IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received May 8, 2020, accepted May 27, 2020, date of publication June 1, 2020, date of current version June 16, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.2998792*

# STA-APSNFIS: STA-Optimized Adaptive Pre-Sparse Neuro-Fuzzy Inference System for Online Soft Sensor Modeling

**JINPING LIU[1], (Member, IEEE), CHURONG JIANG[1], JIEZHOU HE[1], ZHAOHUI TANG[2], YONGFANG XIE[2], (Member, IEEE), AND PENGFEI XU[1]**

[1]Hunan Provincial Key Laboratory of Intelligent Computing and Language Information Processing, Hunan Normal University, Changsha 410081, China
[2]School of Automation, Central South University, Changsha 410083, China

Corresponding author: Jinping Liu (ljp202518@163.com)

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61971188 and Grant 61771492, in part by the National Science Fund for Distinguished Young Scholars under Grant 61725306, in part by the Joint Fund of the National Natural Science Foundation of China and the Guangdong Provincial Government under Grant U1701261, in part by the Hunan Natural Science Fund under Grant 2018JJ3349, in part by the Foundation of Hunan Educational Committee of China under Grant 19B364, in part by the Key Laboratory of Minister of Education for Image Processing and Intelligence Control (Huazhong University of Science and Technology) under Grant IPIC2017-03, and in part by the Postgraduate Student Research and Innovation Projects of Hunan Province under Grant CX2018B31 and Grant CX20190415.

**ABSTRACT** In complex industrial processes (CIPs), due to technical and economic limitations, key performance indicators (KPIs), especially the chemical content-related KPIs, are often difficult to measure in real time, which hinders the propagation of advanced process control technologies. This paper presents a soft sensor-based online KPI inference scheme by a state transition algorithm (STA)-optimized adaptive pre-sparse neuro-fuzzy inference system model, called STA-APSNFIS. It introduces a pre-sparse neural network to the traditional adaptive neuro-fuzzy inference system (ANFIS) model to establish an adaptive pre-sparse neuro-fuzzy inference system (APSNFIS) model to alleviate the adverse effects of data redundancy and noise interference in the detectable process monitoring data, which can effectively reduce the complexity of neuro-fuzzy inference system (NFIS) and speed up its convergence. Successively, to avoid being trapped at a local optimum, the STA-based optimization algorithm is adopted to replace the traditional gradient-based optimization approach to achieve an optimal APSNFIS model. Extensive validation and comparative experiments on nonlinear numeric simulation systems, benchmark Tenessee Eastman (TE) process and a real industrial bauxite flotation process demonstrated that the proposed STA-APSNFIS performed favorably against traditional ANFIS model as well as its variants, e.g., PSO-ANFIS, GA-ANFIS, and some other soft sensor-based KPI inference models.

**INDEX TERMS** Soft sensor modeling, complex industrial process, key performance indicator (KPI), adaptive neuro-fuzzy inference system (ANFIS), state transition algorithm (STA).

## I. INTRODUCTION

Complex industrial processes (CIPs) [1] are often composed of closely-coupled sub-circuit processes, involving a series of strongly-coupled equipment. In the process production, many complex physico-chemical reactions occur in the equipment, usually acting in gas-liquid-solid three-phase mixtures [2]. Specifically, CIPs involve a large amount of non-linear, time-varying parameters correlated with each other, but the detectable process monitoring information is often incomplete or lagged. Hence, many CIPs are often running in a non-ideal operating condition. One of the key reasons is that it is intractable or unavailable to detect some key production indicators (KPIs) of CIPs timely by hardware sensors [3], [4], especially the chemical composition-relevant KPIs, which are mainly obtained by infrequent on-site manual sampling with laboratory assays [5].

Taking a bauxite flotation process as an instance, it is a typical CIP, involving multi-stage sequential sub-circuit processes, such as rougher, scavenger and cleaner, strongly

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

coupled each other [6]–[8]. The key performance evaluation indicators, such as concentrate grade (reflecting the content of aluminiumoxide in the concentrate product, measured by the alumina-to-silica ratio (A/S) in the flotation plant) and recovery rate (reflecting the recovering level of valuable minerals) in each stage, are still unavailable for the on-line flotation process monitoring. Currently, the concentrate grade and recovery rate are largely depended on the laboratory assays, which is a labor-intensive and long-lagged task. Thus, only a few concentrate grade samples can be achieved in the daily process monitoring and each assayed sample has a several-hour delay, leading to the improper control or operation adjustment frequently with concentrate grade fluctuations and low recovery ratios of mineral ores, due to the several-hour delay and scarceness of the concentrate quality indicators.

To overcome the measuring difficulties of product quality-relevant indexes, many soft sensor-based online KPI-inferential models can be found in the literature. Soft sensors are generally inferential models or systems to estimate or predict the KPI by detectable industrial parameters related to the KPI that are easy to measure, which have received extensive attention [9]–[11], [13]. As summarized in Bidar's work [3], [4], soft sensors can be categorized into the model-based and data-driven models. Model-based soft sensors are generally developed based on first principle models, thus they require the in-depth process knowledge. Due to the inherent chaos and complexity of a CIP, mechanism model-driven soft sensors are often impractical in the CIP monitoring.

Currently, most soft sensor models are data-driven methods that use detectable process monitoring data to build inferential models for the online KPI detection. Some multivariate statistics-related regression techniques have attracted extensive attention due to their theoretical simplicities and elaborate mathematic frameworks. Commonly-used methods include the linear models, such as multiple linear regression (MLR) model, linear feature extraction and regression-integrated models, e.g., the principal component analysis (PCA) [14] or independent component analysis (ICA) [15]-based key feature extraction of process monitoring data with a regression model; and principal component regression (PCR) or partial least squares regression (PLSR) [16] models, and so on.

Since linear models cannot fit the complex nonlinear relations in CIPs, researchers pay more attention to the nonlinear models, such as artificial neural networks (ANN) and support vector machines (SVM) [17]. In addition, fuzzy inference-based soft sensor models and Gaussian process regression (GPR) models [18]–[21] are also widely used. For instance, Pani and Mohanta [22] proposed a Takagi-Surgeon fuzzy inference model, which is a multimodal approach, combining multiple linear sub-models to describe the global nonlinear behaviors of CIPs, for the online monitoring of cement clinker quality. In addition, the flourishing variants of abovementioned basis models, such as the recursive least-squares support vector regression (RLSSVR) approach for the online modeling of batch processes with uneven operating durations [23], dynamic fuzzy neural network (D-FNN) method for the KPI prediction of convention velocity of vinyl chloride monomer (VCM) in the polyvinylchloride (PVC) polymerizing process [24], and so on.

Although considerable soft sensor models have been proposed and many of them were reported to be applied in real industrial processes for the online KPI detection, most of them have their own limitations in the real application. Generally, most soft sensor models have low interpretability and they cannot associate system knowledge to improve their system modeling ability for CIP monitoring. For example, though ANN-based soft sensor models have good self-learning and association capability with high-precision fitness ability and no manual intervention, they cannot process and describe fuzzy information and knowledge and they are easy to cause over-fitting due to the noise interference in CIPs, and their working mechanism is not interpretable.

Fuzzy inference system (FIS), using the decision mechanism to express the uncertainty in form of rules, which can well reduce the noise and error interference in the incomplete and noisy industrial process monitoring data, has attracted increasing attention in the online soft sensor modeling of KPIs. However, its adaptive ability is inadequate, which hinders its popularization [25]. Adaptive neuro-fuzzy inference system (ANFIS) is a hybrid model that combines the learning mechanism of ANNs and the fuzzy inference ability of FISs, which has been applied to industrial process monitoring and achieved the state-of-the-art system modeling results [26], [27].

The original ANFIS model achieves the network parameters by solving a least squares problem using the gradient-based optimization algorithm to fit the nonlinear relations in CIPs. It has demonstrated that the converging speed of an ANFIS is slow and it is easy to fall into a local minimum of the nonlinear system [28]. The FIS's structure makes its rule number and parameter quantity increase with a power-law relationship with the number of input features. Hence, the computational cost of the ANFIS-based soft sensor model is expensive and the model converges slowly with the increasing number of input features. Worse still, models with too many parameters are easy to lead to over fitting.

To summarize, despite the strong learning power of ANNs and explicit knowledge representation of FISs, traditional ANFIS-based soft sensor is still intractable to fit complex nonlinear relations in a CIP effectively due to the high-dimensional, redundant, and strong noise-interfering parameters in the CIP. Thus, it will generate a very large network structure that consumes a lot of time and memory space, making it difficult to meet the requirements of the online KPI monitoring for the CIP monitoring.

To achieve a highly efficient and noise-tolerant soft sensor model, this paper proposed a state transition algorithm (STA)-induced adaptive pre-sparse neuro-fuzzy inference system, termed STA-APSNFIS, for the soft modeling of KPIs in CIPs. STA-APSNFIS uses a pre-sparse neural network to improve the traditional ANFIS model, which can effectively reduce the dimensions and reduce the interference of redundancy, inaccuracy and noise in process monitoring data to achieve a highly-efficient and robust model. In addition, the optimal APSNFIS model parameters are achieved by introducing a STA-based optimization algorithm, rather than the traditional gradient-based optimization algorithm. Experiments demonstrated that the proposed STA-APSNFIS model can establish a stable and effective soft sensor model for the online KPI monitoring of CIPs. The main contributions of this paper are threefold.

- A novel end-to-end soft sensor model, termed STA-APSNFIS, for the online KPI monitoring of CIPs, is proposed.
- STA-APSNFIS introduces a pre-sparse neural network to the traditional ANFIS model to make a sparse representation of process monitoring data, which can reduce the adverse interferences of data redundancy and noise in the process monitoring data to ensure the online inferential ability of the FIS for the CIP monitoring.
- The STA optimization algorithm is introduced to optimize the parameters of the APSNFIS model, which can improve the fitting effect and speed up the convergence of the inference model.

Extensive confirmatory and comparative experiments, including numeric simulation systems, benchmarking TE process and a real industrial flotation process, showed that the proposed STA-APSNFIS model had faster convergence speed and higher computational efficiency for the online KPI detection. STA-APSNFIS can achieve a better parameter optimization performance (nonlinear data fitting) than the ANFIS-relevant mainstream models, e.g., PSO-ANFIS and GA-ANFIS, which can effectively avoid the local extremum achieved by the traditional gradient-based optimization algorithm in the model learning.

The remainder of this paper is organized as follows. Section II briefly reviews the basic principles of ANFIS and the main steps of STA. Section III addresses the main thoughts and detailed steps of the proposed STA-APSNFIS model, followed by numeric simulation experiments with confirmatory and comparative case studies on the benchmarking TE process and an industrial bauxite flotation process described in Section IV. Section V concludes the whole paper with possible extensions of this paper.

## II. PRELIMINARIES

This section briefly overviews the basic principle of ANFIS and the main steps of the STA optimization algorithm.

### A. ANFIS

By combination of ANN and fuzzy system, ANFIS [29] is a class of adaptive multi-layer feedforward networks based on the T-S fuzzy inference model. It uses fuzzy neural network to realize three basic processes, fuzzy control, fuzzy reasoning and defuzzification, by the learning mechanism of ANNs to

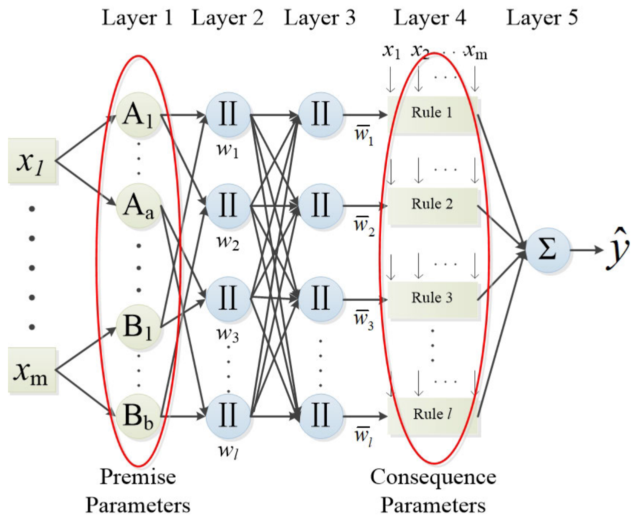



**FIGURE 1.** Schematic of ANFIS.

automatically extract rules from input and output samples. The extraction rules constitute an adaptive neuro-fuzzy controller. Offline training and online updating algorithms are used to adjust the fuzzy rules, so that the system has the properties of adaptive, self-organizing and self-learning. The ANFIS architecture is show in Fig. 1.

As can be seen from Fig.1, an ANFIS consists of five layers. Each layer includes multiple nodes described by a node function.

*Layer 1 (Fuzzy layer):* Fuzzify the input variable and output the membership degree of the corresponding fuzzy set. Each node $i$ of the layer is an adaptive node with a node function, given by,

$$O_i^1 = \mu_{A_i}(x) \quad i = 1 \cdots N \tag{1}$$

where $x$ is the input to node $i$, and $A_i$ is the linguistic label (e.g., small, medium, large, huge); $O_i^1$ is the membership of the fuzzy set associated with $A_i$. The membership function $\mu$ can be any suitable parameterized membership function, such as the bell function with maximum of 1 and minimum of 0, defined by,

$$\mu_A(x) = \frac{1}{1 + \left(\frac{x-c_i}{a_i}\right)^{2b_i}} \tag{2}$$

where $\{a_i, b_i, c_i\}$ is the parameter set, referred to as a premise parameter set. With the change of the values of the parameter set, the membership on the linguistic label $A_i$ will vary accordingly.

*Layer 2 (Product layer):* Each node $i$ collects the algebraic product of all input signals, given by,

$$O_i^2 = \omega_i = \mu_{A_i}(x) \times \mu_{B_i}(x) \cdots \quad i = 1 \cdots M \tag{3}$$

where $M$ is the number of nodes in the first layer. The output $\omega_i$ represents the activation strength of that rule, and the activation function of this layer can also takes the form of minimum, bounded product or strong product.

*Layer 3 (Normalization layer):* Normalize the activation strength of each rule, and the output of each node is the ratio of its activation strength to the sum of activation strength of all rules, namely,

$$O_i^3 = \bar{\omega} = \frac{\omega_i}{\omega_1 + \ldots + \omega_M} \quad i = 1 \cdots M. \tag{4}$$

*Layer 4 (Defuzzification layer):* Each node $i$ is an adaptive node with a node function, used to calculate the contribution of the $i$ th rule to overall output, defined by,

$$O_i^4 = \bar{\omega}_i l_i = \bar{\omega}_i \left( \sum_{j=1}^{n} p_j x_j + r_i \right) \tag{5}$$

where $\bar{\omega}$ is the normalized firing strength from the third layer, and $\{p_1, p_2, \ldots, p_n, r\}$ is the consequent parameter set.

*Layer 5 (Total output layer):* The output of an ANFIS is obtained by summing the outputs obtained by each rule in the defuzzification layer, i.e.,

$$O_i^5 = \sum_i \bar{\omega}_i l_i = \frac{\sum_i \omega_i l_i}{\sum_i \omega_i} \tag{6}$$

Training ANFIS means determination of these parameters using an optimization algorithm [30]. Since the introduction of ANFIS at the first time [29], researchers have proposed many different optimization algorithms to achieve the ANFIS model parameter [31]. These methods can be divided into three types: derivative-based, heuristic-based and hybrid methods, in which heuristic-based parameter optimization methods can generally achieve the relative better results, such as PSO-ANFIS, GA-ANFIS, and so on.

However, learning an ANFIS model with high dimension input data is still intractable. As can be seen from Fig.1, there are two different parameter sets, i.e., the premise parameter set and the consequence parameter set, to be determined. The number of model parameters will increase exponentially with the increase of the dimension of input data. When the feature dimension of input data is great, the complexity of the model learning increases significantly and it is easy to achieve over fitting model learning results due to the commonly existing optimization difficulties in the high dimension parameter space. For Instance, when the input data dimension is $n$ and each feature is divided into $a$ membership degrees, the number of premise parameters is,

$$\text{premise parameter numbers} = 3 \times a^n, \tag{7}$$

and the number of consequent parameters is

$$\text{consequent parameter numbers} = (n+1) \times a^n. \tag{8}$$

### *B. STA*

State transition algorithm (STA) [32], [33] is a recently proposed intelligent optimization algorithm based on the idea of state space transformation. In the initial version, proposed by Zhou *et al.* in 2011 [34], STA has three operators: rotation, expansion and translation for solving the continuous optimization problem. Subsequently, Zhou *et al.* proposed an improved STA in 2013 [35]. Based on the original three operators, a new operator, named axesion transformation, is included in the STA to improve the search ability of a certain dimension.

To avoid falling into the local optimum, a corresponding suboptimal solution selection mechanism is introduced in the latest version of the STA. Compared with genetic algorithms (GA), particle swarm optimization (PSO) and differential evolution algorithm (DE), STA has stronger global search ability, better search accuracy and faster convergence speed [35].

Without loss of generality, considering the following minimum optimization problem,

$$\min_{x \in \Re^n} f(x). \tag{9}$$

The state transition process has the following form:

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k \\ y_{k+1} = f(x_{k+1}) \end{cases} \tag{10}$$

where $x_k \in \Re^n$ represents a state that corresponds to a solution to the optimization problem; $A_k$ with $B_k$ are state transition matrices, which can be considered as operators of the optimization algorithm; $u_k$ is a function related to state $x_k$ and historical state; $f(\bullet)$ is the corresponding fitness function.

#### 1) ROTATION TRANSFORMATION

$$x_{k+1} = x_k + \alpha \frac{1}{n \|x_k\|_2} R_r x_k \tag{11}$$

where $x_k \in \Re^n$, $\alpha$ is a positive constant, called the rotation factor, $n$ as the number of variables; $\|\bullet\|_2$ is 2-norm of vector or Euclidean norm; $R_t \in \Re^{n\times n}$ is a uniformly distributed random matrix between $[-1, 1]$. The rotation transformation has the function of searching in a hypersphere with the maximal radius $\alpha$.

#### 2) TRANSLATION TRANSFORMATION

$$x_{k+1} = x_k + \beta R_t \frac{x_k - x_{k-1}}{\|x_k - x_{k-1}\|_2} \tag{12}$$

where $\beta$ is a positive constant, named translation factor; $R_t \in \Re$ is random number that is uniformly distributed between $[0, 1]$, which makes the algorithm searing along a line from $x_{k-1}$ to $x_k$ in the positive direction of the gradient with the maximum length of $\beta$.

#### 3) EXTENSION TRANSFORMATION

$$x_{k+1} = x_k + \gamma R_e x_k \tag{13}$$

where $\gamma$ is a positive constant, called expansion factor; $R_e \in \Re^{n\times n}$ is a random diagonal matrix obeying Gaussian distribution. The extension operator is able to search the entire search space.

#### 4) AXESION TRANSFORMATION

$$x_{k+1} = x_k + \delta R_a x_k \tag{14}$$

where $\delta$ is a positive constant, named axesion factor; $R_a \in \Re^{n\times n}$ is a random diagonal matrix obeying a Gaussian distribution, and only one element in the random position in the matrix is not zero. The axesion operator can search along the axis of a variable. Thus, it can improve the searching ability of single dimension.

STA is analogous to the other heuristic algorithms, which is based on the individual's iterative search in a neighborhood. As summarized in [36], in the STA iteration, the *rotation* transformation operator has the ability of local searching, and the rotation factor $\alpha$ can adjust the searching range by exponentially reducing from a maximum value $\alpha_{\max}$ to a minimum value $\alpha_{\min}$. The *translation* operator has the function of line search. The *expansion* transformation has the ability of global search and the axesion transformation is used to strengthen the ability of single dimensional search as well as the global search.

The searching procedure of a STA is only related to the optimal population value of the previous generation and the searches are confined within the neighborhood of the optimal value of this population. Hence, STA has the ability of fast and global search.

## III. PROPOSED STA-APSNFIS

This section details the model structure of the proposed STA-APSNFIS with a theoretical analysis of its computational complexity, followed by the main steps of the STA-APSNFIS-based online KPI-inferential procedure. In addition, the feasibility and robustness of the proposed STA-APSNFIS are verified by numerical simulation experiments.

### A. STA-APSNFIS

ANFIS has a good performance of nonlinear system modeling, but its model structure (model parameter number or the final fuzzy rule number) leads to an exponential increasing trend with the dimensional increase of the input data feature. To prevent the structure of the inference system from being too large to train, for the high-dimensional data system, some researchers firstly adopted the dimension reduction algorithm to preprocess the original data set, and then used the ANFIS method to the system modeling. But traditional dimension reduction methods, such as PCA [14], LDA [37], Auto-Encoder [38] and Laplacian Eigenmaps [39], are generally based on the retention of the original data from feature mapping, which ignore the correlations between the data features and the data tags.

Since the process data of a CIP has the property of high dimension, high information redundancy, and uncorrelated data (noise) interference, traditional feature selection method has little help for the final fitting effect of the ANFIS model. Therefore, we introduce a pre-sparse neural network model to improve the data processing ability of the ANFIS model.

The pre-sparse neural network maps the input data features from a high-dimensional space to a low-dimensional space, which can effectively reduce the complexity of the FIS model. The feature mapping parameter matrix depends on the minimum error between the predicted result and the real result. The least square loss function is used to constrain the mapping matrix of the pre-sparse neural network to ensure that

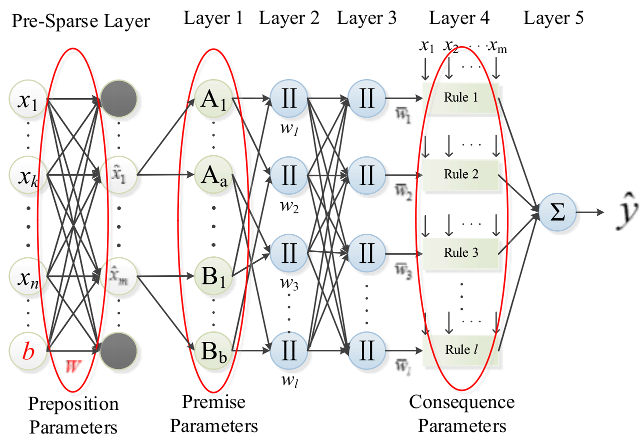



**FIGURE 2. Schematic structure of APSNFIS.**

the mapped data can accurately and effectively predict the target value; thereby it can effectively remove irrelevant data features, decrease the information redundancy and reduce the feature dimension. The APSNFIS model structure is shown in Fig. 2.

As can be seen from Fig.2, by introducing a pre-sparse neural network, the APSNFIS is divided into six layers. The last five layers are consistent with the traditional ANFIS model, and the pre-sparse neural network is located between the input and the original first layer, which is a pre-sparse layer. Each node $i$ in the pre-sparse layer is adaptive neurons with an activation function, and the number of neurons is consistent with the number of input parameters. Its output is used as the input to the ANFIS model, and its output is calculated as,

$$a_i = f\left(\sum_{j=1}^{n} w_i^j x_j + b_i\right), \tag{15}$$

where $w$ is the weight parameter of full connection, $b$ stands for the bias parameter. They are the set of adaptive parameters of the pre-sparse layer; $x$ is a $n$ dimension input variables; $f(\bullet)$ is the activation function, e.g., sigmoid, ReLU and Tanh.

To reduce the complexity of the FIS model, the output of the pre-layer should be minimized, i.e.,

$$\min\left\{\hat{\rho}_i = \frac{1}{m}\sum_{j=1}^{m} a_i^j\left(x^j\right)\right\} \tag{16}$$

where $\hat{\rho}_i$ is the average activation value (output value) of the $i$ nodes of the pre-layer on the input data $x$ with $m$ sample.

Therefore, a sparse penalty term is included to suppress the output of the pre-sparse layer node and ensure the output of the pre-sparse layer neuron node be small enough:

The least (most sparse) output of the pre-layer is used as the input of the FIS model. The KL distance as described in (17) is adopted to measure the average activation value of pre-layer nodes.

$$\mathrm{KL}\left(\rho||\widehat{\rho}_j\right) = \rho\log\frac{\rho}{\widehat{\rho}_j} + (1-\rho)\log\frac{1-\rho}{1-\widehat{\rho}_j} \tag{17}$$

Formula (17) represents the relative entropy of two Bernoulli random variables, whose mean values are $\rho$ and $\widehat{\rho}_j$, respectively, where $\rho$ is a sparsity parameter, generally set as a decimal number close to 0, e.g., set as 0.01; $\widehat{\rho}_j$ is the average activation of the $j$th neuron in the pre-layer. KL-divergence is a standard function for measuring how different two different distributions are.

Then, the loss function of the APSNFIS model can be modified as,

$$J_{APSNFIS} = J_{ANFIS} + \beta \sum_{j=0}^{n} \mathrm{KL}\left(\rho || \hat{\rho}_j\right)$$
$$J_{ANFIS} = \left\| y - \hat{y} \right\|^2. \tag{18}$$

where $y$ is the true value, $\hat{y}$ is the model prediction value; $\beta$ a parameter that is sparsely constrained.

The pre-sparse neural network can effectively learn a set of basis vectors that have a smaller dimension than the original data. Thus, it can effectively reduce the complexity of the traditional FIS model. For example, when the input dimension is $n$ dimension, which is reduced to $m$ dimension through the pre-sparse layer, and each dimension is divided into $a$ membership degrees, then the parameter number of the pre-sparse neural network is,

$$\text{Pre-Sparse Neural Network Number} = n^2 + n \tag{19}$$

The parameter numbers of APSNFIS is:

$$\text{APSNFIS numbers} = 3 \times a^m + (m+1) \times a^m + n^2 + n \tag{20}$$

It can be seen clearly that when the dimension is large, the parameters of the ANFIS model can be greatly reduced by adding a pre-sparse layer, which just introduces a small amount of network parameters (preposition parameters as displayed in Fig.2) but can effectively avoid over fitting in the process of optimization.

In the ANFIS model, a gradient-based method (least squares, gradient descent) is usually applied to learn the model parameters (premise and consequence parameters). However, the gradient-based method is prone to falling into the local optimum. Therefore, some researchers used a meta-heuristic algorithm that uses the PSO or GA with random search properties, i.e., the GA-optimized ANFIS or PSO-optimized ANFIS, attempting to achieve the optimal ANFIS model parameters.

In terms of the merits of STA, this paper proposes a STA-induced APSNFIS model, termed STA-APSNFIS, for the soft sensor modeling of KPIs in CIPs. Compared with the GA or PSO-optimized ANFIS, STA-APSNFIS has better parameter optimization effects and faster fitting speed. The flowchart of STA-APSNFIS is displayed in Fig. 3.

Detailed steps of STA-APSNFIS are summarized as follows.

*Step 1:* Prepare a set of historical process monitoring data and use it to calculate the activation value by (15), and then use the activation value as the input to the first layer of APSNFIS.

*Step 2:* According to the input data dimension of the first layer of APSNFIS, calculate the preposition, premise and consequence parameters number and establish the FIS model.

*Step 3:* Initialize the STA parameters such as rotation factor $\alpha$, translation factor $\beta$, expansion factor $\gamma$, axesion factor $\delta$, minimum error $\varepsilon$, and the maximum number of iterations $T$. Set the current number of iteration, $t = 0$, population number $n$.

*Step 4:* Randomly generate $n$ group of $D$ dimensional APSNFIS model parameter set vector (preposition, premise and consequence parameters), where $D$ is the parameter set size of the APSNFIS. Then, calculate the fitness function value $f$ of each parameter set based on the APSNFIS model. The parameter set with the smallest fitness function value is the parameter set of the optimal state, and it is recorded as $f_{\text{best}}$. Finally, the optimal parameter set is achieved iteratively according to the four operators of the STA procedure. Detailed steps are as follows.

*Step 4.1:* Copy $n$ group of the optimal parameter set and perform the extension operation based on (13) to generate $n$ group of new parameter sets and calculate their minimum fitness function values $f_k$. If $f_k < f_{\text{best}}$, $f_{\text{best}} \leftarrow f_k$ and execute Step 4.5, followed by Step 4.2, otherwise execute Step 4.2 directly.

*Step 4.2:* Copy $n$ group of the optimal parameter set and perform the rotation operation based on (11) to generate $n$ group of new parameter set and calculate their minimum fitness function values $f_k$. If $f_k < f_{\text{best}}$, $f_{\text{best}} \leftarrow f_k$ and go to step 4.5 followed by the Step 4.3, otherwise execute Step 4.3 directly.

*Step 4.3:* Copy $n$ group of the optimal parameter set and perform the axesion operation based on (14) to generate $n$ group of new parameter sets and calculate their minimum fitness function values $f_k$. If $f_k < f_{\text{best}}$, $f_{\text{best}} \leftarrow f_k$ and execute Step 4.5, followed by the Step 4.4; otherwise execute Step 4.4 directly

*Step 4.4:* If the current minimum fitness value is less than the minimum error $\varepsilon$ or the current number of iterations, namely, $t >= T$, execute Step 5, otherwise record $t = t + 1$ and execute Step 4.2.

*Step 4.5:* Copy $n$ group of the optimal parameter set and perform the translation operation based on (12) to generate $n$ group new parameter sets and calculate their minimum fitness function values $f_k$. If $f_k < f_{\text{best}}$, $f_{\text{best}} \leftarrow f_k$.

*Step 5:* The parameter set of the minimum fitness value is used as a parameter set of the APSNFIS model to calculate the output of the APSNFIS model.

## B. COMPLEXITY ANALYSIS

A practical problem with regard to the soft sensor is the time cost of the online testing. If the computational complexity of the soft sensor model is too complex, the online inference time is too long to provide an effective online monitoring of KPIs.

For nonlinear models such as BP network, the computational complexity depends mainly on the number

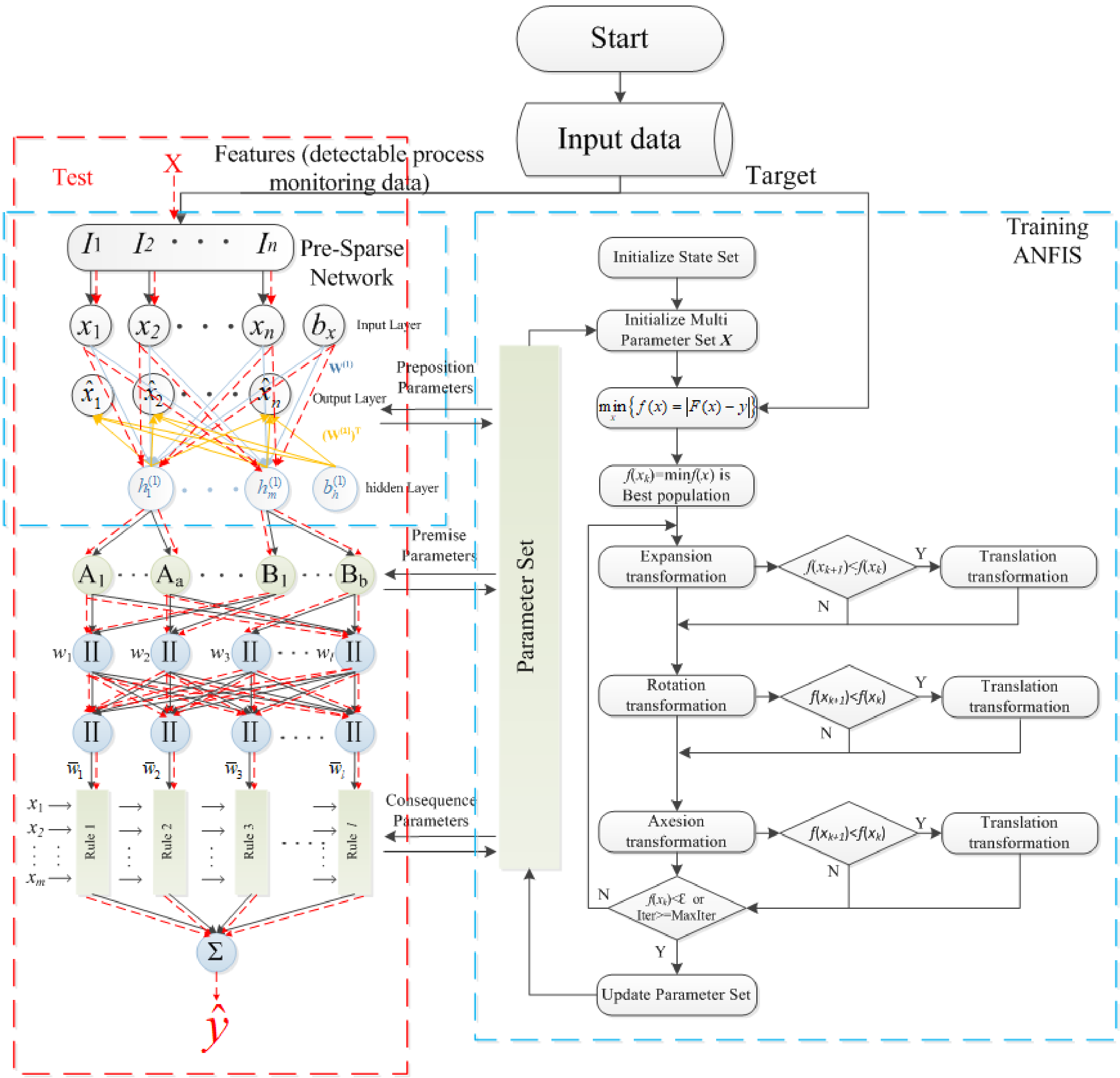



**FIGURE 3.** Flowchart of STA-APSNFIS.

of nonlinear elements, such as sigmoid function and radial basis functions. When the number of network layers is too large, the complexity is increase in power law. With regard to the support vector machines or the kernel tricks-based methods, since the training samples support the regression hyperplane, the computational complexity can be approximated as $O(n_{sv})$, where $n_{sv}$ is the number of support vectors. When the amount of data is larger, SVM will be affected by the number of support vectors, and its computational cost will increase exponentially.

Although the proposed STA-APSNFIS mode consists of six layers of network structure, only the first layer, namely, the pre-sparse network, is a fully connected neural network, whose computational complexity is related to the number of neuron nodes. The remainder five layers network of the STA-APSNFIS model are consistent with the original ANFIS and its time complexity is approximately $O(n)$, where $n$ denotes the feature dimension of input samples. Thus, the total computational complexity is related to the feature dimension of input data.

This paper introduces a pre-sparse neural network to make a sparse representation of the samples features, which can effectively reduce the redundancy in the feature space of the input data. In addition, the sparse constraint can minimize the number of nodes in the pre-sparse layer. Consequently, it can effectively reduce the computational complexity of the FIS and is more conducive to the online soft sensor modeling of CIPs.

### *C. NUMERICAL SIMULATION AND MODEL ROBUSTNESS VERIFICATION*

To verify the effectiveness and robustness of STA-APSNFIS, this section uses the STA-APSNFIS to extract the fuzzy rules of a numeric simulation system, $y = x_1^2 + x_2^2$, in rang of $\{-10 \le x_1 \le 10, -10 \le x_2 \le 10\}$. Firstly, 10,000 samples by this system model with additive Gaussian noise are generated. To validate the robustness of the proposed STA- APSNFIS model, a redundant variable $x_3$ is introduced for model training, i.e., $y = x_1^2 + x_2^2 + 0 \times x_3 + \xi$, where $\xi$ is a Gaussian noise variable.

The STA-APSNFIS pre-sparse layer uses the sigmoid activation function to make the feature reduction of input data from 3 dimensions to 2 dimensions. According to (2), the membership degree calculation of the pre-sparse layer output variables is divided into four rules. Therefore, there are 12 premise parameters and 12 consequence parameters. However, if the original ANFIS model is used, according to (3), the membership degree calculation should be divided into eight rules. Therefore, there are 24 premise parameters and 32 consequence parameters. Therefore, it can be seen that the proposed STA-APSNFIS model effectively reduces the complexity of the model. To achieve stable modeling results, 100 iterations were performed, and the final parameters are shown in Tab. 1.

For a specific input $\{0.6, 7.1, -3.2\}$, adopt (15) to calculate the pro-layer output as $\{0.1371, 0.0794\}$. According to (2), the membership degrees of the four rules with the pre-parameter calculation and its fuzzy model are shown in Fig. 4.

Taking $\{0.8388, 0.4752, 0.3004, 0.8975\}$ into (3) and (4), we can achieve the activation strength of the four rules as $\{0.1601, 0.4783, 0.0907, 0.2709\}$.

Substituting the consequence parameters and the activation strength of each rule into (5) , calculate the output of the input variable on each rule as $\{-2372.1, 2804.8, -1310.8, 1596.7\}$. According to (6), the final output of the model is 718.6510, which is very close to the true value 716.1820 of the test system function $y = x_1^2 + x_2^2$ on $\{0.6, 7.1\}$.

The calculation process has a certain error due to the problem of the inaccuracy (noise interference) of the variables, which is consistent to the real industrial process monitoring. In the computer simulation process, the precision is 14 digits after the decimal point, and the output result is 716.1894. Experiments show that the proposed STA-APSNFIS model can effectively fit the nonlinear system and has strong robustness to the noise contamination and redundancy interference in the process monitoring data.

**TABLE 1.** Rules and parameter setting of STA-APSNFIS.

| | | | | | |
|---|---|---|---|---|---|
| IF | $f_1(x_1,x_2,x_3)$ is $A_1$; $f_2(x_1,x_2,x_3)$ is $B_1$ | Then | $l_1 = p_1 f(x_1) + q_1 f(x_2) + r_1$ | | |
| IF | $f_1(x_1,x_2,x_3)$ is $A_1$; $f_2(x_1,x_2,x_3)$ is $B_2$ | Then | $l_2 = p_2 f(x_1) + q_2 f(x_2) + r_2$ | | |
| IF | $f_1(x_1,x_2,x_3)$ is $A_2$; $f_2(x_1,x_2,x_3)$ is $B_1$ | Then | $l_3 = p_3 f(x_1) + q_3 f(x_2) + r_3$ | | |
| IF | $f_1(x_1,x_2,x_3)$ is $A_2$; $f_2(x_1,x_2,x_3)$ is $B_2$ | Then | $l_4 = p_4 f(x_1) + q_4 f(x_2) + r_4$ | | |
| | | w | | | b |
| Preposition Parameter | $f_1$ | -1.8516 | -0.0088 | -0.0029 | -0.8561 |
| | $f_2$ | -0.0029 | -1.8593 | 0.0021 | -0.8566 |
| | | a | b | c | |
| Premise Parameter | A1 | 0.1223 | 1.9993 | 0.0561 | |
| | A2 | 0.1367 | 2.0002 | 0.2772 | |
| | B1 | 0.1191 | 1.9911 | -0.0679 | |
| | B2 | 0.1589 | 2.0019 | 0.1718 | |
| | | p | q | r | |
| Consequence Parameter | $l_1$ | -1331 | -458020 | 21719 | |
| | $l_2$ | -1659.7 | -28257 | 8334.5 | |
| | $l_3$ | 552.241 | -457330 | 21769 | |
| | $l_4$ | 336.247 | -27718 | 8847.1 | |

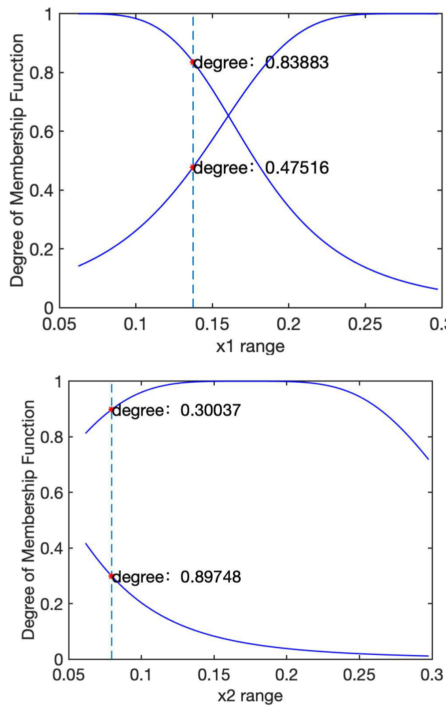



**FIGURE 4.** Fuzzy membership.

## IV. EXPERIMENTAL VALIDATION

To validate the performance of STA-APSNFIS, we carried out experiments from three aspects. (1) Compare the STA-APSNFIS with the fitting effects of ANFIS, GA-ANFIS and PSO-ANFIS on some classic test functions. (2) Validate the effect and compare its performance with other soft sensors on TE process. (3) Carry out comparative experiments on a real flotation industrial process.

The experimental environment is as follows. The hardware configuration is processor: Intel Core i7-8750H, memory: 16.0GB. The software environment is macOS with MATLAB R2018a.

### *A. MODEL PERFORMANCE EVALUATION*

To verify the feasibility of STA-APSNFIS, this section uses some high-dimensional nonlinear functions (listed in

**TABLE 2. Test functions.**

| Function definition | Variable range |
|---|---|
| $f_1 = \sum_{i=1}^{n} x_i^2$ | $[-100,100]$ |
| $f_2 = \sum_{i=1}^{n}\left(x_i^2 - 10\cos\left(2\pi x_i^2\right)+10\right)$ | $[-5.12,5.12]$ |
| $f_3 = \sum_{i=1}^{n}\left(100\left(x_{i+1}^2 + x_i^2\right)^2 + \left(x_i - 1\right)^2\right)$ | $[-30,30]$ |
| $f_4 = \sum_{i=1}^{n}\sin\left(x_i\right)\sin\left(\frac{ix_i^2}{n}\right)^{20}$ | $[0,\pi]$ |
| $f_5 = 3\left(1-x\right)^2 e^{-x^2-(y+1)^2} - 10\left(\frac{x}{5} - x^3 - y^2\right) - \frac{1}{3}e^{-(x+1)^2-}$ | $[-3,3]$ |

Tab.2) to validate the fitting performance of nonlinear systems.

Based on these numeric functions, we compare it with GA-ANFIS [40], PSO-ANFIS [40] and ANFIS [40]. Determining coefficient $R^2$ and root mean square errors (RMSEs) of fitting results were used to evaluate the fitting performance of the proposed STA-APSNFIS. In addition, the running time was also recorded as an evaluation criterion to compare the convergence speed.

If we denote the true function value as $f$ and the model estimation as $f'$, the RMSE and $R^2$ can be computed as,

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(f-f')^2}{n}}, \tag{21}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(|f-f'|)}{\sum_{i=1}^{n}(|f'-f^m|)}. \tag{22}$$

where $f^m = \sum_{i=1}^{n} f'/n$. In order to ensure the stability of the experiment, above functions all have ten-dimensional variables except $f_5$, which is a two-dimensional function. 10000 samples were generated, and the data features were reduced by the PCA algorithm to make a fair comparison. 70% samples were used for training and the remainder 30% samples were used for testing. All results listed in Tab.3 are the average value of 30 independent experiments.

It can be seen from Tab.3 that the STA-APSNFIS has good performance both on the fitting result and the running time on these testing functions. Since the optimization algorithm has a certain randomness, in simple functions such as $f_1, f_3$, the running time of the proposed method is a little longer than the original ANFIS due to the introduction of the pre-sparse layer in the original ANFIS model. However, the proposed STA-APSNFIS can achieve better fitting accuracies. $f_2$ is a complex multi-peak function, whose 3-D profile are shown in Fig. 5. The fitting scatter plot of the STA-APSNFIS and other ANFIS algorithm is shown in Fig. 6

Comparing the statistical results on the test function $f_2$ in Table 3, STA-APSNFIS can effectively avoid falling

**TABLE 3. Test results.**

| Funs | Statistic | ANFIS (PCA) | PSO-ANFIS (PCA) | GA-ANFIS (PCA) | STA-APSNFIS |
|---|---|---|---|---|---|
| $f_1$ | RMSE | 2.4317 | 2.1354 | 2.3125 | **1.1546** |
| | $R^2$ | 0.9587 | 0.9938 | 0.9653 | **0.9992** |
| | Time(s) | **4.56** | 12.48 | 11.97 | 6.47 |
| $f_2$ | RMSE | 7.7463 | 4.5283 | 4.7634 | **1.5375** |
| | $R^2$ | 0.8135 | 0.8863 | 0.8735 | **0.9267** |
| | Time(s) | 79.23 | 47.84 | 57.88 | **22.86** |
| $f_3$ | RMSE | 3.9235 | 2.5347 | 2.7485 | **1.1862** |
| | $R^2$ | 0.9597 | 0.9499 | 0.9336 | **0.9794** |
| | Time(s) | **3.92** | 7.03 | 8.24 | 4.97 |
| $f_4$ | RMSE | 4.3265 | 3.3643 | 3.6437 | **1.3197** |
| | $R^2$ | 0.8979 | 0.9145 | 0.9054 | **0.9435** |
| | Time(s) | 8.61 | 8. 59 | 9.90 | **2.77** |
| $f_5$ | RMSE | 3.9756 | 1.9754 | 2.8317 | **1.3133** |
| | $R^2$ | 0.8539 | 0.9122 | 0.8747 | **0.9574** |
| | Time(s) | 12.53 | 8.22 | 11.94 | **4.57** |

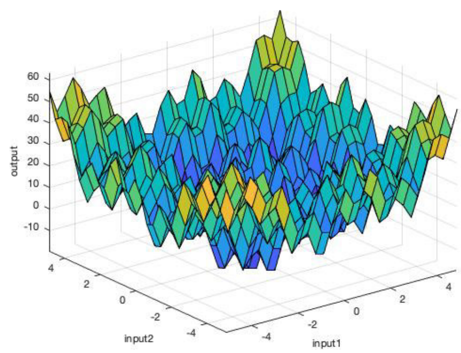



**FIGURE 5.** Three-dimensional profile a two-variate version of $f_2$.

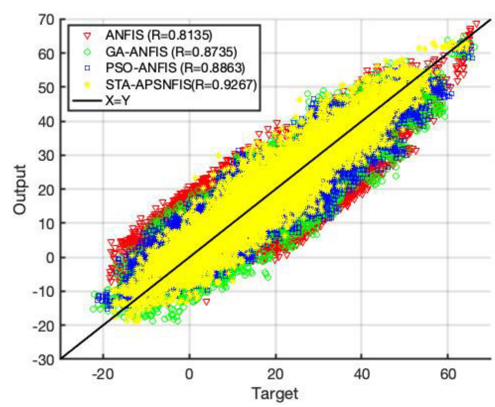



**FIGURE 6.** Scatter plot of the fitting results on $f_2$.

into the local optimum. Compared with GA-ANFIS and PAO-ANFIS, STA-APSNFIS can achieve better fitting results of the complex non-linear functions and has faster fitting speed.

With regard to the performance index, STA-APSNFIS is better than ANFIS and other optimization algorithm-induced ANFIS models in the fitting ability of the nonlinear testing functions. The training speed is faster and the calculation cost is lower. Hence it is more suitable for the online soft sensor modeling in the CIP monitoring.

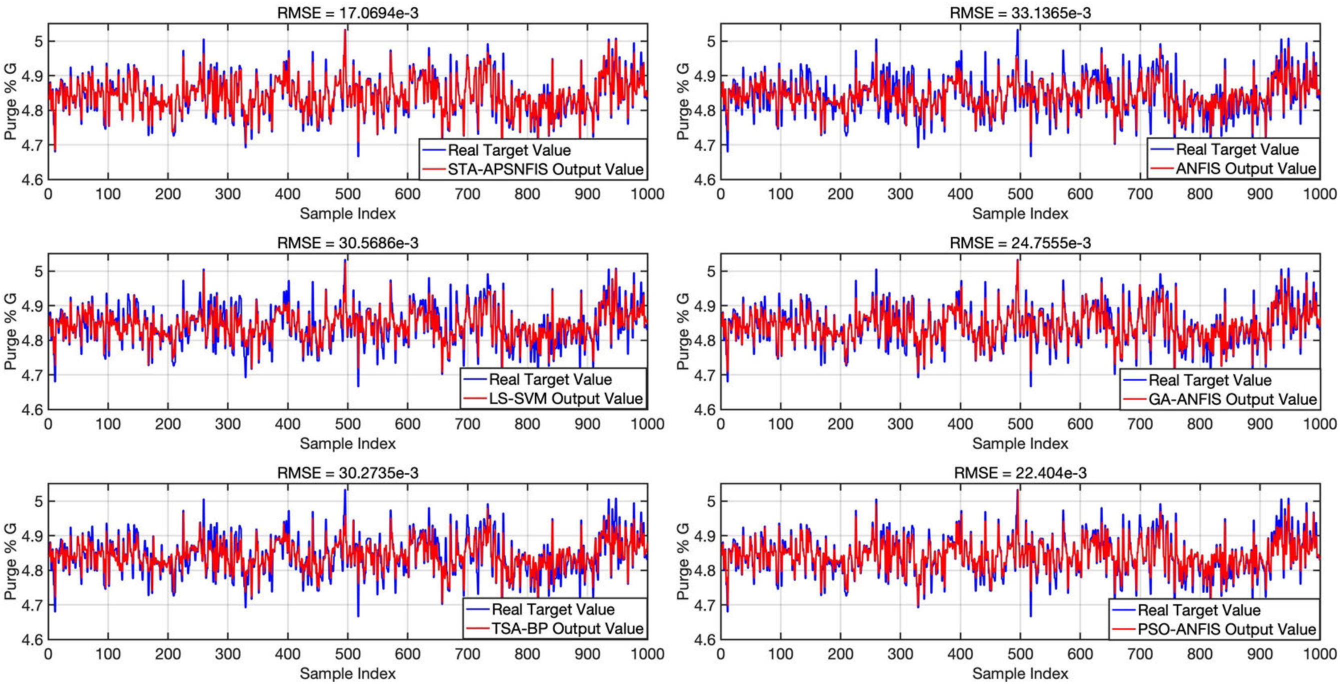



**FIGURE 7.** Purge % G fitting result.

**TABLE 4.** Comparison of KPI soft modeling result on TE Process.

| Products | Statistic | ANFIS | PSO-ANFIS | GA-ANFIS | STA-APSNFIS | LS-SVM | TSA-BP |
|---|---|---|---|---|---|---|---|
| Purge %G (Purge gas composition of component G) | RMSE (e-3) | 33.137 | 22.404 | 24.756 | **17.069** | 30.569 | 30.2735 |
| | $R^2$ | 0.7370 | 0.8798 | 0.8532 | **0.9302** | 0.7762 | 0.7805 |
| | Time(s) | 57.23 | 41.19 | 45.98 | 38.77 | 48.14 | **36.68** |
| Purge % H (Purge gas composition of component H) | RMSE (e-3) | 29.197 | 17.641 | 20.902 | **14.314** | 26.191 | 25.443 |
| | $R^2$ | 0.6753 | 0.8815 | 0.8336 | **0.9220** | 0.7387 | 0.7535 |
| | Time(s) | 51.87 | 38.39 | 45.79 | 39.32 | 41.53 | **37.04** |

### *B. CASE STUDY ON TE PROCESS*

This section uses the complex industrial simulation data set of TE process to verify the performance of STA-APSNFIS, and compare it with some state-of-the-art soft sensor models, including LS-SVM [17], TSA-BP [41], PSO-ANFIS [40], GA-ANFIS [40] and ANFIS [40].

TE process is a widely used benchmark process in the field of industrial control and monitoring [42]. The whole process consists of five major units. The reaction contains eight components, of which A, B, C, D, E are reactants and additives, G, H are final products. The reaction is irreversible. The experimental data set on TE process contains 41 measurements, 12 operational variables. In the 41 measurements, they include 22 continuous process variables and 19 sampling process measurements. Since we only consider the key performance indicators of products G and H (purge gas analysis of G and H, in mol), only the data under normal conditions are used for the KPI prediction.

To make a fair comparison, we performed the PCA-based dimension reduction of the process monitoring data for the ANFIS, GA-ANFIS and PSO-ANFIS modeling. 70% randomly-chosen samples were used for model training and the remainder was used as the testing set. Experimental results of these comparative methods are listed in Table 4.

As can be seen from Tab. 4 that the KPI prediction performance of the STA-APSNFIS model is more accurate than the predictions produced by other soft sensor methods, and the convergence time of the proposed STA-APSNFIS on the data set is the shortest.

To make a more direct explanation, parts of test results selected randomly are displayed in Fig.7 and Fig. 8. As shown in the Fig.7 and Fig.8, the fitting effect between the predicted value and the real value observed by the STA-APSNFIS model is more precise than other soft sensor models. It is worth noting that both GA-ANFIS and PSO-ANFIS have certain optimization effects on the original ANFIS model and the PSO-ANFIS model is superior to the GA-ANFIS model. However, the proposed STA-APSNFIS model in this paper has better predictive ability than other soft sensor models in general in terms of the fitting preciseness and the inferential time.

### *C. CASE STUDY ON A REAL FLOTATION PROCESS*

The proposed STA-APSNFIS model was tentatively applied to a real complex industrial process, industrial bauxite flotation process, to further verify its practical performance. The flowchart of the bauxite flotation process is shown in Fig. 9. It is can be seen clearly that the bauxite flotation process is a continuous complex industrial process involving multiple sub-circuits.

This bauxite flotation circuit includes three basic sub-processes: rougher, cleaner (including cleaner I and cleaner II), and scavenger (including roughing scavenger and cleaning scavenger) [7], [8], [43]–[45]. Wherein, the concentrate froth of the rougher sub-process is collected and pumped to the cleaner I for further processing to improve the cleaner grade. The bottom pulp stream of the rough is pumped into

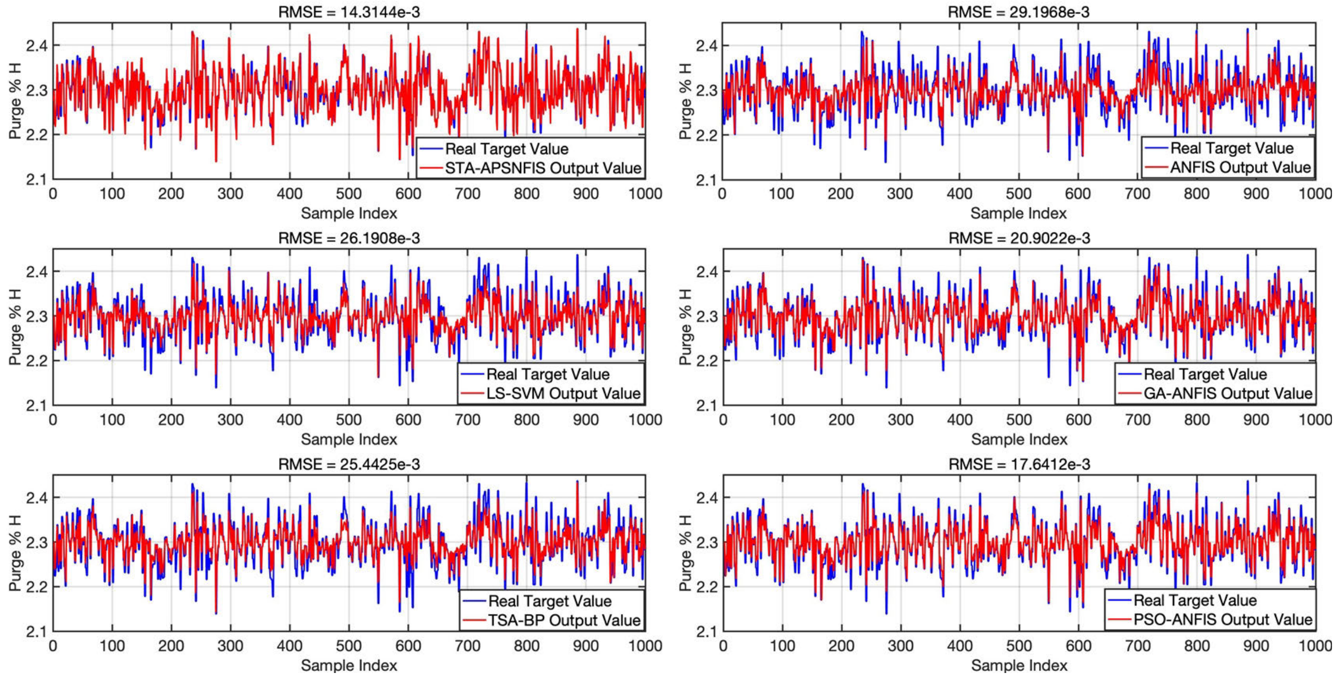



**FIGURE 8.** Purge % H fitting result.

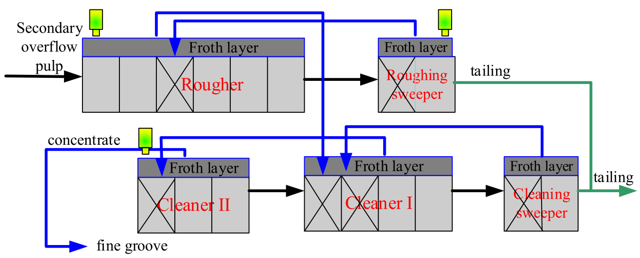



**FIGURE 9.** Bauxite flotation circuit.

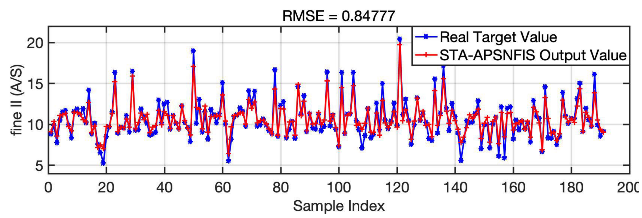



**FIGURE 10.** Concentrate grade prediction of bauxite flotation process.

the roughing sweeper tank to recover the mineral particles that have not been floated by the rough process. The coarse froth (roughing sweep product) is pumped to the rough sub-process for the re-treatment, and the bottom pulp of the coarse sweep is discharged as the tailings. The froth layer of the cleaner I is sent to the cleaner II sub-process, and the final concentrate froth of the cleaner II are subjected to a series of subsequent treatments such as thickening and drying as the alumina beneficiation product of the floatation process. The underflow of the cleaner I is sent to the concentrate sweeper to further recover the unrecovered mineral particles and the fine sweep bottom stream is discharged as tailings.

The froth layer of the cleaner II tank will be collected as the final concentrate product and the concentrate grade (A/S) is still currently unavailable for the on-line testing. Generally, it can only rely on offline sampling and laboratory assays. Laboratory assays are time consuming and labor intensive, and generally only one or two detection values are obtained per day. Therefore, due to the lack of on-line monitoring value of concentrate grade, it is impossible to effectively evaluate the product conditions, in other words, it is difficult to make effective operational adjustments in time to ensure steady-state optimal operation of production.

Therefore, this paper mainly focuses on the beneficiation index (A/S) of the cleaner II tank as the detection object. In the experiment, the process parameters data during the bauxite flotation process was collected for one moth (26 days), including the characteristics of the flotation froth images under various working conditions and the corresponding metallurgical process parameters and concentrate grade data by manual sampling. After the manual collection of concentrate samples and analysis of 8 samples of A/S per day and remove the bad data (process data collect on the repair time of field equipment failure), a total of 197 valid A/S data of manual testing was obtained. 120 of them were randomly selected for model training, and the rest were used for model testing. The monitoring effect on the data set of the STA-APSNFIS is as shown in the Fig. 10.

In order to further evaluate the proposed STA-APSNFIS, we carried out experiments comparing the concentrate grade (A/S of cleaner II) prediction results with that of the soft sensor methods based on ANFIS, PSO-ANFIS, GA-ANFIS, LS-SVM, TSA-BP and STA-APSNFIS in this

**TABLE 5. Comparative result on Bauxite flotation Process.**

| | Statistic | ANFIS [40] | PSO-ANFIS[40] | GA-ANFIS [40] | STA-APSNFIS | LS-SVM [17] | TSA-BP [41] |
|---|---|---|---|---|---|---|---|
| Cleaner II (A/S) | RMSE(e-3) | 1.2397 | 0.93248 | 1.1437 | **0.84777** | 1.4826 | 1.269 |
| | $R^2$ | 0.74512 | 0.85581 | 0.78307 | **0.88081** | 0.63546 | 0.73296 |
| | Time(s) | 36.42 | 25.91 | 29.47 | 11.81 | **11.76** | 26.16 |

flotation process. Quantitative evaluation results can be seen in the Tab. 5.

As can be seen from Tab. 5, in terms of RMSE and $R^2$, STA-APSNFIS can achieve the best results. It indicates that the soft sensor proposed in this paper is feasible in the practical applications of CIPs and it has certain advantages in high-dimensional complex system modeling compared with other soft sensor models. With respect to the time cost, it is slightly lower than the LS-SVM model. However, the response time (around 11s on the test computer) can meet the requirement of the real time industrial process monitoring.

To summary, the application results on the bauxite flotation process show that STA-APSNFIS can effectively predict the concentrate grade of bauxite flotation process, which provides a technique support of online concentrate grade monitoring to ensure stable and optimal operation of the flotation process. It provides an effective reference index for improving the product qualities and reducing the consumptions of mineral resources in the mineral processing.

## V. CONCLUSION

This paper proposes an end-to-end soft-measurement method based on the STA-optimized adaptive pre-sparse neural-fuzzy inference system for online KPI monitoring of CIPs, termed STA-APSNFIS. STA-APSNFIS introduces a pre-sparse neural network to the traditional ANFIS to reduce the adverse effect of data redundancy, measurement noise and inaccuracies in the industrial process data, which can reduce the complexity of the FIS models and effectively solve the problem of traditional ANFIS model being hard to fit high-dimensional system. In addition, the optimal parameters of the APSNFIS model are achieved by the STA optimization algorithm. Experimental results show that the proposed method can effectively avoid the local optimum solution, which has faster convergence speed and better fitting effect than ANFIS, PSO-ANFIS, GA-ANFIS and LS-SVM-based soft-sensor methods, for the online KPI monitoring of CIPs. The proposed method was also validated in a real industrial bauxite flotation process for the online monitoring of the concentrate grade. Experimental results on the real industrial process show that the proposed method can effectively predict the concentrate grade of bauxite flotation process in real time, which lays a foundation to ensure the stable and optimal operation of flotation process to improve the beneficiation performance. The next step is to combine STA-APSNFIS with time series-based industrial process modeling and monitoring approaches to achieve more accurate monitoring results of KPIs for the CIP monitoring. Then, the corresponding process optimization setting method can be put forward to ensure a stable and optimal operation of CIPs, e.g., the industrial flotation process, which lays a foundation for improving the quality of industrial process products and strengthening the industrial control

## REFERENCES

[1] Z. Gao, H. Saxen, and C. Gao, "Guest editorial: Special section on data-driven approaches for complex industrial systems," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2210–2212, Nov. 2013.

[2] S. Xie, Y. Xie, H. Ying, W. Gui, and C. Yang, "A hybrid control strategy for real-time control of the iron removal process of the zinc hydrometallurgy plants," *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5278–5288, Dec. 2018.

[3] K. Popli, A. Afacan, Q. Liu, and V. Prasad, "Development of online soft sensors and dynamic fundamental model-based process monitoring for complex sulfide ore flotation," *Minerals Eng.*, vol. 124, pp. 10–27, Aug. 2018.

[4] F. A. P. Peres and F. S. Fogliatto, "Variable selection methods in multivariate statistical process control: A systematic literature review," *Comput. Ind. Eng.*, vol. 115, pp. 603–619, Jan. 2018.

[5] J. Liu, J. He, Y. Xie, W. Gui, Z. Tang, T. Ma, J. He, and J. P. Niyoyita, "Illumination-invariant flotation froth color measuring via Wasserstein distance-based CycleGAN with structure-preserving constraint," *IEEE Trans. Cybern.*, early access, Mar. 18, 2020, doi: 10.1109/TCYB.2020.2977537.

[6] W. Gui, J. Liu, C. Yang, N. Chen, and X. Liao, "Color co-occurrence matrix based froth image texture extraction for mineral flotation," *Minerals Eng.*, vols. 46–47, pp. 60–67, Jun. 2013.

[7] J. Liu, W. Gui, Z. Tang, H. Hu, and J. Zhu, "Machine vision based production condition classification and recognition for mineral flotation process monitoring," *Int. J. Comput. Intell. Syst.*, vol. 6, no. 5, pp. 969–986, Sep. 2013.

[8] J. Liu, J. Zhou, Z. Tang, W. Gui, Y. Xie, J. He, T. Ma, and J. P. Niyoyita, "Toward flotation process operation-state identification via statistical modeling of biologically inspired Gabor filtering responses," *IEEE Trans. Cybern.*, early access, Apr. 24, 2019, doi: 10.1109/TCYB.2019.2909763.

[9] Y. Jiang, J. Fan, T. Chai, and F. L. Lewis, "Dual-rate operational optimal control for flotation industrial process with unknown operational model," *IEEE Trans. Ind. Electron.*, vol. 66, no. 6, pp. 4587–4599, Jun. 2019.

[10] N.-N. Zhang, C.-C. Zhou, J.-H. Pan, W. Xia, C. Liu, M.-C. Tang, and S.-S. Cao, "The response of diasporic-bauxite flotation to particle size based on flotation kinetic study and neural network simulation," *Powder Technol.*, vol. 318, pp. 272–281, Aug. 2017.

[11] L. Yao and Z. Ge, "Deep learning of semisupervised process data with hierarchical extreme learning machine and soft sensor application," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1490–1498, Feb. 2018.

[12] C. Shang, F. Yang, D. Huang, and W. Lyu, "Data-driven soft sensor development based on deep learning technique," *J. Process Control*, vol. 24, no. 3, pp. 223–233, Mar. 2014.

[13] Q. Sun and Z. Ge, "Probabilistic sequential network for deep learning of complex process data and soft sensor application," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 2700–2709, May 2019.

[14] T. Ait-Izem, M.-F. Harkat, M. Djeghaba, and F. Kratz, "On the application of interval PCA to process monitoring: A robust strategy for sensor FDI with new efficient control statistics," *J. Process Control*, vol. 63, pp. 29–46, Mar. 2018.

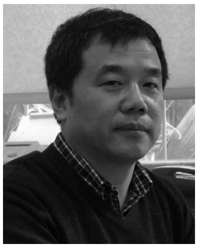
[15] Y. Zhao, R. N. Dai, X. Xiao, Z. Zhang, L. Duan, Z. Li, and C. Z. Zhu, "Independent component analysis-based source-level hyperlink analysis for two-person neuroscience studies," *J. Biomed. Opt.*, vol. 22, pp. 1–12, Feb. 2017.

[16] Y. Liu, Y. Pan, and D. Huang, "Development of a novel adaptive soft-sensor using variational Bayesian PLS with accounting for online identification of key variables," *Ind. Eng. Chem. Res.*, vol. 54, no. 1, pp. 338–350, Jan. 2015.
[17] X. Lu, W. Zou, and M. Huang, "A novel spatiotemporal LS-SVM method for complex distributed parameter systems with applications to curing thermal process," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 1156–1165, Jun. 2016.
[18] Y. Ögren, P. Tóth, A. Garami, A. Sepman, and H. Wiinikka, "Development of a vision-based soft sensor for estimating equivalence ratio and major species concentration in entrained flow biomass gasification reactors," *Appl. Energy*, vol. 226, pp. 450–460, Sep. 2018.
[19] L. Renson, J. Sieber, D. A. W. Barton, A. D. Shaw, and S. A. Neild, "Numerical continuation in nonlinear experiments using local Gaussian process regression," *Nonlinear Dyn.*, vol. 98, no. 4, pp. 2811–2826, Dec. 2019.
[20] J. Liu, W. Zhang, Z. Tang, Y. Xie, T. Ma, J. Zhang, G. Zhang, and J. P. Niyoyita, "Adaptive intrusion detection via GA-GOGMM-based pattern learning with fuzzy rough set-based attribute selection," *Expert Syst. Appl.*, vol. 139, pp. 112845-1–112845-17, Jan. 2020.
[21] J. Liu, W. Zhang, T. Ma, Z. Tang, Y. Xie, W. Gui, and J. P. Niyoyita, "Toward security monitoring of industrial cyber-physical systems via hierarchically distributed intrusion detection," in *Expert Systems With Applications*. Philadelphia, PA, USA: InPress, 2020, pp. 1–23, doi: 10.1016/j.eswa.2020.113578.
[22] A. K. Pani and H. K. Mohanta, "Online monitoring of cement clinker quality using multivariate statistics and takagi-sugeno fuzzy-inference technique," *Control Eng. Pract.*, vol. 57, pp. 1–17, Dec. 2016.
[23] Z. Ge, "Review on data-driven modeling and monitoring for plant-wide industrial processes," *Chemometric Intell. Lab. Syst.*, vol. 171, pp. 16–25, Dec. 2017.
[24] Z. Ge and J. Chen, "Plant-wide industrial process monitoring: A distributed modeling framework," *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 310–321, Feb. 2016.
[25] R. Coteli, H. Acikgoz, F. Ucar, and B. Dandil, "Design and implementation of Type-2 fuzzy neural system controller for PWM rectifiers," *Int. J. Hydrogen Energy*, vol. 42, no. 32, pp. 20759–20771, Aug. 2017.
[26] R. Javadpour and G. M. Knapp, "A fuzzy neural network approach to machine condition monitoring," *Comput. Ind. Eng.*, vol. 45, no. 2, pp. 323–330, Aug. 2003.
[27] H. R. Baghaee, M. Mirsalim, G. B. Gharehpetian, H. A. Talebi, and A. Niknam-Kumle, "A hybrid ANFIS/ABC-based online selective harmonic elimination switching pattern for cascaded multi-level inverters of microgrids," *IEEE Trans. Ind. Electron.*, early access, Apr. 26, 2017, doi: 10.1109/TIE.2017.2694403.
[28] W. Yu and C. Zhao, "Robust monitoring and fault isolation of nonlinear industrial processes using denoising autoencoder and elastic net," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 3, pp. 1083–1091, May 2020.
[29] J.-S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 3, pp. 665–685, May/Jun. 1993.
[30] D. Karaboga and E. Kaya, "Adaptive network based fuzzy inference system (ANFIS) training approaches: A comprehensive survey," *Artif. Intell. Rev.*, vol. 52, pp. 2263–2293, Jan. 2018.
[31] S. Shamshirband, J. Iqbal, D. Petković, and M. A. Mirhashemi, "Survey of four models of probability density functions of wind speed and directions by adaptive neuro-fuzzy methodology," *Adv. Eng. Softw.*, vol. 76, pp. 148–153, Oct. 2014.
[32] X. Zhou, D. Y. Gao, C. Yang, and W. Gui, "Discrete state transition algorithm for unconstrained integer optimization problems," *Neurocomputing*, vol. 173, pp. 864–874, Jan. 2016.
[33] X. Zhou, C. Yang, and W. Gui, "Nonlinear system identification and control using state transition algorithm," *Appl. Math. Comput.*, vol. 226, pp. 169–179, Jan. 2014.
[34] X. Zhou, C. Yang, and W. Gui, "Initial version of state transition algorithm," presented at the Int. Conf. Digit. Manuf. Automat., 2011.
[35] X. Zhou, C. Yang, and W. Gui, "State transition algorithm," *J. Ind. Manage. Optim.*, vol. 8, no. 4, pp. 1039–1056, 2012.
[36] J. Han, C. Yang, X. Zhou, and W. Gui, "A new multi-threshold image segmentation approach using state transition algorithm," *Appl. Math. Model.*, vol. 44, pp. 588–601, Apr. 2017.
[37] S. Laohakiat, S. Phimoltares, and C. Lursinsap, "A clustering algorithm for stream data with LDA-based unsupervised localized dimension reduction," *Inf. Sci.*, vol. 381, pp. 104–123, Mar. 2017.
[38] N. M. Peleato, R. L. Legge, and R. C. Andrews, "Neural networks for dimensionality reduction of fluorescence spectra and prediction of drinking water disinfection by-products," *Water Res.*, vol. 136, pp. 84–94, Jun. 2018.
[39] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, Jun. 2003.
[40] M. Rezakazemi, A. Dashti, M. Asghari, and S. Shirazian, "$H_2$-selective mixed matrix membranes modeling using ANFIS, PSO-ANFIS, GA-ANFIS," *Int. J. Hydrogen Energy*, vol. 42, no. 22, pp. 15211–15225, Jun. 2017.
[41] F. Hu, M. Wang, Y. Zhu, J. Liu, and Y. Jia, "A time simulated annealing-back propagation algorithm and its application in disease prediction," *Mod. Phys. Lett. B*, vol. 32, no. 25, 2018, Art. no. 1850303.
[42] B.-C. Wang and H.-X. Li, "A sliding window based dynamic spatiotemporal modeling for distributed parameter systems with time-dependent boundary conditions," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2044–2053, Apr. 2019.
[43] J. Liu, J. He, W. Zhang, P. Xu, and Z. Tang, "TCvBsISM: Texture classification via B-Splines-Based image statistical modeling," *IEEE Access*, vol. 6, pp. 44876–44893, 2018.
[44] J. Liu, W. Gui, and Z. Tang, "Flow velocity measurement and analysis based on froth image SIFT features and Kalman filter for froth flotation," *Turkish J. Elect. Eng. Comput. Standardcience*, vol. 21, pp. 2378–2396, Dec. 2013.
[45] J. Liu, W. Gui, Z. Tang, C. Yang, J. Zhu, and J. Li, "Recognition of the operational statuses of reagent addition using dynamic bubble size distribution in copper flotation process," *Minerals Eng.*, vol. 45, pp. 128–141, May 2013.

**JINPING LIU** (Member, IEEE) received the B.Eng. degree in computer science and technology, the M.Eng. degree in computer application, and the Ph.D. degree in control science and engineering from the School of Information Science and Engineering, Central South University, in 2006, 2009, and 2013, respectively. He is currently an Associate Professor with the College of Mathematics and Computer Science, Hunan Normal University. His research interests include digital signal processing and pattern recognition.

**CHURONG JIANG** received the B.Eng. degree in computer science and technology from Henan Agricultural University, in 2018. She is currently pursuing the master's degree in computer application with the College of Mathematics and Computer Science, Hunan Normal University. Her research interests include computer vision and pattern recognition with their industrial applications.

**JIEZHOU HE** received the B.Eng. degree in computer science and technology from Hunan Normal University, where he is currently pursuing the master's degree with the College of Mathematics and Computer Science. His research interests include computer vision and pattern recognition.



**ZHAOHUI TANG** received the Ph.D. degree from Central South University (CSU), China, in 2008. He was a Visiting Scholar with the University of Duisburg-Essen, Germany, from 2005 to 2006. He is currently a Professor with the School of Information Science and Engineering, CSU. His research interests include signal processing and industrial process fault diagnosis.

**YONGFANG XIE** (Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees in control science and engineering from Central South University, Changsha, China, in 1993, 1996, and 1999, respectively. From 1999 to 2003, he was with the Tokyo International Information Science Research Institute, Tokyo, Japan, and a Visiting Scholar with the PToPA Research Institute, Tokyo. He is currently a Professor with the School of Automation, Central South University. His research interests include modeling, optimization, and control of complicated industrial processes, distributed robust control, and fault diagnoses.

**PENGFEI XU** received the B.Eng. degree in computer application from Hunan Normal University, in 2000, the M.Eng. degree in computer science and technology from Nankai University, in 2003, and the Ph.D. degree in computer application technology from Central South University, in 2012. He is currently an Associate Professor with the College of Mathematics and Computer Science, Hunan Normal University. His research interests include computer vision and pattern recognition.