models and switching mechanism, the transient response and control performance will be improved greatly.

**V. EXPERIMENTAL STUDIES**

In this section, based on the control framework we built in last section, different experiments will be designed and compared with each other, in order to find the one that best suited to the control problem in our case study. To test the effectiveness of each control method, a testing environment is set up on the basis of a large amount of observations collected in the same timescale of different days. By monitoring the actual plant’s working condition and variation tendency of online data, plus the engineering experience of operators, the data of five benchmark operating conditions are extracted respectively. Then parallel dynamic neural network is utilized to model each condition. In this way, the testing environment is established. Though there may exist some deviation between actual environments and testing ones however, the testing environment can provide a relative accurate platform to test the control performance of different control methods. Here, PDNN nominal model for the first testing operating condition is:

\[
\hat{x}_t = Ax_t + W_{1,t}\sigma(\hat{x}_t) + W_{2,t}γ(u_t)
\]

\[
A = \begin{pmatrix} -20 & 0 \\ 0 & -10 \end{pmatrix}, W_1 = \begin{pmatrix} -25.209 & -12.246 \\ -26.446 & -54.049 \end{pmatrix}, W_2 = \begin{pmatrix} 65.534 & 8.087 \\ -2.885 & 4.353 \end{pmatrix}
\]

The other four conditions are identified by PDNN with high accuracy in much the same way described in last section by tuning parameters. To simplify the illustration, five benchmark models are applied sequentially every 30 seconds. As for adaptive model, unless otherwise noted, we designate parameters of the local model, which appears with the highest probability in the whole process (FIGURE 6 provides reference for it), as initial weights of adaptive model used below. Moreover, weights of the adaptive model would revert to initial ones when switching takes place in benchmark models. Next, the multi-model based control methods are developed to address the setpoint tracking problem in WFGD system.
FIGURE 9. Performance of single model adaptive control: (a) System response (b) Amplitude of control error.

FIGURE 10. Performance of Experience based multi-model control: (a) System response (b) Amplitude of control error (c) Switching sequence.
A. SINGLE ADAPTIVE MODEL ONLY

The rationale for adaptive control is that parameters of model vary with plant’s parameters, and the control error is eliminated gradually in this course. The drawback of this control mode is evident, that is, when wide variations appear in plant’s parameters, the transient response of system becomes rather poor. Two subplots in FIGURE 9 display the system response and control error amplitude under the control of single adaptive model. Though the transient error is not apparent, there exists large overshoot when parameters of plant change abruptly, particularly at 90 second, which indicates that single adaptive control is not an ideal control mode to the setpoint tracking problem in our case study.

B. EXPERIENCE BASED MULTI-MODEL CONTROL

As for WFGD system in coal-fired power plant, load for power generating units varies constantly to satisfy the demand of power grid. Thus in engineering practice, it is commonly used as an important indicator to determine the operating condition. Here, we first determine load range by calculating the difference between maxima and minima. Then the load range is equally split into five parts, and the set of partition thresholds is \{582.59, 674.24, 767.92, 859.04\}. In this way, five local models are built by PDNN with input-output data in each partition interval. The multi-model control system can be constructed by reference to FIGURE 8, and the initial weights of adaptive model, which would be reinitialize every thirty second, are also taken as parameters of one of local models that trained with the highest number of data. As shown in FIGURE 10, in contrast to the conventional single model adaptive control, the transient error is reduced significantly in this control mode. The switching sequence in FIGURE 10(c) (adaptive model is numbered six) reveals that local models can exert alternate control under different environments. However, the effect of local models in this case is still limited, e.g. both 3rd and 4th model are barely utilized during the simulation. Meanwhile proximity between models may also cause overshoot in response due to model switching, when there is actually no change taking place in environment. Overall, based on obtained results, there still exists scope for development of local models.
C. INTELLIGENT-BASED PDNN MULTI-MODEL CONTROL

In this case, local models are established by partitioned operating conditions detailed above, and the structure of control system remains the same. The resulting system response, error amplitude and switching sequence are presented in FIGURE 11. In comparison with the former two control modes, it can be clearly seen that the control performance is improved markedly, especially in transient response. The switching sequence indicates that each local model serves to one type of operating condition. The closeness between the local model and environment enables the transient error to reduce rapidly. Meanwhile, adaptive model works for eliminating the steady-state error by parameter adjustment on the basis of initial values. The fundamental difference between control mode in Part B and Part C lies in the choice of local models, or equivalently, the technique to partition operating conditions. By comparison, the partition method proposed in this work outperforms the one commonly used in engineering practice.

Comment:

The operating condition based control framework taken in this work is just based on the desulfurization case we studied, however, it can be extended to a large number of industrial process control. The major advantage of this framework is its flexibility. For instance, black-box model and grey-box model can coexist in multi-model control structure. Further, other intelligent algorithms are also available to extract information of process data. In fact, there may not exist single best structure in this control framework. We need only consider the best possible realization in light of practical condition and demand in process industries.

In what follows, the possible improvements of the control mode in Part C will be discussed.

D. INTEGRAL FEEDBACK LQR AND PDNN BASED MULTI-MODEL CONTROL

The overall desulfurization process is typically of high nonlinearity and complicity. But once the process is split into several parts, the local process may become easy to deal with. In such a case, a linear model is sufficient to describe this sub process, and a wide range of control strategies in modern control can therefore be utilized. Among operating conditions partitioned earlier, the first operating condition, which is a relatively simple one, is chosen to be identified with a linear model. Here, this process is identified with a state-space model, and subspace-based N4SID method in [32] is adopted in the course of modelling. As the model is used for control purpose ultimately, thus the trade-off between model accuracy and control simplicity must be made. By comparing models with different orders in terms of two above indicators, the second-order system is given by

\[
\begin{align*}
\dot{x}_r &= Ax_r + Bu_t \\
y &= Cx_r
\end{align*}
\]

(25)

with \( A = \begin{pmatrix} -0.876 & 0.958 \\ 0.187 & -2.755 \end{pmatrix}, B = \begin{pmatrix} 0.688 \\ -1.688 \end{pmatrix}, C = I \), and it can be verified that pair \((A, B)\) is controllable. FIGURE 12 shows the comparison between identified and measured outputs.

It is seen that the state space model can provide desirable approximation of the prescribed process. In what follows, LQR with integral feedback would be designed to address the setpoint tracking problem under the operating condition of interest. As for this type of control, integral of control error is also taken as a state variable (hereafter known as integral state \(x_I\)) and fed back to open-loop system. Suppose that \(r_t\) and \(y_t\) are reference signal and system output respectively, then the control error is defined as \(e = r_t - y_t = r_t - Cx_r\).

By definition, \(x_I = e = r_t - Cx_r\) and we can rewrite system dynamics (21) in terms of augmented matrices as

\[
\begin{bmatrix} \dot{x}_r \\ \dot{x}_I \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x_r \\ x_I \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u \quad y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x_r \\ x_I \end{bmatrix}
\]

(26)

To simplify notations, the augmented state matrix and input matrix in (22) are denoted as \(A_{aug}\) and \(B_{aug}\). The control law is calculated by

\[
u_t = -K^*x_{aug}.x_{aug} = [x_I \ x_I]^T
\]

(27)

where \(x_{aug}\) is the state vector with the inclusion of integral state \(x_I\), and \(K\) is the state feedback matrix which can be obtained by minimizing the following performance index

\[
J = \int_0^t x_{aug}^T Q x_{aug} + u^TR u \quad t > 0
\]

(28)

with \(Q\) and \(R\) are respectively being the state and control weighting matrix, which should be chosen to make trade-off between control performance and control effort. From [33], \(K\) is calculated as

\[
K = R^{-1}B^T P
\]

(29)
and $P$ is solved by algebraic Riccati equation as shown below

$$A^T P + PA - PB_R^{-1}B^T P + Q = 0 \quad (30)$$

In our simulation studies, we select weighting matrices $Q$ and $R$ as $Q = \text{diag}(250,250,30,30)$, $R = \text{diag}(0.1,0.1)$. First of all, the algebraic Riccati equation (26) can be solved by referring to [34], [35] with given $A_{aug}$, $B_{aug}$, $Q$ and $R$. Then substitute the solution $P$ into (25) yields the following feedback gain $K$.

$$K = \begin{bmatrix} 9.364 & 49.115 & -5.083 & -17.926 \\ -49.115 & 9.364 & 19.403 & -2.086 \end{bmatrix} \quad (31)$$

Then LQR design has been realized at this stage. After that, for multi-model control system built in Part C, we replace corresponding PDNN model/controller pair with state space model and controller established in this part, while other parameters and settings in system remain unchanged. The system response and variation of control error are presented in former two subplots in FIGURE. 13. As for control mode in Part C and Part D, the major difference lies in the control of the first operating condition, which determines initial performance of the system. By comparing the results in FIGURE. 11 and FIGURE. 13, it can be found that both of two control modes can achieve desired control performance under different operating conditions. Further, it takes a bit less time for PDNN-based control mode in Part C to reach desired reference inputs in the first operating condition. FIGURE. 13(c) displays initial control inputs in the case of two control modes (for comparison, the results are shown in a truncated timeframe). It turns out that the control input is reduced significantly by introducing optimal control into the system. Evidently, the latter control mode can achieve better trade-off between control effort and tracking performance. This case suggests that, in practical industrial scenarios, it is...
feasible to implement modern control methods, e.g. optimal control in this case, for process with relatively low complexity. The optimal control introduced facilitates the adjustment of parameters such that required control performance can be easily obtained, meanwhile the introduction of varied control strategies can further improve the control flexibility for the whole system.

E. PDNN BASED GLOBAL CONTROL

Control modes in previous cases focus on choosing the best model for a certain environment, namely only one model is available at any instant. In this case, the control task is performed in a way that integrate all models together and each model becomes a participant in the control process. The rationale for this control mode is that different weights are assigned to local controllers based on identification performance of corresponding models. To measure the validity of ith model at instant t, the following smooth Gaussian function is introduced

\[
g(J_{it}) = e^{-\frac{J_{it}^2}{2c^2}} c > 0, \quad i = 1, \cdots n
\]  

(32)

in which \(c\) represents the width of Gaussian function, \(J_{it}\) is the performance index in (20) at instant \(t\) and \(n\) denotes the number of models. It is known that function \(g\) can map each \(J_{it} \geq 0\) into interval \((0,1]\). Note that the assigned weight of each model is actually a relative weight, which measures the comparative identification accuracy of each model. The relative weight of ith model at instant \(t\) is defined in terms of \(g(J_{it})\) as

\[
w_{it} = \frac{g(J_{it})}{\sum_{i=1}^{n} g(J_{it})}
\]  

(33)

Let \(\{u_{it}\}_{i=1}^{n}\) be control inputs of different models in system at instant \(t\). Then we can derive the global control law by connecting each model’s control input vector with the use of
corresponding relative weight.

\[ u(t) = \sum_{i=1}^{n} w_i u_i(t) \]  

(34)

Prior to simulation studies, a major issue needs to be concerned is the selection of width \( c \) in Gaussian function. From a mathematical point of view, the smaller width \( c \) is, the faster Gaussian function value drops and vice versa. Thus the width of Gaussian has direct influence on relative weight of each model. As an example, the switching scheme used in previous cases can be roughly regarded as the case where width is sufficiently small. Theoretically, it is our opinion that the width selection is mainly dependent on the degree of similarities between operating conditions. More concretely, as for several entirely different environments, if a Gaussian function with overly large width is utilized, then invalid control inputs (i.e., control input derived from inaccurate models) would account for a large proportion in overall control input, which would definitely yield poor control performance. In our case study, the Gaussian width is chosen as \( c = 0.8 \) by comparing partitioned operating conditions with each other plus multiple trials. Next, on the basis of control system constructed in Part C, global control is realized without varying the configuration of model parameters. Relevant simulation results are presented in FIGURE. 14. In the second two figures present plant responses and control error respectively, and relative weights for all models during the simulation is shown in the third figure. It follows from FIGURE. 14 that the transient error, which arises during the switching of two operating conditions, is acceptable, however, the steady-state error fails to decay to zero asymptotically. Based on simulation results, we conjecture that this type of control mode may be suited for eliminating transient error, for the switching of environments is typically a progressive process, and there is practically no explicit boundary between two environments, thus the mutual work of several valid models can yield a smoother transition in some cases. As for steady-state control, tuning with adaptive models is normally sufficient, the incorporation of multiple local model may cause control deviation during this process.

**VI. CONCLUSION**

Nowadays, with the easy access to a big volume of data in industrial process, the data-driven intelligent control has become realisable. In this work, the divide and conquer strategy is fully used to address the control problem in the flue gas desulfurization. Based on the process data collected in coal-fired power plant, we made detailed analysis on operating conditions such that local models can be derived by parallel dynamic neural network. Then multi-model adaptive control is applied to the control problem in our case study and yield desired control performance. The main contribution of this work is the integration of operating condition information and control techniques. On the other hand, the flexibility and comprehensiveness of proposed control framework are of practical use especially in industrial process, which is also illustrated in our work.

**ACKNOWLEDGMENT**

The authors would like to appreciate the editor and reviewers for their helpful comments and suggestions to improve the quality of this article.

**REFERENCES**

[1] B. Jiang, Y. Xie, D. Xia, and X. Liu, “A potential source for PM2.5: Analysis of fine particle generation mechanism in Wet Flue Gas Desulfurization System by modeling drying and breakage of slurry drople,” Environ. Pollut., vol. 246, pp. 249–256, Mar. 2019.

[2] T. Lan, X. Zhang, Q. Yu, and L. Lei, “Study on the relationship between absorbed S (IV) and pH in the seawater flue gas desulfurization process,” Ind. Eng. Chem. Res., vol. 51, no. 12, pp. 4478–4484, 2012.

[3] (2014). National Development and Reform Commission (China). Accessed: Sep. 2014. [Online]. Available: http://www.ndrc.gov.cn/ztzl/201409/20140919_626240.html

[4] G. Pipitone and O. Bolland, “Modeling of oxy-combustion flue gas desulfurization by seawater absorption,” Environ. Prog. Sustain. Energy, Off. Publication Amer. Inst. Chem. Eng., vol. 28, no. 1, pp. 20–29, 2009.

[5] D. Kong, Y. Zhang, N. Li, Q. Zhou, R. Luo, and T. Xu, “Experimental investigation on gas-liquid flow, heat and mass transfer characteristics in a dual-contact-flow absorption tower,” Chem. Eng. Res. Des., vol. 92, no. 1, pp. 13–24, 2014.

[6] P. Córdoba, “Status of flue gas desulfurisation (FGD) systems from coal-fired power plants: Overview of the physical-chemical control processes of wet limestone FGDs,” Fuel, vol. 144, pp. 274–286, Mar. 2015.

[7] A. L. Villanueva-Perales, F. J. G. Ortiz, P. Ollero, and F. M. Gil, “Controllability analysis and decentralized control of a wet limestone flue gas desulfurization plant,” Ind. Eng. Chem. Res., vol. 47, no. 24, pp. 9931–9940, Dec. 2008.

[8] J. Li, K. Shen, C.-C. Zhou, H.-T. Xu, X.-Q. Zhu, and W. Wang, “Optimization for pH value control of flue gas desulfurization tower,” Huajing Xuebao ya Jishu, vol. 35, no. 4, pp. 43–46, 2012.

[9] S. S. Guo and Z. Dong, “Application of DMC algorithm on pH control of WFGD system,” Appl. Mech. Mater., vol. 568–570, pp. 1090–1094, Jun. 2014.

[10] Y. Wu, L. Shen, and L. Zhang, “Study on nonlinear pH control strategy based on external recurrent neural network,” Procedia Eng., vol. 15, pp. 866–871, 2011.

[11] Z.-K. Yang, C.-Y. Liu, X.-L. Song, Z.-Y. Song, and Z.-S. Wang, “Application of RBF neural network PID in wet flue gas desulfurization of thermal power plant,” in Proc. Int. Conf. Mach. Learn. Cybern. (ICMLC), Jul. 2016, pp. 301–306.

[12] X.-L. Li, C. Jia, D.-X. Liu, and D.-W. Ding, “Nonlinear adaptive control using multiple models and dynamic neural networks,” Neurocomputing, vol. 136, pp. 190–200, Jul. 2014.

[13] R. del Valle-Zermeno, J. Formosa, and J. M. Chimeneos, “Wet flue gas desulfurization using alkaline agents,” Rev. Chem. Eng., vol. 31, no. 4, pp. 303–327, Jan. 2015.

[14] F. J. Gutiérrez Ortiz, “A simple realistic modeling of full-scale wet limestone FGD units,” Chem. Eng. J., vol. 165, no. 2, pp. 426–439, Dec. 2010.

[15] M. Glomba, “Technical description of parameters influencing the pH value of suspension absorbed used in flue gas desulfurization systems,” J. Air Waste Manage. Assoc., vol. 60, no. 8, p. 1009, 2010.

[16] D. Verma, P. Goel, V. Patil-Shinde, and S. S. Tambe, “Use genetic programming for selecting predictor variables and modeling in process identification,” in Proc. Indian Control Conf. (ICC), Jan. 2016, pp. 230–237.

[17] R. E. Bellman and S. E. Dreyfus, Applied Dynamic Programming, Princeton, NJ, USA: Princeton Univ. Press, 2015.

[18] A. Liaw and M. Wiener, “Classification and regression by random forest,” R News, vol. 2, no. 3, pp. 18–22, 2002.

[19] X. Wang, T. Liu, X. Zheng, H. Peng, J. Xin, and B. Zhang, “Short-term prediction of groundwater level using improved random forest regression with a combination of random features,” Appl. Water Sci., vol. 8, no. 5, p. 125, Sep. 2018.

[20] G. M. Boratyn, S. Datta, and S. Datta, “Biologically supervised hierarchical clustering algorithms for gene expression data,” in Proc. Int. Conf. IEEE Eng. Med. Biol. Soc., Aug. 2006, pp. 5515–5518.
[21] F. Gullo, G. Ponti, A. Tagarelli, and S. Greco, “An information-theoretic approach to hierarchical clustering of uncertain data,” Inf. Sci., vol. 402, pp. 199–215, Sep. 2017.

[22] O. Arbelaitz, I. Gururutxaga, J. Muguerza, J. M. Pérez, and I. Perona, “An extensive comparative study of cluster validity indices,” Pattern Recognit., vol. 46, no. 1, pp. 243–256, Jan. 2013.

[23] R. Layton, P. Watters, and R. Dazeley, “Evaluating authorship distance methods using the positive silhouette coefficient,” Natural Lang. Eng., vol. 19, no. 4, pp. 517–535, Oct. 2013.

[24] H. B. Zhou and J. T. Gao, “Automatic method for determining cluster number based on silhouette coefficient,” Adv. Mater. Res., vol. 951, pp. 227–230, May 2014.

[25] S. Lukasik, P. A. Kowalski, M. Charytanowicz, and P. Kulczycki, “Clustering using flower pollination algorithm and calinski-harabasz index,” in Proc. IEEE Conge. Evol. Comput. (CEC), Jul. 2016, pp. 2724–2728.

[26] X.-L. Li, C. Jia, K. Wang, and J. Wang, “Trajectory tracking of nonlinear system using multiple series-parallel dynamic neural networks,” Neurocomputing, vol. 168, pp. 1–12, Nov. 2015.

[27] W. Yu and A. S. Poznyak, “Indirect adaptive control via parallel dynamic neural networks,” IEEE Proc. - Control Theory Appl., vol. 146, no. 1, pp. 25–30, Jan. 1999.

[28] J. Zhu, S.-C. Ye, J. Bai, Z.-Y. Wu, Z.-H. Liu, and Y.-F. Yang, “A concise algorithm for calculating absorption height in spray tower for wet limestone-gypsum flue gas desulfurization,” Fuel Process. Technol., vol. 129, pp. 15–23, May 2014.

[29] K. S. Narendra and A. M. Annaswamy, Stable Adaptive Systems. Chelmsford, MA, USA: Courier Corporation, 2012.

[30] K. S. Narendra, O. A. Driollet, M. Feiler, and K. George, “Adaptive control using multiple models, switching and tuning,” Int. J. Adapt. Control Signal Process., vol. 17, no. 2, pp. 87–102, 2003.

[31] K. S. Narendra, J. Balakrishnan, and M. K. Ciliz, “Adaptation and learning using multiple models, switching, and tuning,” IEEE Control Syst. Mag., vol. 15, no. 3, pp. 37–51, Jun. 1995.

[32] K. S. Narendra, J. Balakrishnan, and M. K. Ciliz, “Adaptation and learning using multiple models, switching, and tuning,” IEEE Control Syst. Mag., vol. 15, no. 3, pp. 37–51, Jun. 1995.

[33] J. M. Douglas, Process Dynamics and Control: Control System Synthesis, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1972.

[34] W. F. Arnold and A. J. Laub, “Generalized eigenproblem algorithms and software for algebraic Riccati equations,” Proc. IEEE, vol. 72, no. 12, pp. 1746–1754, Dec. 1984.

[35] K. Zhang, H. Zhang, Y. Mu, and C. Liu, “Decentralized tracking optimization control for partially unknown fuzzy interconnected systems via reinforcement learning method,” IEEE Trans. Fuzzy Syst., early access, Jan. 13, 2020, doi: 10.1109/TFUZZ.2020.2966418.

XIAOLI LI received the B.E. and M.E. degrees from the Dalian University of Technology, in 1994 and 1997, respectively, and the Ph.D. degree from Northeastern University, China, in 2000. From 2000 to 2003, he was a Postdoctoral Research Fellow with Tsinghua University, China, and the Université Libre de Bruxelles, Belgium. He is currently a Professor with the Beijing University of Technology. His research interests include intelligent control, multiple model control, adaptive control, and robust control.

QUANBO LIU received the B.E. degree in control engineering from the Beijing University of Technology, in 2017, where he is currently pursuing the M.S. degree in control engineering. His research interests include multiple model control and adaptive control.

KANG WANG received the B.E. and Ph.D. degrees in control science and engineering from the School of Automation and Electrical Engineering, University of Science and Technology Beijing, in 2012 and 2018, respectively. He is currently a Postdoctoral Researcher with the Faculty of Information Technology, Beijing University of Technology. His research interests include artificial neural networks, optimal control, and intelligent control.

FUQIANG WANG received the B.E. and Ph.D. degrees from North China Electric Power University, in 2003 and 2013, respectively. From 2013 to 2015, he was a Postdoctoral Research Fellow with Shenhua Guohua (Beijing) Electric Power Research Institute Corporation, China, where he is currently a Senior Engineer. His research interests include intelligent control, multiple model control, adaptive control, system identification, and predictive control.

GUIMEI CUI received the B.E. degree from Baotou Iron and Steel Institute, in 1985, the M.E. degree from the Harbin Institute of Shipping, in 1988, and the Ph.D. degree from the Beijing University of Science and Technology, China, in 2006. Since 2000, she has been the Deputy Dean of the School of Information Engineering, Inner Mongolia University of Science and Technology, responsible for the laboratory construction and graduate training of the school. Her research interests include complex system modeling and optimal control.

YANG LI received the bachelor’s degree from Shenyang University, in 1999, the master’s degree from Shenyang Normal University, in 2003, and the Ph.D. degree from the Chinese Academy of Social Sciences, in 2006. She is currently a Lecturer with the School of International Studies, Communication University of China (CUC). Her research interests include intercultural communications, translating, and automation.

***